



# SCIENTIFIC REPORTS



OPEN

## Network-aided Bi-Clustering for discovering cancer subtypes

Guoxian Yu , Xianxue Yu & Jun Wang 

Received: 27 October 2016  
Accepted: 28 March 2017  
Published online: 21 April 2017

Bi-clustering is a widely used data mining technique for analyzing gene expression data. It simultaneously groups genes and samples of an input gene expression data matrix to discover bi-clusters that relevant samples exhibit similar gene expression profiles over a subset of genes. The discovered bi-clusters bring insights for categorization of cancer subtypes, gene treatments and others. Most existing bi-clustering approaches can only enumerate bi-clusters with constant values. Gene interaction networks can help to understand the pattern of cancer subtypes, but they are rarely integrated with gene expression data for exploring cancer subtypes. In this paper, we propose a novel method called Network-aided Bi-Clustering (NetBC). NetBC assigns weights to genes based on the structure of gene interaction network, and it iteratively optimizes sum-squared residue to obtain the row and column indicative matrices of bi-clusters by matrix factorization. NetBC can not only efficiently discover bi-clusters with constant values, but also bi-clusters with coherent trends. Empirical study on large-scale cancer gene expression datasets demonstrates that NetBC can more accurately discover cancer subtypes than other related algorithms.

Gene expression means that cells transfer the genetic information in deoxyribonucleic acid (DNA) into a protein molecule with biological activity through transcription and translation in life process<sup>1</sup>. Microarray techniques enable researchers simultaneously measure expression levels of numerous genes<sup>2</sup>. The measurements of gene expression under many specific conditions are often represented as a gene expression data matrix, of which each row corresponds to a gene and each column represents the expression levels under a specific condition<sup>3,4</sup>. The specific conditions usually relate to environments, patients, tissues, time points, and they are also synonymously called as samples. One key step of analyzing gene expression data is to identify clusters of genes, or of conditions<sup>4</sup>. For example, cancer can be classified into subtypes based on the pervasive differences in their gene expression patterns, and thus to provide a cancer patient with precise treatment<sup>5,6</sup>.

Many clustering approaches have been proposed to analyze gene expression data, such as *k*-means<sup>7</sup>, hierarchical clustering<sup>8</sup>, local self-organizing maps<sup>9</sup>, local adaptive clustering<sup>10</sup> and so on. Tavazoie *et al.*<sup>7</sup> applied *k*-means to group gene expression data by assigning a sample to its nearest centroid, which is calculated by averaging all samples in that cluster. Eisen *et al.*<sup>8</sup> applied average-link hierarchical clustering to cluster yeast gene expression data. Hierarchical clustering iteratively merges two closest clusters from singleton clusters, or partitions clusters into sub-ones by taking all samples as the single initial cluster. Average-link method uses the average distance between members of two clusters. These approaches have enabled researchers to explore the association between biological mechanisms and different physiological states, as well as to identify gene expression signatures. These traditional approaches, however, separately group gene expression data from genes dimension only. They can not discover the patterns that similar genes exhibit similar behaviors only over a subset of conditions (or samples), or relevant samples exhibit similar expression profiles over a subset of genes<sup>11</sup>. Patients of a cancer subtype may show similar expression profiles on a number of genes, instead of all<sup>5,6</sup>.

Bi-clustering becomes an alternative to traditional clustering approaches for gene expression data analysis. Bi-clustering (or co-clustering), simultaneously groups genes and samples, it aims at discovering the patterns (or bi-clusters) that some genes exhibit similar expression values only on a subset of conditions<sup>12,13</sup>. One can obtain a set of genes co-regulated under a set of conditions via bi-clustering. Bi-clustering shows great potentiality to find biological significance patterns<sup>14</sup>, which usually include: with constant values in the entire bi-cluster, with constant values in rows, with constant values in columns, with additive constant values and with multiplicative coherent values<sup>15,16</sup>.

Many bi-clustering techniques have been applied to gene expression data analysis<sup>17</sup>. Cheng *et al.*<sup>12</sup> pioneered a bi-clustering solution for grouping gene expression data, whose exact solution is known as a NP-hard problem.

College of Computer and Information Science, Southwest University, Chongqing, China. Correspondence and requests for materials should be addressed to J.W. (email: [kingjun@swu.edu.cn](mailto:kingjun@swu.edu.cn))

To combat with this problem, they used a greedy search to discover bi-clusters with low mean-squared residue score. Particularly, they iteratively removed or added genes and conditions from gene expression data matrix to find a bi-cluster, whose mean-squared residue score is below a certain threshold. However, this iterative solution can only produce one bi-cluster at a time, and it is hard to set a suitable threshold. Bergmann *et al.*<sup>18</sup> proposed an iterative signature algorithm to iteratively search bi-clusters based on two pre-determined thresholds, one for matrix rows (representing genes) and the other one for matrix columns (representing samples). Obviously, the specification of these two thresholds affects the composition of bi-clusters. Therefore, similar as the solution proposed by Cheng *et al.*<sup>12</sup>, this signature algorithm is also heavily dependent on suitable setting of thresholds.

Researchers also move toward concurrently discovering multiple bi-clusters at a time. For instance, bi-clustering based on graph theory<sup>19,20</sup>, information theory<sup>21</sup>, statistical method<sup>22</sup>, matrix factorization<sup>23</sup>. Sun *et al.*<sup>24</sup> contributed a heuristic algorithm called Biforce, which transforms the data matrix into a weighted bipartite graph and judges the connection between nodes by a user-specified similarity threshold. Next, Biforce edits the bipartite graph by deleting or inserting edges to obtain bi-clusters. Kluger *et al.*<sup>20</sup> proposed a spectral bi-clustering algorithm to simultaneously group genes and samples to find distinctive patterns from gene expression data. This algorithm is based on the observation that the structure of gene expression data can be found in the eigenvectors across genes or samples. It firstly computes several largest left and right singular vectors of the normalized gene expression data matrix, and then uses normalized cut<sup>25</sup> or *k*-means on the matrix reconstructed by the left and right eigenvectors to obtain the row and column labels. Shan *et al.*<sup>22</sup> proposed Bayesian co-clustering (BCC). BCC assumes that the genes (or samples) of gene expression data are generated by a finite mixture of underlying probability distributions, i.e., multivariate normal distribution. The entry of gene expression data matrix can be generated by the joint distributions of genes and conditions. One advantage of BCC is that it computes the probability of genes (or samples) belonging to several bi-clusters, instead of exclusively partitioning genes (or samples) into only one bi-cluster, but BCC suffers from a long runtime cost on large scale gene expression data. Dhillon *et al.*<sup>21</sup> proposed an information-theoretic bi-clustering algorithm. Likewise BCC, this algorithm views the entry of gene expression data matrix as the estimation of joint probability of row-column distributions and optimizes these distributions by maximizing mutual information between entries of bi-clusters. But this approach is restricted to non-negative data matrix. Murali *et al.*<sup>26</sup> proposed the concept of conserved gene expression motifs (xMOTIFS), each motif is defined as a subset of genes whose expressions are simultaneously conserved for a subset of samples. xMOTIFS aims to discover large conserved gene motifs that cover all the samples and classes in the data matrix. Hochreiter *et al.*<sup>27</sup> proposed a factor analysis bi-clustering (FABIA) algorithm based on multiplicative model, which accounts for the linear dependency between gene expression profiles and samples. Lazzeroni *et al.*<sup>28</sup> proposed a plaid model bi-clustering (Plaid), the entries of each bi-cluster are modelled by a general additive model and extracted by row and column indicator variables.

To explore bi-clusters with coherent trends, Cho *et al.*<sup>29</sup> proposed a minimum sum-squared residue co-clustering (MSSRCC) solution to identify bi-clusters. MSSRCC iteratively obtains row and column clusters by a *k*-means like algorithm on row and column dimensions while monotonically decreasing the sum-squared residue. MSSRCC can discover multiple bi-clusters with coherent trends, or constant values. Gene expression data are always with a limit number of samples but with thousands of genes<sup>4</sup>. Distance between samples turns to be isometric as the number of genes (or gene dimension) increase<sup>30</sup>. MSSRCC firstly reduces the gene dimension by choosing genes with large deviation of expression levels among samples, and then applies bi-clustering on the pre-selected gene expression data to identify bi-clusters. However, this selection may lose the information hidden in the gene expression data, since the biological sense is not always straight<sup>31</sup>.

More recently, molecular interaction networks are also incorporated into bi-clustering to improve the performance of discovering cancer subtypes<sup>32–35</sup>. Knowing the subtype of a cancer patient can provide directional clues for precise treatment. Hofree *et al.*<sup>36</sup> proposed a network-based stratification method to integrate somatic cancer with gene interaction networks. This approach initially groups cancer patients with mutations in similar network regions and then performs bi-clustering on the gene expression profiles using graph-regularized non-negative matrix factorization<sup>37</sup>. Liu *et al.*<sup>38</sup> proposed a network-assisted bi-clustering (NCIS) to identify cancer subtypes via semi-non-negative matrix factorization<sup>37</sup>. NCIS assigns weights to genes as the importance indicator of genes in the clustering process. The weight of each gene refers to both the gene interaction network and gene expression profiles. However, NCIS can only discover bi-clusters with constant values.

The identified bi-clusters by a bi-clustering algorithm depend on the adopted objective function of that algorithm<sup>17</sup>. MSSRCC uses sum-squared residue as the objective function but it does not incorporate the gene interaction network. NCIS assigns weights to genes by referring to both the absolute deviation of genes expression profiles among samples and gene interaction network, but NCIS can only find bi-clusters with constant values, since its objective function is to minimize the distance between all entries of a bi-cluster and the centroid, defined by average of all entries in that bi-cluster.

To simultaneously discover multiple bi-clusters with constant or coherent values and to synergy bi-clustering with gene interaction network for cancer subtypes discovery, we introduce a novel method called Network aided Bi-Clustering (NetBC for short). NetBC firstly assigns weights to genes based on the structure of gene interaction network and the deviation of gene expression profiles. Next, it iteratively optimizes sum-squared residue to generate the row and column indicative matrices by matrix factorization. After that, NetBC takes advantage of the row and column indicative matrices to generate bi-clusters. To quantitatively and comparatively study the performance of NetBC, we test NetBC and other related comparing methods on several publicly available cancer gene expression datasets from The Cancer Genome Atlas (TCGA) project<sup>39</sup>. We use the clinical features of patients to evaluate the performance because the true subtypes of these samples belonging to are unknown. Experimental results show that NetBC can better group patients into subtypes than comparing methods. We further conduct experiments on cancer gene expression datasets with known subtypes to comparatively study the performance of NetBC. NetBC again demonstrates better results than these methods.

## Results and Discussion

To comparatively evaluate the performance of the proposed NetBC, we compared NetBC with NCIS<sup>38</sup>, MSSRCC<sup>29</sup>, BCC<sup>22</sup>, Cheng and Church (CC)<sup>12</sup>, BiMax<sup>14</sup>, Biforce<sup>24</sup>, xMOTIFs<sup>26</sup>, FABIA<sup>27</sup>, and Plaid<sup>28</sup>. Since NCIS, MSSRCC and BCC aim to extract non-overlapping bi-clusters with checkerboard structure, we compare NetBC with NCIS, MSSRCC and BCC on separating samples on two large scale cancer gene expression datasets from TCGA<sup>34</sup> and several cancer gene expression datasets with known subtypes. CC, BiMax, Biforce, xMOTIFs, FABIA, and Plaid aim to extract arbitrarily positioned overlapping bi-clusters, we compare NetBC with CC, BiMax, Biforce, xMOTIFs, FABIA, and Plaid by relevance and recovery on synthetic datasets with implanted bi-clusters.

**Determining the number of Gene clusters ( $k$ ) and sample clusters ( $d$ ).** Determining the number of clusters is a challenge for most clustering methods. Here we adopt a widely used method to find the number of gene clusters  $k$  (or sample clusters  $d$ ) that best fits the gene expression data matrix<sup>38,40</sup>. If the number of gene clusters  $k$  (or sample clusters  $d$ ) is suitable, we would expect that the gene separation (or sample separation) would change very little in different runs. For each run, we define an adjacency matrix of genes  $\mathbf{M}_g$  with size  $m \times m$  and an adjacency matrix of samples  $\mathbf{M}_s$  with size  $n \times n$ ,  $\mathbf{M}_g(i, j) = 1$  when gene  $i$  and gene  $j$  belong to the same cluster and  $\mathbf{M}_g(i, j) = 0$ , otherwise. Similarly,  $\mathbf{M}_s(i, j) = 1$  when sample  $i$  and sample  $j$  belong to the same cluster and  $\mathbf{M}_s(i, j) = 0$ , otherwise. The consensus matrices  $\overline{\mathbf{M}}_g$  and  $\overline{\mathbf{M}}_s$  are the average of many base  $\mathbf{M}_g$  and  $\mathbf{M}_s$ , which are obtained by repeatedly running the clustering method. The entry of  $\overline{\mathbf{M}}_g$  ( $\overline{\mathbf{M}}_s$ ) is among 0 and 1.  $\overline{\mathbf{M}}_g$  reflects the similarity between genes and  $1 - \overline{\mathbf{M}}_g$  denotes the distance between genes. If the selection of  $k$  is suitable,  $\overline{\mathbf{M}}_g$  is rather stable among multiple runs. In other words,  $\overline{\mathbf{M}}_g(i, j)$  is close to 0 or 1. If the selection of  $d$  is suitable,  $\overline{\mathbf{M}}_s$  will not sharply fluctuate in different runs. We use the cophenetic correlation coefficient  $\rho(\overline{\mathbf{M}}_g)$  and  $\rho(\overline{\mathbf{M}}_s)$  to evaluate the stability of the consensus matrix. The cophenetic correlation coefficient  $\rho(\overline{\mathbf{M}}_g)$  is obtained by calculating Pearson correlation between the distance matrix  $1 - \overline{\mathbf{M}}_g$  and the distance matrix obtained by the linkage used in the reordering of  $\overline{\mathbf{M}}_g$ <sup>38,40</sup>. To determine suitable  $k$  and  $d$ , we evaluate the stability of bi-clustering results over a range of combinations with  $k$  and  $d$ . We select the combination of  $k$  and  $d$  that produces the largest  $\frac{\rho(\overline{\mathbf{M}}_g) + \rho(\overline{\mathbf{M}}_s)}{2}$ .

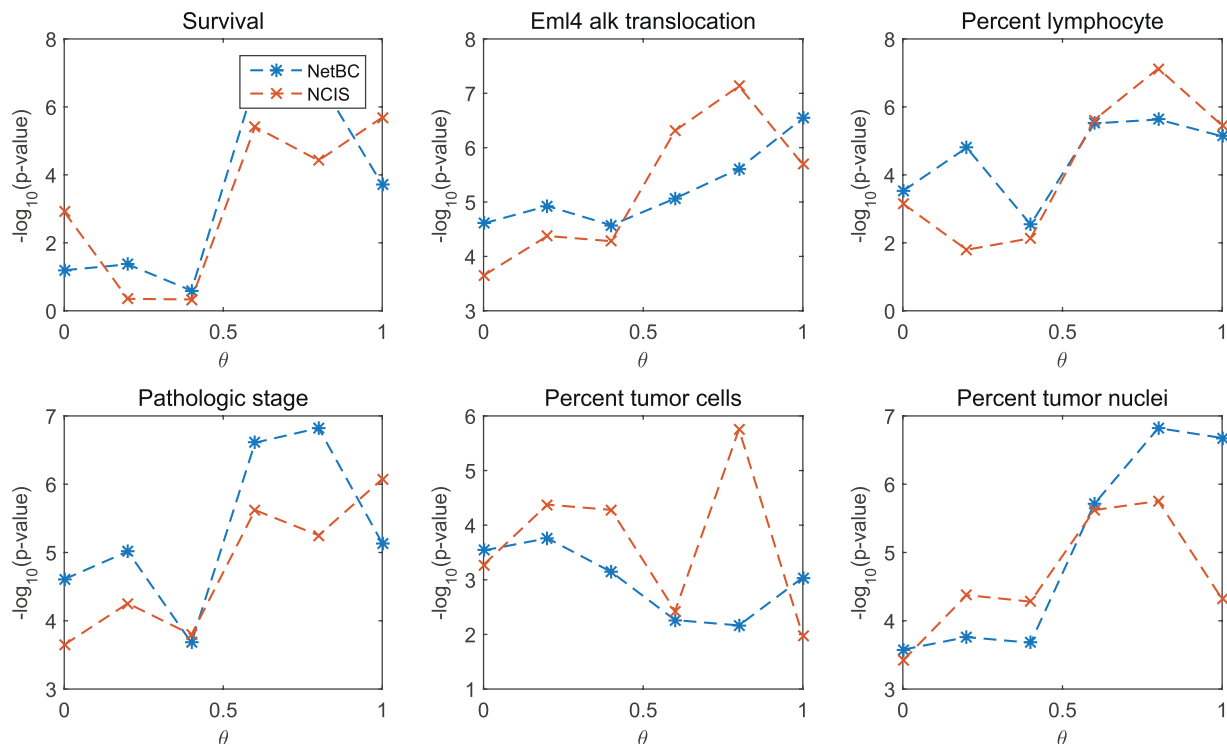
**Results on TCGA cancer gene expression data.** To comparatively evaluate the performance of the proposed NetBC, we compare NetBC with other related and representative bi-clustering methods on separating samples, including NCIS<sup>38</sup>, MSSRCC<sup>29</sup>, and BCC<sup>22</sup> on two large scale cancer gene expression data from TCGA<sup>34</sup>. Since all the selected comparing aim to extract non-overlapping bi-clusters with checkerboard structure, we can use the dependence test between different clinical features and the discovered subtypes to evaluate their performance.

The lung cancer gene expression data contains 1298 patients (samples) with gene expression profiles of 20530 genes. The cancer subtypes of these samples are unknown. For comparison, we adopt the clinical features to study the performance of NetBC and these comparing methods. The clinical features are survival analysis, percent lymphocyte, eml4 alk translocation performed, pathologic stage, percent tumor cells stage, percent tumor nuclei. We choose relapse-free survival (RFS) for survival analysis. RFS means the length of time after primary treatment to a cancer patient that survives without any signs or symptoms of that cancer. RFS is one way to measure how well the treatment works. Percent lymphocyte means different percentages of infiltration of lymphocyte. Eml4 alk translocation clinical feature means whether Eml4 gene and alk gene are fused, the fusion of these two genes can lead to lung cancer. Pathologic stage represents different stages of the cancer pathologic. Percent tumor cell stage represents the percentages of tumor cells in the total cells. Percent tumor nuclei stage represents the percentages of tumor nuclei in a malignant neoplasm specimen. After removing samples with missing data of clinical features, 486 samples with 20530 genes are remained in the lung cancer dataset.

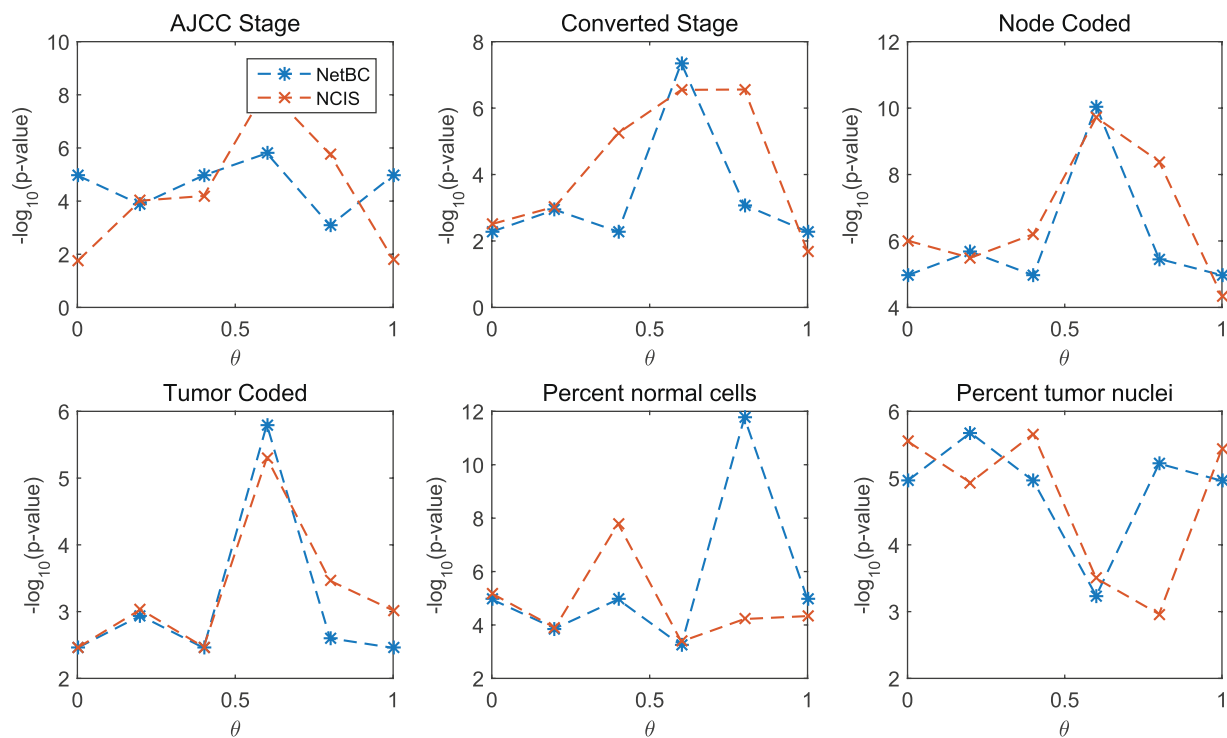
The breast cancer gene expression dataset contains 1241 patients with 17814 genes. Removing the samples that lack of clinical data, we finally retain 416 samples with 17372 genes. The clinical features to evaluate these bi-clustering methods contain AJCC Stage, Converted stage, Node coded, Tumor coded, Percent normal cells, and Percent tumor nuclei. We also make survival analysis for breast cancer, but no method has a  $p$ -value smaller than 0.05, one possible reason is the insufficient clinical data. The AJCC stage represents the different stages of the cancer based on the present lymph nodes. Converted stage represents different stages of the cancer. Node coded means different Node status of patients. Tumor coded means different types of tumor. Percent normal cells represents different percentages of normal cells in the malignant neoplasm specimen. Percent tumor nuclei represents different percentages of the tumor nuclei in the malignant neoplasm specimen.

The gene interaction network used for experiments are combined with networks collected from BioGRID<sup>41</sup> (version: 3.4.138, access date: July 1, 2016), HPRD<sup>42</sup> (version: 9, access date: February 15, 2017) and STRING<sup>43</sup> (version: 10.0, access date: February 15, 2017). To fairly compare NetBC with NCIS, the collected gene interaction network for both NetBC and NCIS is directed. Since the TCGA cancer gene expression data is too large, we use  $k$ -means to initialize the indicative matrix  $\mathbf{R}$  and  $\mathbf{C}$  of NetBC, NCIS and MSSRCC. The number of iterations for these methods is set as 300. We set  $k = 7$  and  $d = 6$  for the TCGA lung cancer gene expression data and  $k = 11$  and  $d = 6$  for the TCGA breast cancer gene expression data based on the cophenetic correlation coefficient over a range of combinations of  $k$  and  $d$  ( $k$  from 1 to 12, and  $d$  from 4 to 6).

$\theta$  is a scalar parameter to balance the contribution of gene interaction network and the deviation of gene expression profiles among samples when assigning genes weights. The significance levels of the difference between different clinical features and subtypes discovered by NetBC and NCIS with a range over different  $\theta$  are given in Fig. 1 (Lung) and Fig. 2 (Breast). The  $p$ -value is adjusted by Benjamini & Hochberg method<sup>44</sup>. From Fig. 1 and Fig. 2, we can see that the input value of  $\theta$  affects the experimental results of NetBC and NCIS.  $\theta \in (0, 1)$  means assigning weights to genes according to both the variation of gene expression levels and gene interaction network. We can find that NetBC and NCIS with  $\theta \in (0.5, 1)$  show better performance than their cousins  $\theta \in (0, 0.5)$  in most



**Figure 1.**  $p$ -value of the dependence test between different clinical features and sub-types of Lung cancer discovered by NetBC (or NCIS) under different input values of  $\theta$ . For the survival time, we use logrank test. For the eml4 alk translocation performed, percent lymphocyte infiltration, percent tumor nuclei, and percent tumor nuclei, we use the Chi-squared test. The  $p$  value is adjusted by Benjamini & Hochberg method. Larger  $-\log_{10}(p\text{-value})$  means better performance.



**Figure 2.**  $p$ -value of the dependence test between different clinical features and sub-types of Breast cancer discovered by NetBC (or NCIS) under different input values of  $\theta$ . For all the clinical features, we use the Chi-squared test. The  $p$  value is adjusted by Benjamini & Hochberg method. Larger  $-\log_{10}(p\text{-value})$  means better performance.

Method	Survival	Eml4 alk translocation	Percent lymphocyte	Pathologic stage	Percent tumor cells	Percent tumor nuclei
NetBC	$3.03 \times 10^{-1}$	<b><math>9.98 \times 10^{-3}</math></b>	$2.95 \times 10^{-2}$	<b><math>9.98 \times 10^{-3}</math></b>	<b><math>2.91 \times 10^{-2}</math></b>	<b><math>2.79 \times 10^{-2}</math></b>
NCIS	$5.42 \times 10^{-2}$	$2.60 \times 10^{-2}$	$4.26 \times 10^{-2}$	$2.60 \times 10^{-2}$	$3.78 \times 10^{-2}$	$3.28 \times 10^{-2}$
MSSRCC	<b><math>1.19 \times 10^{-2}</math></b>	$7.85 \times 10^{-2}$	<b><math>2.48 \times 10^{-2}</math></b>	$4.34 \times 10^{-2}$	$6.30 \times 10^{-2}$	$5.23 \times 10^{-2}$
BCC	$8.54 \times 10^{-1}$	$3.78 \times 10^{-1}$	$8.54 \times 10^{-1}$	$8.54 \times 10^{-1}$	$9.97 \times 10^{-1}$	$7.57 \times 10^{-1}$

**Table 1.** *p* value of the dependence test between different clinical features and the discovered subtypes of Lung cancer. For the survival time, we use logrank test. For the eml4 alk translocation performed, percent lymphocyte infiltration, percent tumor nuclei, and percent tumor nuclei, we use the Chi-squared test. The *p* value is adjusted by Benjamini & Hochberg method. The smaller the *p* value, the better the performance is.

Method	AJCC Stage	Converted Stage	Node Coded	Tumor Coded	Percent normal cells	Percent tumor nuclei
NetBC	<b><math>6.96 \times 10^{-3}</math></b>	$1.02 \times 10^{-1}$	<b><math>6.96 \times 10^{-3}</math></b>	$8.50 \times 10^{-2}$	<b><math>7.01 \times 10^{-3}</math></b>	<b><math>7.01 \times 10^{-3}</math></b>
NCIS	$1.80 \times 10^{-2}$	$4.82 \times 10^{-2}$	$4.11 \times 10^{-3}$	<b><math>4.82 \times 10^{-2}</math></b>	$2.03 \times 10^{-2}$	$7.20 \times 10^{-3}$
MSSRCC	$2.30 \times 10^{-2}$	<b><math>2.62 \times 10^{-2}</math></b>	$1.42 \times 10^{-2}$	$7.13 \times 10^{-2}$	$1.42 \times 10^{-2}$	$3.17 \times 10^{-2}$
BCC	$9.5 \times 10^{-1}$	$9.5 \times 10^{-1}$	$9.5 \times 10^{-1}$	$9.5 \times 10^{-1}$	$9.5 \times 10^{-1}$	$9.5 \times 10^{-1}$

**Table 2.** *p* value of the dependence test between different clinical features and the discovered subtypes of Breast cancer. For all the clinical features, we use the Chi-squared test. The *p* value is adjusted by Benjamini & Hochberg method. The smaller the *p* value, the better the performance is.

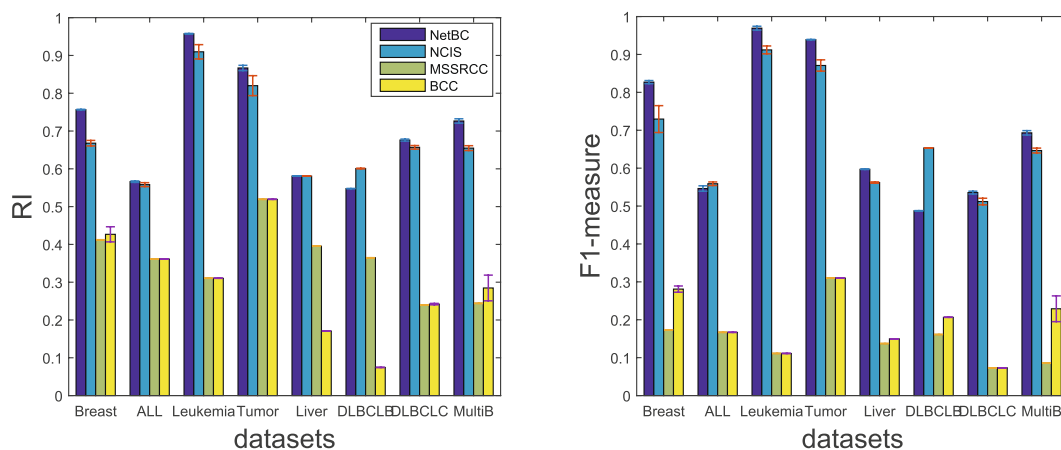
cases. This observation demonstrates that assigning weights to genes according to both the variation of gene expression levels and gene interaction network can improve the performance of bi-clustering than using gene expression profiles along. NetBC also outperforms NCIS in majority cases.

To fairly compare NetBC, NCIS with BCC and MSSRCC,  $\theta$  is setting as 0 for both NetBC and NCIS. Parameters of BCC are the number of gene clusters (or samples clusters), and the initialization of Gaussian distribution ( $\mu, \sigma$ ).  $\mu$  and  $\sigma$  are fixed as random values provided in their demo codes, since there is no prior knowledge about the distribution of samples. The significance levels of the difference between subtypes discovered by these bi-clustering methods and the clinical features are given in Table 1 (Lung) and Table 2 (Breast). The *p*-value is adjusted by Benjamini & Hochberg method<sup>44</sup>. In these tables, a smaller *p* value indicates better results. From Table 1 and Table 2, we can see that NetBC has smaller *p*-value than other methods on most clinical features. Given the *p*-value threshold 0.05, NetBC successfully divides cancer patients into subtypes according to clinical features. BCC can not separate the cancer subtypes as well as others. That is principally because it assumes that patients are generated by a finite mixture of underlying probability distributions. Since the cancer gene expression data has limit samples with a large amount of genes, it is difficult to well estimate these underlying distributions. Although both NetBC and NCIS assign weights to genes as the importance indicator of genes, NetBC performs much better than NCIS on most clinical features. The main difference between them is that the objective function of NetBC is to minimize the sum-squared residue, while NCIS is to minimize the sum-squared distance between entries and centroids of bi-clusters. NCIS can only discover bi-cluster with constant values, NetBC can not only discover bi-clusters with constant values but also bi-clusters with coherent trend values. We can also observe that NetBC outperforms MSSRCC. MSSRCC utilizes a similar objective function as NetBC to minimize the sum-squared residue, the main difference between them is that NetBC assigns weights to genes, but MSSRCC does not. NetBC uses matrix factorization to get row and column indicative matrices and MSSRCC iteratively obtains row and column clusters by a *k*-means like algorithm on row and column dimensions. This observation shows that assigning weights to genes can improve the performance of bi-clustering.

**Results on cancer gene expression data with known subtypes.** We also apply NetBC, NCIS, MSSRCC and BCC in clustering cancer gene expression datasets with known subtypes. Table 3 provides the brief description of these datasets. Breast contains three subtypes: samples from patients who developed distant metastases within 5 years (34 samples), samples from patients who continued to be disease-free after a period of at least 5 years (44 samples), samples from patients with BRCA germline mutation (20 samples). ALL contains three types of leukemia: 19 acute lymphoblastic leukemia (ALL) B-cell, 8 ALL (T-cell), 11 acute myeloid leukemia (AML). Liver contains four subtypes: sprague dawley (67 samples), wistar (32 samples), wistar kyoto (21 samples), fisher (2 samples). Leukemia contains three subtypes: ALL (B-cell) (10 samples), ALL (T-cell) (samples), AML (10 samples). Tumor contains two subtypes: cancer patients (31 samples) and normal (19 samples). DLBCLB includes three subtypes of diffuse large B cell lymphoma, 'oxidative phosphorylation' (42 samples), 'B-cell response' (51 samples), and 'host response' (87 samples). DLBCLC contains four subtypes of diffuse large B cell lymphoma according to statistical differences of the survival analysis: 17 samples, 16 samples, 13 samples, 12 samples. MultiB contains four subtypes: breast cancer (5 samples), prostate cancer (9 samples), lung cancer (7 samples), and colon cancer (11 samples). The adopted gene interaction networks are also collected from BioGRID<sup>41</sup>, HPRD<sup>42</sup> and STRING<sup>43</sup>.

Dataset	Source	#Subtypes(d)	#samples(n)	#genes(m)
Breast	53	3	98	1213
ALL	54	3	38	5571
Leukemia	54	3	30	4412
Tumor	54	2	50	12422
Liver	55	4	122	8799
DLBCLB	56	3	180	661
DLBCLC	56	4	58	3759
MultiB	56	4	32	5565

**Table 3.** Details of 8 cancer gene expression datasets. #Subtypes is the number of cancer subtypes (or clusters), #samples is the number of samples, and #genes is the number of genes.



**Figure 3.** RI (a) and F1-measure (b) of different bi-clustering methods on eight datasets.

Since the ground truth sample clusters of these datasets are known, we adopt two widely used metrics: rand index ( $RI$ )<sup>45</sup> and  $F1$ -measure<sup>46</sup> to evaluate the quality of clustering. Suppose the ground truth subtypes of samples in the gene expression data matrix are  $\mathcal{C} = \{C_1, \dots, C_d\}$ , the clusters produced by a clustering method are  $\mathcal{C}' = \{C'_1, \dots, C'_d\}$ .  $np_1$  represents the number of pairs of samples that are both in the same clusters of  $\mathcal{C}$  and also both in the same clusters of  $\mathcal{C}'$ ;  $np_2$  represents the number of pairs of samples that are in the same clusters of  $\mathcal{C}$  but in different clusters of  $\mathcal{C}'$ ;  $np_3$  represents the number of pairs of samples that are in different clusters of  $\mathcal{C}$  but in the same clusters of  $\mathcal{C}'$ ;  $np_4$  represents the number of pairs of samples that are in different clusters of  $\mathcal{C}$  and also in different clusters of  $\mathcal{C}'$ .  $RI$  is defined as follows:

$$RI = \frac{np_1 + np_4}{np_1 + np_2 + np_3 + np_4}$$

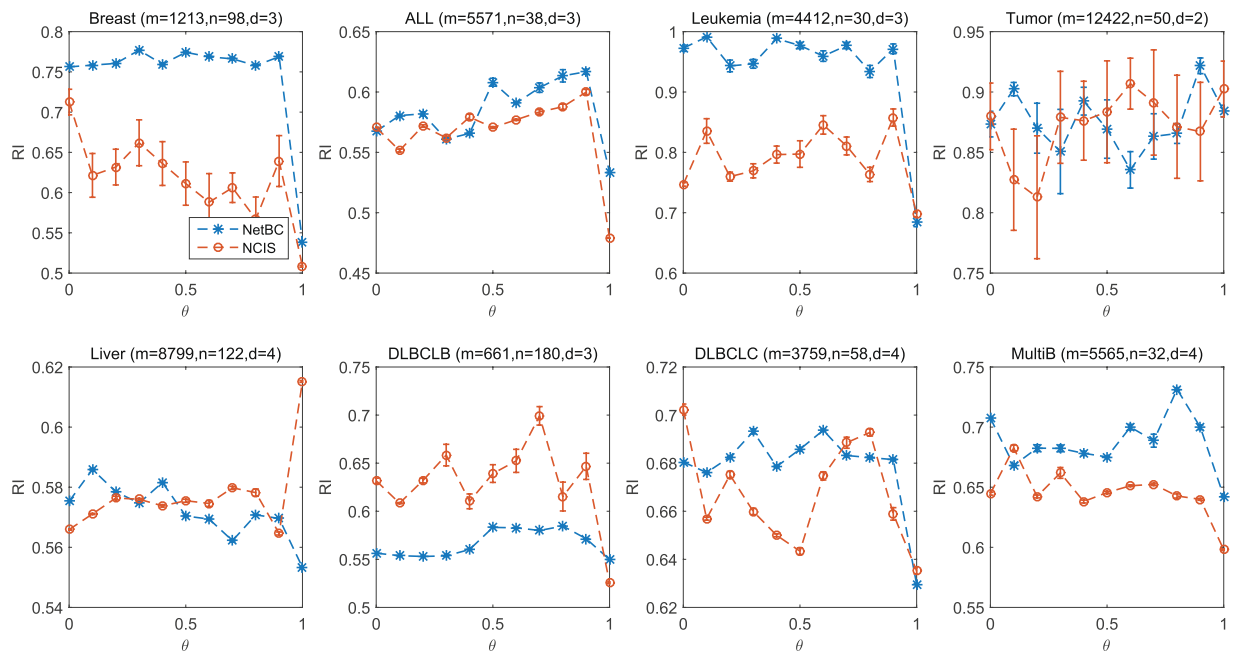
$F1$ -measure is the harmonic mean of precision and recall and is defined as follows:

$$F1 - measure = \frac{2 * Pr * Re}{Pr + Re}$$

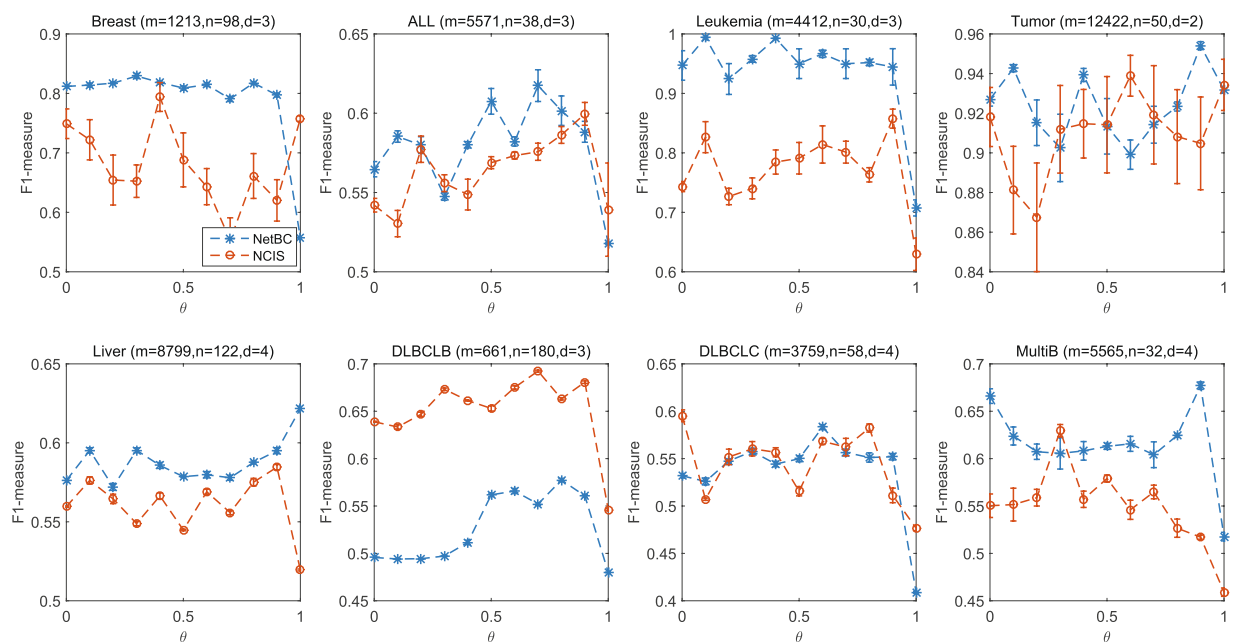
where  $Pr$  and  $Re$  are precision and recall, defined as follows:

$$Pr = \frac{np_1}{np_1 + np_3} \quad Re = \frac{np_1}{np_1 + np_2}$$

We compare the performance of NetBC with NCIS, MSSRCC and BCC. We perform the experiment by randomly initializing the row and column indicative matrices  $\mathbf{R}$  and  $\mathbf{C}$  of NetBC, NCIS and MSSRCC. The number of iterations for NetBC and NCIS is set as 300, and  $\theta = 0$  is used for both NetBC and NCIS. The parameter setting of BCC is similar to the experiment on TCGA cancer gene expression data. We perform 10 independent experiments and report the average results for each method on a particular dataset. We determine the number of gene clusters  $k$  based on the cophenetic correlation coefficient and set  $d$  as the ground truth number of cancer subtypes. Figure 3 provide the experimental results of these comparing methods with respect to RI and F1-measure. We can observe that NetBC performs better than other methods under both RI and F1-measure. This observation indicates that NetBC is an effective bi-clustering approach to identify cancer subtypes.



**Figure 4.** RI under different input values of  $\theta$ .



**Figure 5.** F1-measure under different input values of  $\theta$ .

We also compare NetBC and NCIS when incorporating gene interaction network on real cancer gene expression data with known subtypes.  $\theta$  controls the balance of referring to gene interaction network and the variation of gene expression profiles when assigning weights to genes. We analyze the influence of  $\theta$  by varying its value from 0 to 1 with stepsize 0.1. We perform the experiment by randomly initializing the indicative matrices  $\mathbf{R}$  and  $\mathbf{C}$  of NetBC, NCIS, and fix the number of iterations for both NetBC and NCIS as 300. Figures 4 and 5 reveal the RI and F1-measure of NetBC and NCIS on eight cancer gene expression datasets. The reported experimental results are the average of ten independent runs for each particular dataset under each input value of  $\theta$ . We can see that NetBC consistently outperforms NCIS over a range of  $\theta$  values in most cases. This experimental results again demonstrate that NetBC improves the performance of cancer subtypes discovery. We can observe that NetBC ( $0.1 \leq \theta \leq 0.9$ ) generally has better performance than NetBC when  $\theta = 0$  (or  $\theta = 1$ ). From these observations, we can conclude that assigning weights to genes by referring to both gene expression profiles and gene interaction network shows advantage than assigning weights to genes by using gene interaction network (or by

gene expression profiles) alone. However, there is no clear pattern to choose the most suitable  $\theta$ . The possible reason is that  $\theta$  is not only related to the deviation of expression profiles, but also the quality of gene interaction network and gene expression data. Adaptively choosing a suitable  $\theta$  is an important future pursue. In summary, these empirical study shows that integrating gene interaction networks with gene expression profiles can generally boost the performance of bi-clustering in discovering cancer subtypes.

**Results on synthetic datasets.** Let  $\mathcal{E}$  denote the set of implanted bi-clusters and  $\mathcal{E}'$  denote the set of bi-clusters discovered by a bi-clustering method. We can measure the similarity between the implanted bi-clusters and discovered bi-clusters by using the average bi-cluster relevance score<sup>14</sup> as follows:

$$S_G(\mathcal{E}', \mathcal{E}) = \frac{1}{|\mathcal{E}'|} \sum_{(\mathcal{G}', \mathcal{S}') \in \mathcal{E}'(\mathcal{G}, \mathcal{S}) \in \mathcal{E}} \max \frac{\mathcal{G}' \cap \mathcal{G}}{\mathcal{G}' \cup \mathcal{G}} \quad (1)$$

where each  $\mathcal{E}'$  (or  $\mathcal{E}$ ) contains a gene set  $\mathcal{G}'$  (or  $\mathcal{G}$ ) and a sample set  $\mathcal{S}'$  (or  $\mathcal{S}$ ). The relevance score reflects to what extent the discovered bi-clusters represent implanted bi-clusters in the gene dimension. Similarly, average bi-cluster recovery is defined as  $S_G(\mathcal{E}, \mathcal{E}')$ , which evaluates how well implanted bi-clusters are covered by a bi-clustering method.

Here, we compare NetBC with several bi-clustering methods on synthetic datasets with known bi-clusters. The codes of these comparing methods are online available, including Cheng and Church (CC)<sup>12</sup>, BiMax<sup>14</sup>, FABIA<sup>27</sup>, Plaid<sup>28</sup>, xMOTIFs<sup>26</sup> and Biforce<sup>24</sup>.

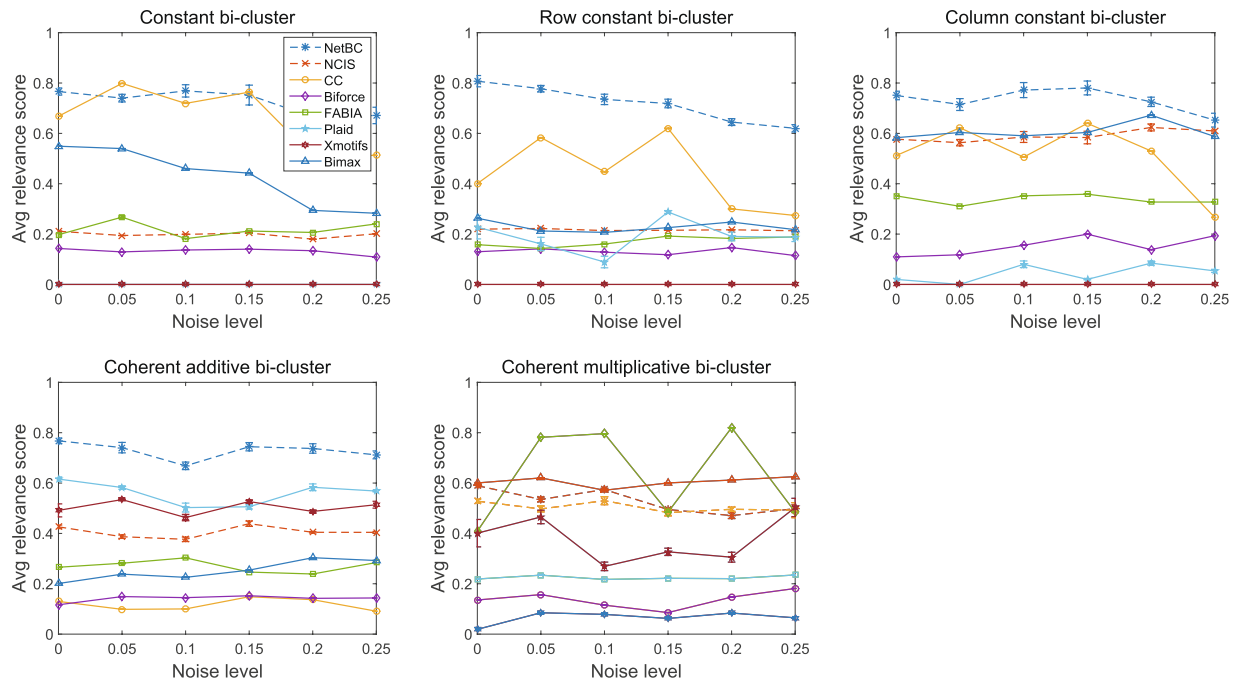
Since NetBC and NCIS aim to discover non-overlapping bi-clusters, the synthetic datasets are generated without overlapping bi-clusters in the same way as that done by Prelic *et al.*<sup>14</sup> and Wang *et al.*<sup>47</sup> Particularly, these synthetic datasets are generated with five different types of bi-clusters, each synthetic dataset contains one type of bi-clusters. The five types of bi-clusters are (i) constant bi-cluster; (ii) row-constant bi-cluster; (iii) column-constant bi-cluster; (iv) additive coherent bi-cluster (or shift bi-cluster); (v) multiplicative coherent bi-cluster (or scale bi-cluster). The background matrices of these synthetic datasets are with entries randomly chosen from Gaussian distribution  $N(0, 1)$ . Each bi-cluster is generated by choosing a submatrix from the background matrix with the entries modified according to one of the five rules: (i) constant bi-cluster is generated by randomly selecting an entry of a submatrix and replacing other entry values with this entry value; (ii) row-constant bi-cluster is generated by randomly selecting a base column within a selected submatrix and copying it to other columns in this submatrix; (iii) column-constant bi-cluster is generated by randomly selecting a base row within a selected submatrix and copying it to other rows in this submatrix; (iv) additive coherent bi-cluster is generated by randomly selecting a base row within a selected submatrix and replacing other rows in this submatrix by shifting the base rows; (v) multiplicative coherent bi-cluster is generated by randomly selecting a base row within a selected submatrix and replacing other rows in this submatrix by scaling the base rows. Then, we also add noise to these synthetic datasets to study the robustness of these comparing methods. Noise is simulated by adding random value from normal distribution to each entry of the synthetic gene expression data matrix. The noise level is increased by enlarging the standard deviation  $\sigma$  from 0.05 to 0.25 with stepsize 0.05.

It is crucial to select suitable parameters for bi-clustering tools. We follow the solution used by Eren *et al.*<sup>48</sup> and Sun *et al.*<sup>24, 49</sup> to select major parameters of these comparing bi-clustering tools over a range of values when they perform the best in the specified range. We set the number of the bi-clusters as the ground truth number of implanted bi-clusters for all bi-clustering approaches.  $\delta$  and  $\alpha$  are critical to the accuracy and runtime of CC.  $\delta$  controls the maximum mean squared-residue in a bi-cluster. By default  $\delta = 1$ , we run CC with different  $\delta$  between 0 and 1 with stepsize 0.01. We set  $\delta = 0.03$ , since CC performs the best when  $\delta = 0.03$ .  $\alpha$  controls the tradeoff between running time and accuracy. By default  $\alpha = 1$ , a larger  $\alpha$  produces higher accuracy but asks for more runtime, we set  $\alpha = 1.5$  since the synthetic datasets are not too large. The number of max iterations for each layer in Plaid affects its results and its default value is 20, and we set it as 50 since Plaid performs best with the number of max iterations fixed as 50. We also select the row.release and column.release of Plaid in [0.5, 1] and set them as 0.6. BiMax and xMOTIFs are dependent on how the data are discretized. We performs xMOTIFs on synthetic data discretized with number of levels from 0 and 50 with stepsize 1. xMOTIFs performs best on synthetic data with the number of levels as 5. BiMax requires binary input data, we perform BiMax on synthetic data that are binary discretized with different thresholds from 0.05 to 1 with stepsize 0.05. BiMax performs best with threshold 0.1. We run Biforce with parameter  $t_0$  (edge threshold) from 0.05 to 1 with stepsize 0.05. Biforce performs best with  $t_0 = 0.1$ . NetBC and NCIS depend on the initialization of indicative matrices  $\mathbf{R}$  and  $\mathbf{C}$ , we randomly initialize them for both NetBC and NCIS for all experiments on synthetic datasets. The number of iterations for NetBC and NCIS is set as 300.  $\theta$  is set as 0, since the gene interaction networks of these synthetic datasets are not available.

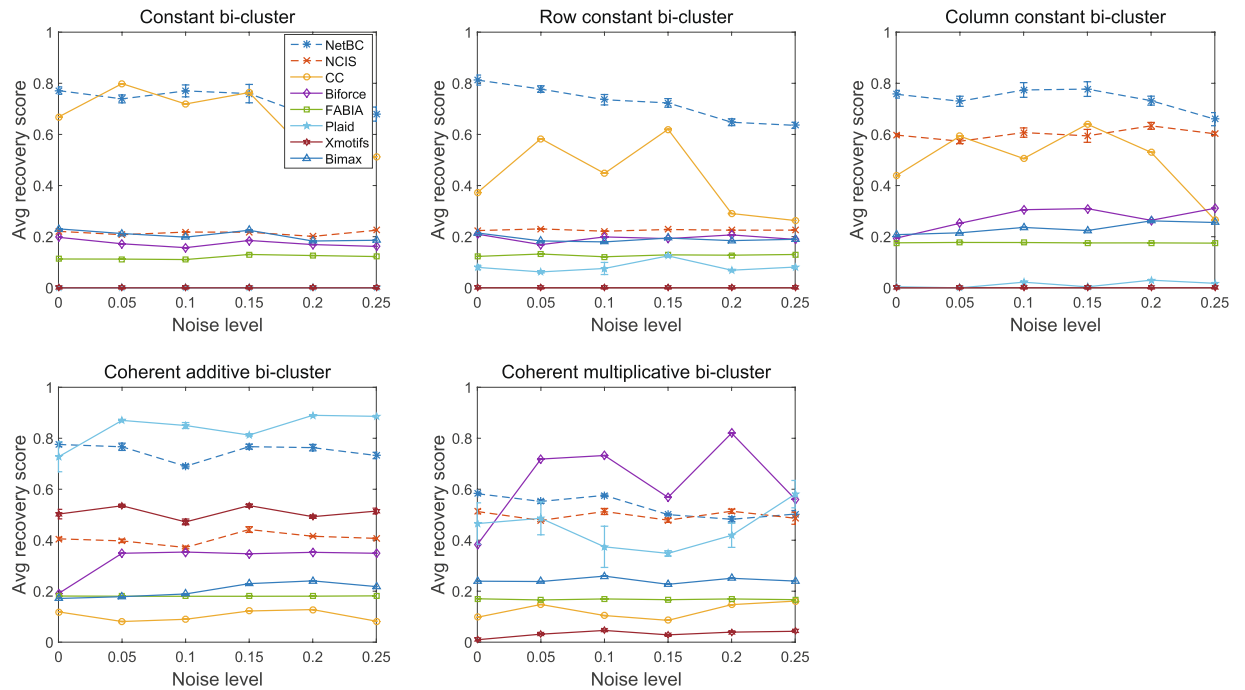
From Figs 6 and 7, we can see that, even without incorporating gene interaction networks, NetBC still achieves better performance than most bi-clustering methods on the constant, row constant, column constant, coherent additive bi-clusters. CC achieves relatively better performance on constant, row constant and column constant bi-clusters. Plaid achieves better performance on coherent additive bi-clusters than other methods. xMOTIFs is skilled in extracting additive bi-clusters. Bimax is good at extracting column constant bi-clusters. On the coherent multiplicative bi-clusters, Biforce performs better than NetBC. Furthermore, we can see that NetBC outperforms NCIS on all these five types of bi-clusters and NetBC is generally robust to noise.

To further assess the performance of NetBC and NCIS, we compared their performance on separating samples in Leukemia cancer gene expression data with simulated noisy genes. Leukemia contains 38 samples and 4412 genes. The gene interaction network (obtained from BioGrid, HPRD and STRING) of Leukemia contains 162987 edges. We assign weights to genes according to the variation of gene expression profiles across samples and gene interaction network. Next, we choose genes with lowest weight as 'uninformative' genes and tag them as noisy genes. We permute the expression levels of these noisy genes with random numerics between the maximum and





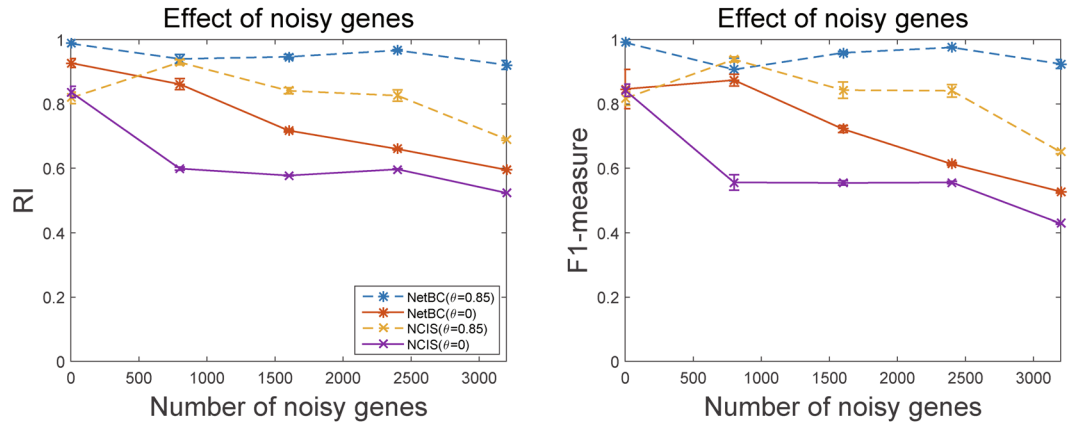
**Figure 6.** Relevance of bi-clustering methods on five types of bi-clusters.



**Figure 7.** Recovery of bi-clustering methods on five types of bi-clusters.

minimum value of the Leukemia gene expression data matrix. Figure 8 reports the results of NetBC and NCIS under different number of noisy genes.

From this Figure, we can see that NetBC almost always outperforms NCIS, but the performance of NetBC and NCIS with  $\theta = 0$ , and of NCIS with  $\theta = 0.85$  downgrades as the number of noisy genes increase. The reason is that noisy genes not only mislead the variance of gene expression profiles, but also successively mislead the weights assigned to genes. We can find that NetBC and NCIS with  $\theta = 0.85$  generally have better performance than with  $\theta = 0$ , and NetBC with  $\theta = 0.85$  is much less affected by noisy genes than NCIS with  $\theta = 0.85$ . These experimental results demonstrate that incorporating gene interaction network to assign weight to genes can improve the



**Figure 8.** Performance of NetBC and NCIS under different number of noisy genes.

performance of bi-clustering than assigning weights to genes based on the variation of gene expression profiles alone, and also show that NetBC can more effectively incorporating gene interaction network than NCIS.

We also explore the performance of NetBC and NCIS over random perturbations in the gene interaction network. Figure 9 reports the experimental results of NetBC and NCIS with  $\theta = 0.85$  on Leukemia dataset with randomly added, deleted and rewired edges between genes in the network. Here, NetBC and NCIS with  $\theta = 0$  are not considered, since  $\theta = 0$  means the network is excluded. By comparing the results in Fig. 8 with those in Fig. 9, we can see that, even with some fluctuations both NetBC and NCIS are relatively robust to perturbations of network. NetBC sometimes even obtains better results as more edges added or deleted. This observation is accountable, since the edges in the original network are not complete but with some noisy (missing) edges, and some randomly added (deleted or perturbed) edges just coincide with missing (or noisy) edges. We believe the performance of NetBC and NCIS can be further improved with the improved quality of gene interaction network.

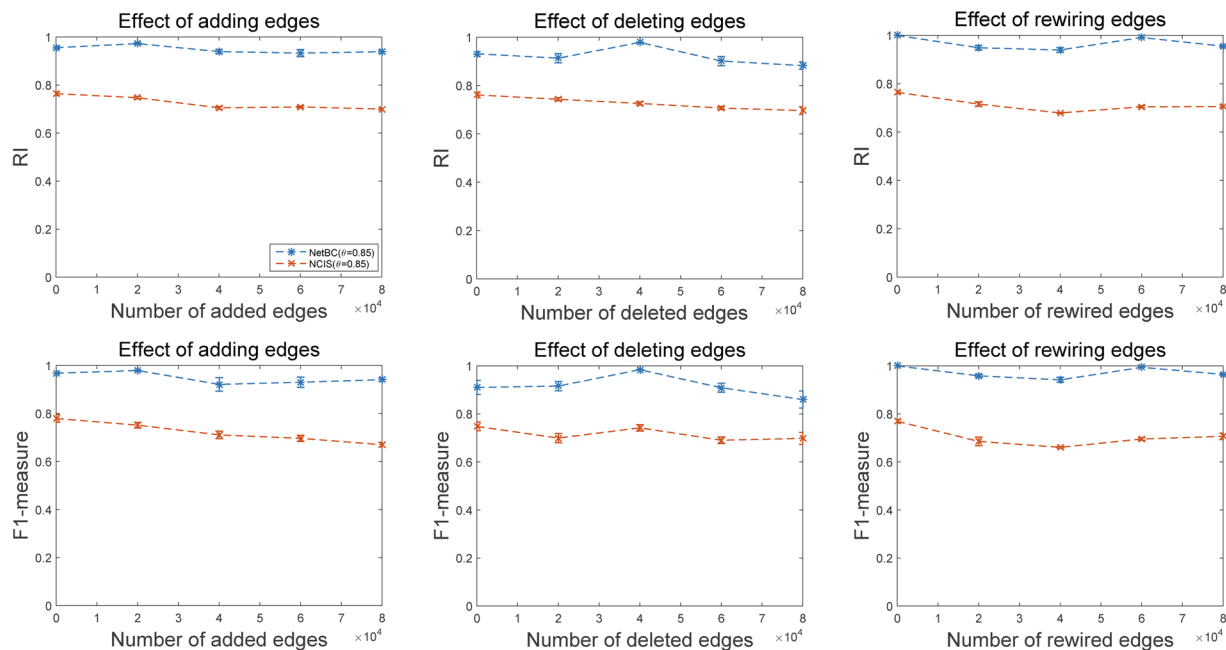
**Runtime analysis.** Bi-clustering is often involved with long runtime costs, especially when the gene expression data matrix is large. Therefore, it is highly challenging to develop an efficient bi-clustering algorithm. Suppose the number of iterations is  $T$ . In each iteration, NetBC takes  $O(k^2m + k^3 + kmn)$  to compute  $\mathbf{R}(\mathbf{R}^T\mathbf{R})^{-1}\mathbf{R}^T\mathbf{G}$  (see Method Section for meanings of these matrix symbols),  $O(n^2d + kmn + k^3)$  to compute  $\mathbf{W}\mathbf{X}_1\mathbf{X}_2^T$ ,  $O(k^2m + kmn + km^2)$  to compute  $\mathbf{X}_3^T\mathbf{W}\mathbf{X}_4$  and  $O(dmn + d^3 + dn^2)$  to compute  $\mathbf{G}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T$ . Since  $d \ll n, k \ll m$  and  $m$  is generally larger than  $n$ . The overall time complexity of NetBC is  $O(T((k + d)mn + km^2))$ . NCIS also works on the same size matrices, it has similar time complexity as NetBC.

Here, we record the runtime change of NetBC and of several widely used bi-clustering methods as the number of genes increases from 1000 to 5000 and fixed the number samples as 200, and report the recorded results in Fig. 10. All the comparing methods are implemented on a desktop computer (Windows 7, 8GB RAM, Intel(R) Core(TM) i5-4590). The parameters of these methods are sets as default values. From Fig. 10, we can see that Fabia runs faster than other comparing methods. The runtime cost of NetBC is slightly larger than MSSRCC, since NetBC additionally utilizes gene interaction networks to assign weights to genes. NetBC is slightly faster than NCIS and they both increase relative slowly with the increase number of genes. The runtime cost of BCC is the largest and increases sharply, since BCC assumes the genes (or samples) of gene expression data are generated by a finite mixture of underlying probability distributions and it takes much time to estimate these distributions. Similarly, Plaid assumes the values of bi-clusters can be explained by additive model, it also needs relatively larger runtime cost than other comparing methods. The runtime cost of Biforce is larger than Plaid. In summary, NetBC can hold comparable efficiency with state-of-the-art bi-clustering methods and it generally obtains better performance than related methods.

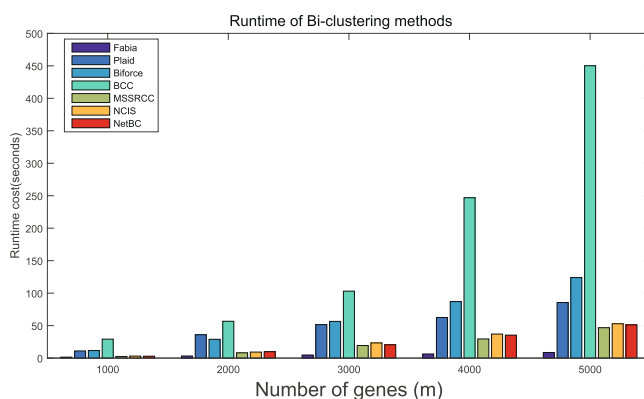
## Methods

Let  $\mathbf{G} \in \mathbb{R}^{m \times n}$  represent the gene expression data for  $m$  genes and  $n$  samples, the  $(i, j)$ -th entry of  $\mathbf{G}$  is given by  $g_{ij}$ . A bi-cluster is a subset of rows that exhibit similar behavior across a subset of columns, and it can be described as a sub-matrix of  $\mathbf{G}$ . Let  $\mathcal{I}$  represent a set of row indices for a row cluster and  $\mathcal{J}$  represent a set of column indices for a column cluster. A sub-matrix  $\mathbf{G}_{\mathcal{I}, \mathcal{J}}$  of  $\mathbf{G}$  determined by  $\mathcal{I}$  and  $\mathcal{J}$  encodes a bi-cluster.

**Objective.** The target of bi-clustering is to discover multiple bi-clusters from a gene expression data matrix, all the entries of a bi-cluster exhibit similar numerical values as much as possible. This target can be achieved by reordering the rows and columns of the matrix to group similar rows and similar columns together<sup>13</sup>. Traditional clustering algorithms group the similar genes (or samples) together by assigning genes (or samples) to their nearest cluster centroids. Ideally, entries are with constant value in the entire bi-cluster. For real world gene expression data, constant bi-clusters are usually distorted by noises. A common criterion to evaluate a bi-cluster is the sum of squared differences between each entry of a bi-cluster and the mean of that bi-cluster<sup>17</sup>. The squared difference between an entry  $g_{ij}$  and the mean of the corresponding bi-cluster is computed as below:



**Figure 9.** Performance of NetBC and NCIS with randomly added, deleted and rewired edges of gene interaction network.



**Figure 10.** Runtime costs of eight bi-clustering methods under different number of genes, the number of samples is fixed as 200.

$$h_{ij} = \left( g_{ij} - \tilde{g}_{\mathcal{I}\mathcal{J}} \right)^2 \tag{2}$$

where  $\tilde{g}_{\mathcal{I}\mathcal{J}} = \frac{1}{|\mathcal{I}| \cdot |\mathcal{J}|} \sum_{i \in \mathcal{I}, j \in \mathcal{J}} g_{ij}$  is the mean of all entries in the bi-cluster,  $|\mathcal{I}|$  is the cardinality of  $\mathcal{I}$ . Bi-clustering tries to find all combinations of  $\mathcal{I}$  and  $\mathcal{J}$ , each combination has the minimum  $\sum_{i \in \mathcal{I}, j \in \mathcal{J}} h_{ij}$ .

In fact, the criterion to evaluate a bi-clustering method depends on the types of bi-clusters that can be identified by that method<sup>15</sup>. The criterion defined in Eq. (2) aims at finding bi-clusters with constant values. However, researchers may be not only interested with bi-clusters with constant values, but also bi-clusters with clear trends (or patterns), which generally include five major patterns<sup>17</sup>: (i) with constant values in the entire bi-cluster; (ii) with constant values in rows; (iii) with constant values in columns; (iv) with additive coherent values; (v) with multiplicative coherent values. If  $g_{ij} = u$  ( $i \in \mathcal{I}, j \in \mathcal{J}$ ), then  $(\mathcal{I}, \mathcal{J})$  corresponds to a bi-cluster with constant value. If  $g_{ij} = u + \alpha_i$  (or  $g_{ij} = u\alpha_i$ ) with  $i \in \mathcal{I}, j \in \mathcal{J}$ , then  $(\mathcal{I}, \mathcal{J})$  corresponds to a bi-cluster with constant value in rows. Similarly, if  $g_{ij} = u + \beta_j$  (or  $g_{ij} = u\beta_j$ ), then  $(\mathcal{I}, \mathcal{J})$  corresponds to a bi-cluster with constant value in columns. If  $g_{ij} = u + \alpha_i + \beta_j$  (or  $g_{ij} = u \times \alpha_i \times \beta_j$ ), then  $(\mathcal{I}, \mathcal{J})$  corresponds to a bi-cluster with additive (or multiplicative) coherent values. These five patterns of bi-clusters are recognized to have biological significance. Figure 11 provides illustrative examples for these five patterns of bi-clusters.

1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.0	3.0	4.0	1.0	2.0	5.0	4.0	1.0	2.0	1.5	0.5
1.0	1.0	1.0	1.0	2.0	2.0	2.0	2.0	1.0	2.0	3.0	4.0	2.0	3.0	6.0	5.0	2.0	4.0	3.0	1.0
1.0	1.0	1.0	1.0	3.0	3.0	3.0	3.0	1.0	2.0	3.0	4.0	4.0	5.0	8.0	7.0	4.0	8.0	6.0	2.0
1.0	1.0	1.0	1.0	4.0	4.0	4.0	4.0	1.0	2.0	3.0	4.0	5.0	6.0	9.0	8.0	3.0	6.0	4.5	1.5
Constant bi-cluster				Constant rows bi-cluster				Constant columns bi-cluster				Coherent values additive bi-cluster				Coherent values multiplicative bi-cluster			

**Figure 11.** Example of the five types of bi-clusters.

To discover bi-clusters of these patterns, we use the squared-residue to quantify the difference between an entry  $g_{ij}$  of a bi-cluster  $(\mathcal{I}, \mathcal{J})$  and the row mean, column mean and bi-cluster mean of that bi-cluster. Here we adopt a formula suggested by Cheng *et al.*<sup>12</sup> as follows:

$$h_{ij} = \left( g_{ij} - \tilde{g}_{i\mathcal{J}} - \tilde{g}_{\mathcal{I}j} + \tilde{g}_{\mathcal{I}\mathcal{J}} \right)^2 \tag{3}$$

where  $\tilde{g}_{i\mathcal{J}} = \frac{1}{|\mathcal{J}|} \sum_{j \in \mathcal{J}} g_{ij}$  is the row mean,  $\tilde{g}_{\mathcal{I}j} = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} g_{ij}$  is the column mean,  $\tilde{g}_{\mathcal{I}\mathcal{J}} = \frac{1}{|\mathcal{I}| |\mathcal{J}|} \sum_{i \in \mathcal{I}, j \in \mathcal{J}} g_{ij}$  is the bi-cluster mean. The smaller the squared residue  $h_{ij}$ , the larger the coherence is.

Suppose  $\mathbf{G}$  is partitioned into  $k$  row (gene) clusters and  $d$  column (sample) clusters. We use  $\mathbf{R} \in \mathbb{R}^{m \times k}$  ( $\sum_{k'=1}^k \mathbf{R}_{i,k'} = 1$ ) as the indicative matrix for gene clusters, if  $\mathbf{R}_{ik'} = 1$ , gene  $i$  belongs to gene cluster  $k'$ . Similarly, we use matrix  $\mathbf{C} \in \mathbb{R}^{n \times d}$  ( $\sum_{d'=1}^d \mathbf{C}_{j,d'} = 1$ ) as the indicative matrix for sample clusters, if  $\mathbf{C}_{jd'} = 1$ , sample

$j$  belongs to sample cluster  $d'$ .  $\mathbf{R}$  and  $\mathbf{C}$  has the following form:

$$\mathbf{R} = \begin{matrix} & \begin{matrix} k \\ \left[ \begin{array}{cccc} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 1 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 1 \end{array} \right] \end{matrix} & \begin{matrix} m \\ \left. \vphantom{\begin{matrix} k \\ \left[ \begin{array}{cccc} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 1 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 1 \end{array} \right]} \right\} \end{matrix} \end{matrix} \mathbf{C} = \begin{matrix} & \begin{matrix} d \\ \left[ \begin{array}{cccc} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 1 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 1 \end{array} \right] \end{matrix} & \begin{matrix} n \\ \left. \vphantom{\begin{matrix} d \\ \left[ \begin{array}{cccc} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 1 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 1 \end{array} \right]} \right\} \end{matrix} \end{matrix}$$

Assume row-cluster  $k'$  ( $1 \leq k' \leq k$ ) has  $m_{k'}$  rows, and  $m_1 + m_2 + \dots + m_k = m$ . Since there are  $k$  row clusters, we use  $\mathcal{I}_{k'}$  to represent the row index set of  $k'$ -th row cluster. Then, we can obtain:

$$(\mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{G})_{ij} = \sum_{k'=1}^k \mathbf{R}_{ik'} (\mathbf{R}^T \mathbf{R})_{k'k'}^{-1} (\mathbf{R}^T \mathbf{G})_{k'j} = \sum_{k'=1}^k \mathbf{R}_{ik'} \frac{1}{m_{k'}} (\mathbf{R}^T \mathbf{G})_{k'j} = \tilde{g}_{\mathcal{I}j} \tag{4}$$

where  $m_{k'} = |\mathcal{I}_{k'}|$  and  $(\mathbf{R}^T \mathbf{G})_{k'j} = \sum_{i' \in \mathcal{I}_{k'}} \mathbf{G}_{i'j}$ .

Similarly, we can obtain  $\mathbf{G} \mathbf{C} (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T = \tilde{g}_{i\mathcal{J}}$ ,  $\mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{G} \mathbf{C} (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T = \tilde{g}_{\mathcal{I}\mathcal{J}}$ . To explore multiple bi-clusters, NetBC minimizes the sum-squared residue over all genes and samples. Based on the above analysis, we can measure the overall sum-squared residue of multiple bi-clusters discovered by NetBC using  $\mathbf{G}$ ,  $\mathbf{R}$  and  $\mathbf{C}$  as follow:

$$\Psi(\mathbf{R}, \mathbf{C}) = \|\mathbf{G} - \mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{G} - \mathbf{G} \mathbf{C} (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T + \mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{G} \mathbf{C} (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T\|^2$$

s.t.  $\mathbf{R}_{ik'} \geq 0$ ,  $\mathbf{C}_{jd'} \geq 0$ ,  $\sum_{k'=1}^k \mathbf{R}_{i,k'} = 1$ ,  $\sum_{d'=1}^d \mathbf{C}_{j,d'} = 1$ . (5)

where  $(\mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{G})_{ij}$  is the row mean,  $(\mathbf{G} \mathbf{C} (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T)_{ij}$  is the column mean, and  $(\mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{G} \mathbf{C} (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T)_{ij}$  is the bi-cluster mean of  $(i, j)$  entry of  $\mathbf{G}$  in its corresponding bi-cluster, respectively.

The above objective function of NetBC can also be reformulated as:

$$\Psi(\mathbf{R}, \mathbf{C}) = \|(\mathbf{I} - \mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T) \mathbf{G} (\mathbf{I} - \mathbf{C} (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T)\|^2 \tag{6}$$

where  $\mathbf{I} \in \mathbb{R}^{m \times m}$  is an identity matrix, since  $(\mathbf{I} - \mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T) \mathbf{G} = \mathbf{G} - \mathbf{R}(\mathbf{R}^T \mathbf{R})^{-1} \mathbf{R}^T \mathbf{G}$ .

**Assigning weights to genes.** Gene expression data usually has a large amount of genes but with a few samples. The similarity between samples turns to be isometric as the gene dimensionality increasing<sup>30</sup>. Usually, genes are selected based on the absolute deviation value of the gene expression profile among samples to reduce the gene dimension. Feature selection methods, like principle component analysis (PCA)<sup>30</sup>, are applied to select genes to reduce the gene dimension. But selecting a subset of genes may result in information loss on clustering gene expression data especially when the biological sense is usually not straight. For this reason, assigning weights to genes to indicate the importance of the gene is more reasonable<sup>38</sup>. NetBC assigns weights to genes by using both the gene expression profiles and gene interaction network. Genes, who regulate more genes in the gene

interaction network and show larger expression variations across samples than other genes, are viewed more important to identify cancer subtypes, and will be given larger weights.

NetBC uses the GeneRank algorithm<sup>51</sup> to assign weights to genes. Suppose interactions between  $m$  genes are encoded by  $\mathbf{P} \in \mathbb{R}^{m \times m}$ . If there is directed (or undirected) an interaction from gene  $i$  to gene  $j$ ,  $\mathbf{P}_{ij} = 1$  (or  $\mathbf{P}_{ij} = \mathbf{P}_{ji} = 1$ ); otherwise,  $\mathbf{P}_{ij} = 0$ . The importance of a gene depends on the quantity and importance of its interacting partners, a gene that interacts with more genes and shows larger variation of expression profiles across samples should be assigned with a larger weight. Based on this assumption, the weight is set as follows:

$$\mathbf{w}_i^t = (1 - \theta)\mathbf{e}_i + \theta \sum_{j=1}^m \frac{\mathbf{P}_{ij}\mathbf{w}_j^{t-1}}{\text{deg}_j}, 1 \leq i \leq m \tag{7}$$

where  $\mathbf{w}_i^t$  denotes the weight of gene  $i$  in the  $t$ -th iteration,  $\mathbf{e}_i$  is the absolute value of expression profiles change for gene  $i$  among all samples.  $\text{deg}_j = \sum_{i=1}^m \mathbf{P}_{ij}$  means the total number of genes that have interactions with gene  $j$ .  $\theta$  balances the weight from gene expression profiles and gene interaction network, it also enables isolated genes to be accessed and it is usually set as 0.85. Eq. (7) is guaranteed to convergence when  $0 \leq \theta \leq 1$ <sup>38</sup>. The final optimized weights for all genes in Eq. (7) can be computed as follows:

$$(\mathbf{I} - \theta\mathbf{PD}^{-1})\mathbf{w} = (1 - \theta)\mathbf{e} \tag{8}$$

where  $\mathbf{D} \in \mathbb{R}^{m \times m}$  is a diagonal matrix with the  $j$ -th diagonal element equal to  $\text{deg}_j$ . When  $0 < \theta < 1$ , the weights of the genes are assigned according to Eq. (8). When  $\theta = 0$ , the weights of genes are completely dependent on the deviation of gene expression profiles. When  $\theta = 1$ , the weights of genes are assigned only based on the interacting partners of genes in the interaction network.

**Optimization.** After assigned weights to genes, the objective function of NetBC can be rewritten as:

$$\Phi(\mathbf{R}, \mathbf{C}) = \|(\mathbf{I} - \mathbf{R}(\mathbf{R}^T\mathbf{R})^{-1}\mathbf{R}^T)\mathbf{G}(\mathbf{I} - \mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T)\|^2 \mathbf{W} \tag{9}$$

where  $\mathbf{W} \in \mathbb{R}^{n \times n}$  is diagonal matrix with  $\mathbf{W}_{ii} = \mathbf{w}_i$ .

To optimize Eq. (9), we can iteratively optimize row indicative matrix  $\mathbf{R}$  and column indicative matrix  $\mathbf{C}$  by alternatively fixing one of them as constant. Let  $\mathbf{X}_1 = \mathbf{G}(\mathbf{I} - \mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T)$ ,  $\mathbf{X}_2 = (\mathbf{R}^T\mathbf{R})^{-1}\mathbf{R}^T\mathbf{X}_1$ , then the objective function to optimize  $\mathbf{R}$  can be rewritten as follows:

$$\begin{aligned} Q(\mathbf{R}) &= \sum_{i=1}^m \|(\mathbf{X}_1)_i - (\mathbf{R}\mathbf{X}_2)_i\|^2 \mathbf{W}_{ii} \\ &= \text{tr}(\mathbf{X}_1^T\mathbf{W}\mathbf{X}_1 - 2\mathbf{X}_1^T\mathbf{W}\mathbf{R}\mathbf{X}_2 + \mathbf{X}_2^T\mathbf{R}^T\mathbf{W}\mathbf{R}\mathbf{X}_2) \\ \text{s.t.} \quad &\mathbf{R} \geq 0, \quad \sum_{k'=1}^k \mathbf{R}_{i,k'} = 1. \end{aligned} \tag{10}$$

Let  $\mathbf{L}_1 \in \mathbb{R}^{m \times k}$  be the Lagrangian multipliers for  $\mathbf{R}(\mathbf{R} \geq 0)$ , then the Lagrangian function for  $\mathbf{R}$  is:

$$L(\mathbf{R}, \mathbf{L}_1) = Q(\mathbf{R}) - \text{tr}(\mathbf{L}_1\mathbf{R}^T) \tag{11}$$

To solve  $\mathbf{R}$ , we let  $\frac{\partial L(\mathbf{R})}{\partial \mathbf{R}} = 0$ . Based on Karush Kuhn-Tucker conditions<sup>52</sup>, we can get

$$(-\mathbf{A}^+ + \mathbf{A}^- + \mathbf{R}\mathbf{B}^+ - \mathbf{R}\mathbf{B}^-)_{ij}\mathbf{R}_{ij} = 0 \tag{12}$$

where  $\mathbf{A} = \mathbf{W}\mathbf{X}_1\mathbf{X}_2^T$ ,  $\mathbf{B} = \mathbf{X}_2\mathbf{X}_2^T$ ,  $\mathbf{A}^+ = \frac{|\mathbf{A}| + \mathbf{A}}{2}$  and  $\mathbf{A}^- = \frac{|\mathbf{A}| - \mathbf{A}}{2}$ ,  $\mathbf{B}^+$  and  $\mathbf{B}^-$  are similarly defined as  $\mathbf{A}^+$  and  $\mathbf{A}^-$ . Then we can obtain the optimal  $\mathbf{R}$  as follows:

$$\mathbf{R}_{ik'} = \mathbf{R}_{ik'} \sqrt{\frac{(\mathbf{A}^+ + \mathbf{W}\mathbf{R}\mathbf{B}^-)_{ik'}}{(\mathbf{A}^- + \mathbf{W}\mathbf{R}\mathbf{B}^+)_{ik'}}} \tag{13}$$

After that we can fix  $\mathbf{R}$  to update the column indicative matrix  $\mathbf{C}$ . Similarly, we set  $\mathbf{X}_3 = (\mathbf{I} - \mathbf{R}(\mathbf{R}^T\mathbf{R})^{-1}\mathbf{R}^T)\mathbf{G}$ ,  $\mathbf{X}_4 = \mathbf{X}_3\mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}$ . Then the objective function to obtain optimal  $\mathbf{C}$  is:

$$\begin{aligned} Q(\mathbf{C}) &= \sum_{i=1}^m \|(\mathbf{X}_3)_i - (\mathbf{X}_4\mathbf{C}^T)_i\|^2 \mathbf{W}_{ii} \\ &= \text{tr}(\mathbf{X}_3^T\mathbf{W}\mathbf{X}_3 - 2\mathbf{X}_3^T\mathbf{W}\mathbf{X}_4\mathbf{C}^T + \mathbf{C}\mathbf{X}_4^T\mathbf{W}\mathbf{X}_4\mathbf{C}^T) \end{aligned} \tag{14}$$

Let  $\mathbf{L}_2 \in \mathbb{R}^{n \times d}$  be the Lagrangian multiplier for  $\mathbf{C}$ , then the Lagrangian function for  $\mathbf{C}$  is as below:

$$L(\mathbf{C}, \mathbf{L}_2) = Q(\mathbf{C}) - \text{tr}(\mathbf{L}_2\mathbf{C}^T) \tag{15}$$

Similarly, we can obtain the optimal formula of  $\mathbf{C}$ .

$$C_{jd'} = C_{jd'} \sqrt{\frac{(\mathbf{D}^+ + \mathbf{CF}^-)_{jd'}}{(\mathbf{D}^- + \mathbf{CF}^+)_{jd'}}}$$
(16)

where  $\mathbf{D} = \mathbf{X}_3^T \mathbf{W} \mathbf{X}_4$ ,  $\mathbf{F} = \mathbf{X}_4^T \mathbf{W} \mathbf{X}_4$ . The optimal  $\mathbf{R}$  and  $\mathbf{C}$  can be iteratively optimized via Eq. (13) and Eq. (16) until  $\Phi(\mathbf{R}, \mathbf{C})$  convergency.

**Data availability.** The Matlab codes of NetBC can be accessed from <http://mlda.swu.edu.cn/codes.php?name=NetBC>.

## References

1. Brazma, A. & Vilo, J. Gene expression data analysis. *FEBS Letters* **480**, 17–24 (2000).
2. Kallioniemi, O. P., Wagner, U., Kononen, J. & Sauter, G. Tissue microarray technology for high-throughput molecular profiling of cancer. *Human Molecular Genetics* **10**, 657–662 (2001).
3. Ben-Dor, A., Friedman, N. & Yakhini, Z. Class discovery in gene expression data. Proceedings of the 5th Annual International Conference on Computational Biology, 31–38 (2001).
4. D'haeseleer, P. How does gene expression clustering work? *Nature Biotechnology* **23**, 1499–1502 (2005).
5. Perou, C. M., Sørlie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Rees, C. A. & Fluge, Ø. Molecular portraits of human breast tumours. *Nature* **406**, 747–752 (2000).
6. Sørlie, T., Perou, C. M., Tibshirani, R., Aas, T., Geisler, S., Johnsen, H. & Thorsen, T. Gene expression patterns of breast carcinomas distinguishing tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences* **98**, 10869–10874 (2001).
7. Tavazoie, S., Hughes, J. D., Campbell, M. J., Cho, R. J. & Church, G. M. Systematic determination of genetic network architecture. *Nature Genetics* **22**, 281–285 (1999).
8. Eisen, M. B., Spellman, P. T., Brown, P. O. & Botstein, D. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences* **95**, 14863–14868 (1998).
9. Vesanto, J. & Alhoniemi, E. Clustering of the self-organizing map. *IEEE Transactions on Neural Networks* **11**, 586–600 (2000).
10. Domeniconi, C., Gunopulos, D., Ma, S., Yan, B., Al-Razgan, M. & Papadopoulos, D. Locally adaptive metrics for clustering high dimensional data. *Data Mining and Knowledge Discovery* **14**, 63–97 (2007).
11. Ben-Dor, A., Chor, B., Karp, R. & Yakhini, Z. Discovering local structure in gene expression data: the order-preserving submatrix problem. *Journal of Computational Biology* **10**, 373–384 (2003).
12. Cheng, Y. & Church, G. M. Biclustering of expression data. Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology, 93–103 (2000).
13. Hartigan, J. A. Direct clustering of a data matrix. *Journal of the American Statistical Association* **267**, 123–129 (1972).
14. Prelić, B. S. & Zimmermann, P. A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics* **22**, 1122–1129 (2006).
15. Madeira, S. C. & Oliveira, A. L. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **1**, 24–25 (2004).
16. Veroneze, R., Banerjee, A. & Von Zuben, F. J. Enumerating all maximal biclusters in numerical datasets. *Information Sciences* **379**, 288–309 (2017).
17. Tanay, A., Sharan, R. & Shamir, R. Biclustering algorithms: A survey. *Handbook of Computational Molecular Biology* **9**, 122–124 (2005).
18. Bergmann, S., Ihmels, J. & Barkai, N. Iterative signature algorithm for the analysis of large-scale gene expression data. *Physical Review E* **67**, 031902 (2003).
19. Denitto, M., Farinelli, A. & Bicego, M. Biclustering gene expressions using factor graphs and the max-sum algorithm. Proceedings of the 24th International Joint Conference on Artificial Intelligence, 925–931 (2015).
20. Kluger, Y., Basri, R., Chang, J. T. & Gerstein, M. Spectral biclustering of microarray data: coclustering genes and conditions. *Genome Research* **13**, 703–716 (2003).
21. Dhillon, I. S., Mallela, S. & Modha, D. S. Information-theoretic co-clustering. Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 89–98 (2003).
22. Shan, H. & Banerjee, A. Bayesian co-clustering. Proceedings of the 8th IEEE International Conference on Data Mining, 530–539 (2008).
23. Carmona-Saez, P., Pascual-Marqui, R. D., Tirado, F., Carazo, J. M. & Pascual-Montano, A. Biclustering of gene expression data by non-smooth non-negative matrix factorization. *BMC Bioinformatics* **7**, 1 (2006).
24. Sun, P., Speicher, N. K., Röttger, R., Guo, J. & Baumbach, J. Bi-Force: large-scale bicluster editing and its application to gene expression data biclustering. *Nucleic Acids Research* **42**, e78 (2014).
25. Shi, J. & Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**, 888–905 (2000).
26. Murali, T. & Kasif, S. Murali, T. and Kasif, S. Extracting conserved gene expression motifs from gene expression data. *Pacific Symposium on Biocomputing* **8**, 77–88 (2003).
27. Hochreiter, S., Bodenhofer, U. & Heusel, M. FABIA: factor analysis for bicluster acquisition. *Bioinformatics* **26**, 1520–1527 (2010).
28. Lazzeroni, L. & Owen, A. *et al.* Lazzeroni, L. & Owen, A. Plaid models for gene expression data. *Statistica Sinica* **12**, 61–86 (2002).
29. Cho, H. & Dhillon, I. S. Coclustering of human cancer microarrays using minimum sum-squared residue coclustering. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **5**, 385–400 (2008).
30. Steinbach, M., Ertöz, L. & Kumar, V. The challenges of clustering high dimensional data. In: *New Directions in Statistical Physics* **273**, 273–309 (2004).
31. Jiang, D., Tang, C. & Zhang, A. Cluster analysis for gene expression data: a survey. *IEEE Transactions on Knowledge and Data Engineering* **16**, 1370–1386 (2004).
32. Shim, J. E. & Lee, I. Network-assisted approaches for human disease research. *Animal Cells and Systems* **19**, 231–235 (2015).
33. Barabási, A. L., Gulbahce, N. & Loscalzo, J. Network medicine: a network-based approach to human disease. *Nature Reviews Genetics* **12**, 56–68 (2011).
34. Chuang, H. Y., Lee, E., Liu, Y. T., Lee, D. & Ideker, T. Network-based classification of breast cancer metastasis. *Molecular Systems Biology* **3**, 140 (2007).
35. Hanisch, D., Zien, A., Zimmer, R. & Lengauer, T. Co-clustering of biological networks and gene expression data. *Bioinformatics* **18**, S145–S154 (2002).
36. Hofree, M., Shen, J. P., Carter, H., Gross, A. & Ideker, T. Network-based stratification of tumor mutations. *Nature Methods* **10**, 1108–1115 (2013).
37. Ding, C., Li, T., Peng, W. & Park, H. Orthogonal nonnegative matrix t-factorizations for clustering. Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 126–135 (2006).

38. Liu, Y., Gu, Q., Hou, J. P., Han, J. & Ma, J. A network-assisted co-clustering algorithm to discover cancer subtypes based on gene expression. *BMC Bioinformatics* **15**, 1 (2014).
39. Network, C. G. A. Comprehensive molecular portraits of human breast tumours. *Nature* **490**, 61–70 (2012).
40. Brunet, J. P., Tamayo, P. & Golub, T. R. *et al.* Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the National Academy of Sciences* **101**, 4164–4169 (2004).
41. Stark, C., Breitkreutz, B. J., Reguly, T., Boucher, L., Breitkreutz, A. & Tyers, M. BioGRID: a general repository for interaction datasets. *Nucleic Acids Research* **34**, D535–D539 (2006).
42. Prasad, T. K., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S., Mathivanan, S. & Balakrishnan, L. Human protein reference database 2009 update. *Nucleic Acids Research* **37**, D767–D772 (2009).
43. Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J. & Kuhn, M. STRING v10: protein-protein interaction networks, integrated over the tree of life. *Nucleic Acids Research* **43**, D447–D452 (2015).
44. Shaffer, J. P. Multiple hypothesis testing. *Annual Review of Psychology* **46**, 561–576 (1995).
45. Rand, W. M. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* **66**, 846–850 (1971).
46. Van Rijsbergen, C. J. Information retrieval. Butterworths, London (1979).
47. Wang, Z., Li, G., Robinson, R. W. & Huang, X. UniBic: Sequential row-based biclustering algorithm for analysis of gene expression data. *Scientific Reports* **6**, 23466 (2016).
48. Eren, K., Deveci, M., Kucuktunc, O. & Catalyurek, U. V. A comparative analysis of biclustering algorithms for gene expression data. *Briefings in Bioinformatics* **14**, 279–292 (2013).
49. Sun, P., Guo, J. & Baumbach, J. BiCluE-Exact and heuristic algorithms for weighted bi-cluster editing of biomedical data. *BMC Proceedings* **7**, S9 (2013).
50. Wold, S., Esbensen, K. & Geladi, P. Principal Component Analysis. *Chemometrics and Intelligent Laboratory Systems* **2**, 37–52 (1987).
51. Morrison, J. L., Breitling, R., Higham, D. J. & Gilbert, D. R. GeneRank: using search engine technology for the analysis of microarray experiments. *BMC Bioinformatics* **6**, 1 (2005).
52. Boyd, S., Vandenberghe, L. Convex optimization. *Cambridge University Press*, (2004).
53. Van't Veer, L. J., Dai, H. & Van De Vijver, M. J. Gene expression profiling predicts clinical outcome of breast cancer. *Nature* **415**, 530–536 (2002).
54. Tamayo, P., Scanfeld, D. & Ebert, B. L. Metagene projection for cross-platform, cross-species characterization of global transcriptional states. *Proceedings of the National Academy of Sciences* **104**, 5959–5964 (2007).
55. Jolly, R. A., Goldstein, K. M. & Wei, T. Pooling samples within microarray studies: a comparative analysis of rat liver transcription response to prototypical toxicants. *Physiological Genomics* **22**, 346–355 (2005).
56. Rosenwald, A., Wright, G. & Chan, W. C. The use of molecular profiling to predict survival after chemotherapy for diffuse large-B-cell lymphoma. *New England Journal of Medicine* **346**, 1937–1947 (2002).

## Acknowledgements

The authors are grateful to the reviewers' comments on significantly improving this paper. This work is supported by Natural Science Foundation of China (61402378), Natural Science Foundation of CQ CSTC (cstc2014jcyjA40031 and cstc2016jcyjA0351), Fundamental Research Funds for the Central Universities of China (XDJK2362015XK07 and XDJK2016B009).

## Author Contributions

G.-X.Y., X.-X.Y. and J.W. proposed the theoretical method, G.-X.Y. and J.W. designed the experiments and analyzed the results, X.-X.Y. implemented the experiments, G.-X.Y., X.-X.Y. and J.W. wrote and revised the manuscript.

## Additional Information

**Competing Interests:** The authors declare that they have no competing interests.

**Publisher's note:** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2017