Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 19(9): 8741–8759.
DOI: 10.3934/mbe.2022405
Received: 06 March 2022
Revised: 09 May 2022
Accepted: 11 May 2022
Published: 16 June 2022

Research article

Iterative bicluster-based Bayesian principal component analysis and least squares for missing-value imputation in microarray and RNA-sequencing data

Saskya Mary Soemartojo, Titin Siswantining*, Yoel Fernando, Devvi Sarwinda, Herley Shaori Al-Ash, Sarah Syarofina and Noval Saputra

Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Indonesia, Indonesia

* **Correspondence:** Email: titin@sci.ui.ac.id.

Abstract: Microarray and RNA-sequencing (RNA-seq) techniques each produce gene expression data that can be expressed as a matrix that often contains missing values. Thus, a process of missing-value imputation that uses coherence information of the dataset is necessary. Existing imputation methods, such as iterative bicluster-based least squares (bi-iLS), use biclustering to estimate the missing values because genes are only similar under correlative experimental conditions. Also, they use the row average to obtain a temporary complete matrix, but the use of the row average is considered to be a flaw. The row average cannot reflect the real structure of the dataset because the row average only uses the information of an individual row. Therefore, we propose the use of Bayesian principal component analysis (BPCA) to obtain the temporary complete matrix instead of using the row average in bi-iLS. This alteration produces new missing values imputation method called iterative bicluster-based Bayesian principal component analysis and least squares (bi-BPCA-iLS). Several experiments have been conducted on two-dimension independent gene expression datasets, which are microarray (e.g., cell-cycle expression dataset of yeast saccharomyces cerevisiae) and RNA-seq (gene expression data from schizosaccharomyces pombe) datasets. In the case of the microarray dataset, our proposed bi-BPCA-iLS method showed a significant overall improvement in the normalized root mean square error (NRMSE) values of 10.6% from the local least squares (LLS) and 0.6% from the bi-iLS. In the case of the RNA-seq dataset, our proposed bi-BPCA-iLS method showed an overall improvement in the NRMSE values of 8.2% from the LLS and 3.1% from the bi-iLS. The additional computational time of bi-BPCA-iLS is not significant compared to bi-iLS.

Keywords: biclustering; microarray; normalized root mean square error; RNA sequencing

1. Introduction

Molecular biology research on the molecular basis of biological activity requires data. Biologists acquire these data by using approaches and technologies such as microarray and RNA-sequencing (RNA-seq) techniques. Microarray technology is used to detect the sequences of nucleic acids and simultaneously thousands of gene transcripts from samples [1]. RNA-seq is a sequencing technique that can show the existence and amount of RNA in a biological sample by using next generation sequencing. Both techniques produce a high-dimensioanal gene expression data matrix with rows that indicate genes, columns that indicate experimental conditions, and cells that indicate the expression of that gene under those conditions. Gene expression data are very important for acquiring knowledge about cells, but there are frequently missing values. These missing values are often caused by experimental errors such as hybridization failures in microarray datasets and missing read counts in RNA-seq datasets. However, further analysis of these datasets requires a complete data matrix. Therefore, missing-value imputation approaches that use coherence the data are needed.

Two well-known missing-value imputation methods are LLS and BPCA. BPCA estimates missing values in the target gene (gene that contains missing values) by using a linear combination of principal components with parameters estimated using a Bayesian method. LLS uses a linear combination of the target gene and its similar genes to estimate the missing values in the target gene, and it uses clustering to measure gene similarities. In reality, genes are similar only under certain experimental conditions, so this similarity should only be measured by considering the related experimental conditions instead of all of the conditions. This is why clustering should be performed in rows and columns simultaneously, which is called biclustering [2]. Biclustering aims to identify local patterns in genes and conditions at the same time. The output of the biclutering technique is biclusters [3]. The use of this technique in LLS gives a better estimation of the missing values. Biclustering collates genes and conditions based on a weighted distance and correlation, respectively. Then, a regression model is used for least square-based missing-value estimation. An iterative framework is applied to improve the selection of coherent genes and correlated conditions. This method is called iterative bicluster-based least squares or bi-iLS [4].

Bi-iLS uses the row average to fill in all of the missing values in the target gene to obtain a temporary complete matrix. However, the row average is viewed as being flawed. The row average cannot reflect the real structure of the dataset because it only uses the information from an individual row. Thus, BPCA is considered better than the row average due to it reflecting the global covariance structure in all genes [5]. In this study, BPCA was used to obtain the temporary complete matrix in bi-iLS instead of the row average. This modification resulted in a new imputation method called bi-BPCA-iLS.

In this paper, the framework and implementation of our proposed bi-BPCA-iLS algorithm for missing-value imputation has been presented. The proposed missing-value imputation method will be implemented on a microarray dataset of *Saccharomyces cerevisiae* and an RNA-seq dataset of *Schizosaccharomyces pombe*.

2. Missing values

In theory, every data point has a probability of being missing. The process of setting this probability is called the missing-data mechanism or response mechanism, while the models of these processes are

called missing-data or response models [6–9]. Missing values can be categorized into three groups [10]. If the probability of a data point becoming missing is the same for all, then the missing values are called missing completely at random (MCAR). If the probability of a data point becoming missing is the same only for certain groups based on observational data, then the missing values are called missing at random (MAR). If missing values are neither MCAR nor MAR, then they are called missing not at random (MNAR) or not missing at random (NMAR). In other words, missing values in NMAR are independent of unobserved data [11].

3. Clustering and biclustering

Clustering is a technique that groups data points into several groups or clusters. In gene expression data, the purpose of clustering is to group genes into clusters where each cluster consists of genes that are similar to each other and dissimilar to genes from other clusters [12]. Biclustering in gene expression data is the simultaneous clustering of rows and columns [13]. The aim of biclustering is to find groups of similar genes based only on correlated experimental conditions. The output of biclustering is a bicluster. Genes are similar under certain experimental conditions, so biclustering is preferable to clustering. A comparison of biclustering and clustering in two-dimensional gene expression data matrices can be seen in Figure 1 [14]. Figure 1(a) indicates a clustering technique of genes based on all conditions, while Figure 1(b) shows a biclustering technique of genes based only on correlative experimental conditions.

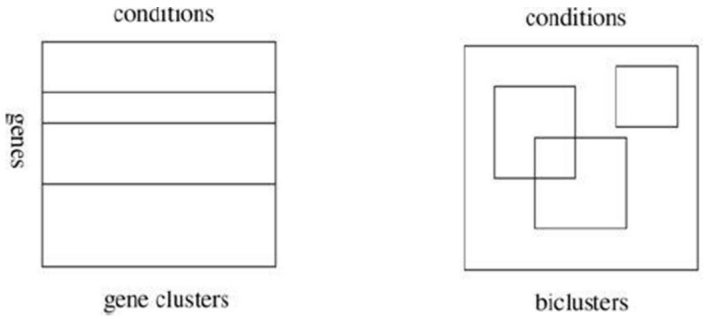

Figure 1. Comparisons of biclustering and clustering. Source: Tanay et al. [15].

4. Imputation method

4.1. LLS

LLS is a missing-value imputation method that identifies tne coherent information in gene expression data. There are two steps to the LLS method. The first step is to select k similar genes using Euclidean distance. The second step is to estimate the missing values [16, 17]. This neighbor-based imputation method suits datasets that have a structure with dominant local similarities and high complexity [18].

Let a matrix $\mathbf{E}$ be the expression matrix consisting of m genes and n conditions. Assuming that the

gene g_1 has k similar genes ($g_{s1}, g_{s2}, \ldots, g_{sk}$) given the Euclidean distance and p missing values in the first p conditions, then the target gene y can be defined.

$$\begin{pmatrix} g_s \\ g_{s1} \\ g_{s2} \\ . \\ . \\ . \\ g_{sk} \end{pmatrix} = \begin{pmatrix} \alpha & \mathbf{w} \\ \mathbf{B} & \mathbf{A} \end{pmatrix} = \begin{pmatrix} \alpha_1 & \alpha_2 & ... & \alpha_p & w_1 & w_2 & ... & w_{n-p} \\ B_{1,1} & B_{1,2} & ... & B_{1,p} & A_{1,1} & A_{1,2} & ... & A_{1,n-p} \\ B_{2,1} & B_{2,2} & ... & B_{2,p} & A_{2,1} & A_{2,2} & ... & A_{2,n-p} \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ B_{k1} & B_{k2} & ... & B_{kp} & A_{k1} & A_{k2} & ... & A_{k,n-p} \end{pmatrix},$$

where α is a vector of $1 \times p$ consisting of p missing values, w is a vector of $1 \times (n-p)$ consisting of the non-missing values in the target gene and the matrices B and A are the k similar genes' corresponding columns with α and w, respectively. Vector $\mathbf{X}$ can be defined as the solution to the least squares problem with A^T and w.

$$||A^T x - w^T||.$$

The solution of this least squares problem is

$$\hat{x} = (AA^T)^{-1}Aw^T = (A^T)^+ w^T,$$

where A^+ is the pseudoinverse of the matrix $\mathbf{A}$. Hence, the missing values in the target gene g_1 can be estimated using

$$\hat{a} = B^T \hat{x} = B^T (A^T)^+ w.$$

To choose the proper value of k, LLS uses a heuristic algorithm by applying artificial missing values to genes. These artificial missing values will be estimated using different values of k, then the value of k that produces the lowest estimation error will be chosen as the proper value of k [16].

4.2. bi-iLS

Bi-iLS is updated from the imputation method called LLS [16] in two aspects, i.e., the use of biclustering and an iterative framework. Bi-iLS can recognize gene similarities only under certain correlative conditions (biclustering), while LLS takes account all of the conditions in a data matrix. This makes bi-iLS preferable to LLS for gene expression data [4]. This imputation method suits data that have a dominant local similarity structure [18]. There are two parameters that need to be defined in the early stage of this process, namely k (for k similar genes) and T_0.

Let the matrix $\mathbf{E}$ be the expression matrix consisting of m genes and n conditions. A gene that has p missing values is called the target gene. Assuming that all p missing values are in the first p conditions without a loss of generality, the target gene is defined as

$$g_t^T = \begin{pmatrix} \alpha & w \end{pmatrix},$$

where α is a vector of $1 \times p$ comprising p missing values and w is a vector of $1 \times (n-p)$ consisting of the non-missing values in the target gene. Similar to LLS, the first step of bi-iLS is to select k similar genes of target genes by using the Euclidean distance. The measurement of Euclidean distance requires a complete matrix, so bi-iLS uses the row average to fill n all of the missing values and obtain the temporary complete matrix. After selecting k similar genes, they are defined as

$$\begin{pmatrix} g_{s1}^T \\ \dots \\ g_{sk}^T \end{pmatrix} = \begin{pmatrix} B & A \end{pmatrix},$$

where $g_(s_1)^T$ denotes k similar genes, while the matrices **B** and **A** denote, respectively, the expression values for the first p conditions and remaining $(n - p)$ conditions of the selected similar genes. Every condition has a different correlation with the other conditions. So, to account for the correlation or weight of each condition in the identification of the missing values, matrix **R** is defined as

$$R = B^T A.$$

Matrix **R**, with the size of $p \times (n - p)$, represents the weighted correlations between other conditions and the condition where the missing values in the target gene are found. The (j, v)th element of **R** is denoted by $r_j(v)$. The larger the value of $r_j(v)$, the larger are the weights and stronger are the correlations between the conditions with the missing values. Then, using **R**, k similar genes are reselected. Reselection of the k similar genes uses the weighted Euclidean distance of the target gene g_t and other genes g_s based on the location of the jth missing values. The equation is

$$d_j(g_t, g_s) = \frac{\sqrt{\sum_{v=p+1}^{n} r_j(v-p)^2 [g_t(v) - g_s(v)]^2}}{\sqrt{\sum_{v=p+1}^{n} r_j(v)^2}},$$

where $g(v)$ denotes the vth element of g_t or g_s. Then, upon estimating the jth missing values for the target gene, conditions that are uncorrelated are removed from the least squares framework. Let

$$r_{j,max} = \max_{v \in 1,\dots,n-p} |r_j(v)|;$$

then the conditions are said to be related if

$$|r_j(v)| \geq T_0 \cdot r_{j,max}.$$

where T_0 is a pre-defined parameter using the same heuristic algorithm to find the proper value of k. The removal of uncorrelated conditions redefines matrices **A** and **B** and w. Hence, we have

$$g_t^T = \begin{pmatrix} \alpha_j & w_j \end{pmatrix},$$

where α_j denotes the jth missing values and w_j denotes the non-missing values of correlated conditions. Also, we have

$$\begin{pmatrix} g_{s1}^T \\ \dots \\ g_{sk}^T \end{pmatrix} = \begin{pmatrix} B_j & A_j \end{pmatrix},$$

where B_j represents the jth columns of the data and A_j denotes a matrix consisting of the correlated columns of the k similar genes. Similar to LLS, a regression model $\alpha_j = B_j^T x_j$ is needed to estimate the jth missing value where x_j contains the regression coefficient for k similar genes. x_j can be obtained by minimizing the least squares error, as follows:

$$||A^T x - w^T||.$$

Thus, the jth missing value in the target gene can be estimated by using

$$\alpha_j = B_j^T \hat{x}_j = B_j^T (A^T)_j^+ w_j^T,$$

where $(A^T)_j^+$ is the pseudoinverse of A_j^T.

An iterative framework is applied to improve the selection of similar genes. A complete matrix output from the ith iteration will be the temporary complete matrix in the $(i + 1)$th iteration. This iteration process will be repeated until it reaches the maximum iteration or a specific criterion. The complete framework of bi-iLS can be seen in Figure 2.

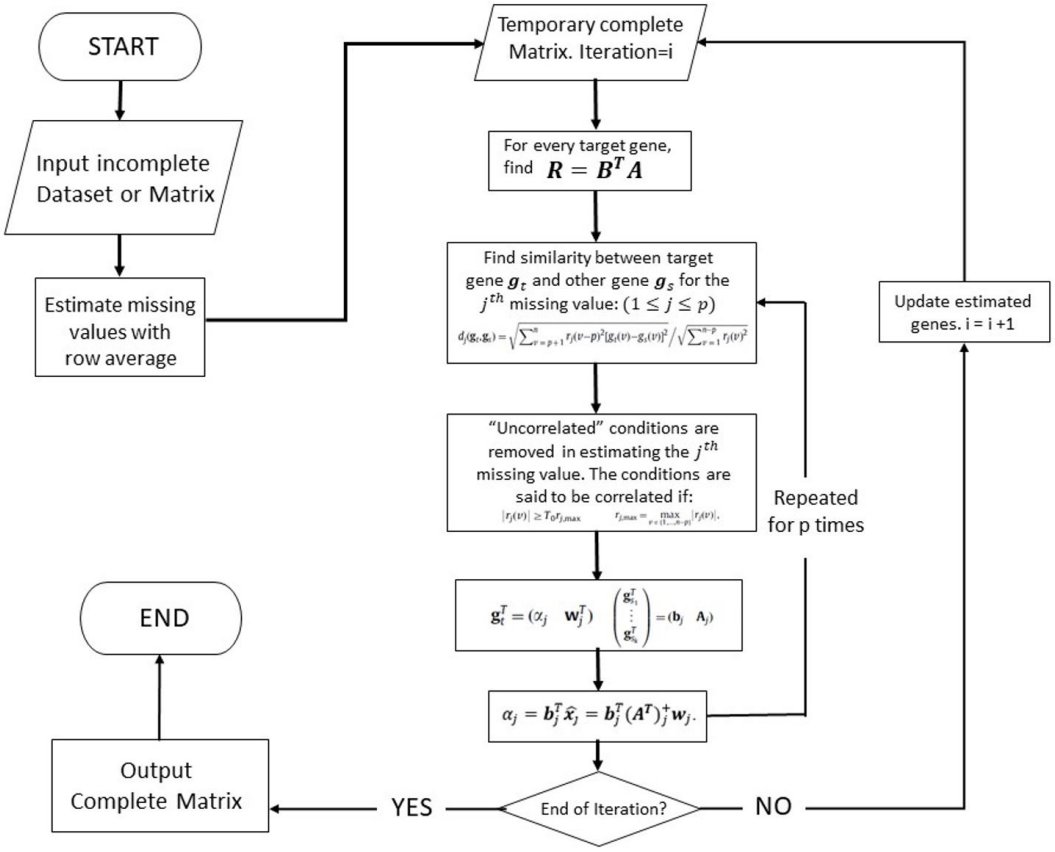

Figure 2. Complete framework of bi-iLS algorithm.

4.3. BPCA

There are three main steps of the BPCA based imputation method: principal component (PC) regression, Bayesian estimation and application of an expectation-maximization (EM) repetitive algorithm [19]. In PC regression, Principal Component Analysis (PCA) represents the D-dimensional vector y as a linear combination of K principal axis vectors w_l $(1 \le l \le K$ and $K < D)$, as follows:

$$y = \sum_{l=1}^{K} x_l w_l + \epsilon,$$

where D is the quantity of columns in data, x_l is a factor score and $\in$ is the residual error. Assuming that there are no missing values, PCA can find $w_l = \sqrt{\lambda_l} u_l$ where λ_l and u_l respectively denote the eigenvalues and eigenvectors of the corresponding covariance matrix of y. If missing values are present,

then the principal axis vectors are split into two parts, i.e., $W = (W^{obs}, W^{miss})$ where W^{obs} and W^{miss} denote a matrix that has column vectors $w_1^{obs}, \ldots, w_K^{obs}$ and $w_1^{miss}, \ldots, w_K^{miss}$, respectively. Factor scores $x = (x_1, \ldots, x_K)$ are obtained by minimizing the residual error of the observed part as follows:

$$||y^{obs} - W^{obs}x||^2.$$

This is a simple least squares problem that can be solved easily. Hence, the missing part of y can be estimated as

$$y^{miss} = W^{miss}x.$$

However, these parameters are still unknown. BPCA uses a probabilistic PCA model under the assumption that the residual error $\in$ and x_l $(1 \leq l \leq K)$ obey normal distributions. The parameters W, μ and τ form a parameter set $\theta \equiv \{W, \mu, \tau\}$. BPCA uses Bayesian estimation to estimate these parameters. It is used here because it can locate the best dimensions for latent space. This estimation is done by applying the EM algorithm until convergence is reached. This imputation method is appropriate for data with lower complexity structures [20].

5. Proposed imputation method

The proposed bi-BPCA-iLS algorithm updates the bi-iLS algorithm during the process of obtaining the temporary complete matrix. Other than the process of obtaining the temporary complete matrix, bi-BPCA-iLS and bi-iLS are the same. In bi-ILS, the row average is used to fill in all of the missing values for the target genes to obtain a temporary complete matrix. However, the use of the row average to fill in the missing values is considered unsatisfactory. Row averages cannot reflect the structure of the data because they only use the information of a single row or gene [21]. Also, use of the row average is not an effective approach when there is an outlier in the target gene. Hence, the use of BPCA to get a temporary complete matrix is thought to be better than the use of the row average. BPCA can reflect the global covariance structure of all genes [5]. The main idea behind the proposed bi-BPCA-iLS method is to use BPCA instead of the row average to get a temporary complete matrix in the bi-iLS framework. This alteration means that bi-BPCA-iLS becomes an updated and improved missing-value imputation method. As mentioned before, bi-iLS matched to data that have a dominant local similarity structure and high complexity, while BPCA suits data with a structure of lower complexity. The idea of combining BPCA with bi-iLS makes bi-BPCA-iLS become more robust for data with a lower complexity structure. The complete framework of bi-BPCA-iLS can be seen in Figure 3. The differences table for the LLS, bi-iLS and bi-BPCA-iLS methods is given as Table 1.

Table 1. Differences between LLS, bi-iLS and bi-BPCA-iLS.

	LLS	Bi-iLS	Bi-BPCA-iLS
Gene similarity	Clustering	Biclustering	Biclustering
Temporary complete matrix	Row-average	Row-average	BPCA
Parameters	k	k and T_0	k and T_0
Process of iteration	No	Yes	Yes
Authors	Kim et al. [16]	Cheng et al. [4]	Newly Proposed

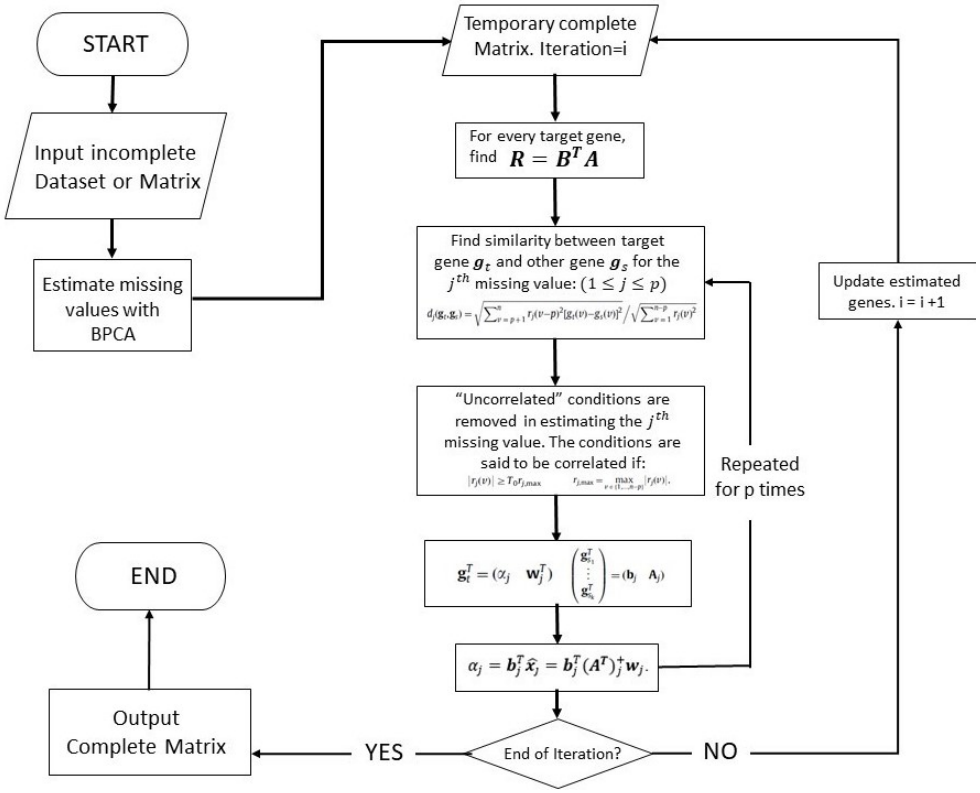

Figure 3. Complete framework of proposed algorithm, bi-BPCA-iLS.

Table 1 shows the differences between the three least squares-based imputation algorithms. LLS uses clustering to measure gene similarity, while bi-iLS and bi-BPCA-iLS use biclustering, which, as mentioned before, is considered to have higher efficacy. The row average is used in LLS and bi-iLS to obtain the temporary complete matrix, while bi-BPCA-iLS uses BPCA. Only bi-iLS and bi-BPCA-iLS iterates the imputation process. Our proposed imputation algorithm is the newest among these.

6. Experiments

The proposed method has been implemented and evaluated on two-dimensional gene expressions: a microarray dataset and an RNA-seq dataset [22]. Bi-iLS was proven to perform well on the microarray datasets of Spellman 1998 for *Saccharomyces cerevisiae* [4], so bi-BPCA-iLS was also implemented on this dataset to make a performance comparison. Also, both bi-BPCA-iLS and bi-iLS were implemented on RNA-seq to analyze their performances on different gene expression datasets.

6.1. Microarray Data

The microarray dataset is a cell cycle expression dataset for the yeast *Saccharomyces cerevisiae*; it has been synchronized using a CDC15 temperature-sensitive mutant [23]. According to Spellman et al., the samples of mRNA were taken every 10 minutes for 300 minutes. However, there were several missing time points in the published data. In fact, samples were taken every 20 minutes from 10 min to 70 min, and then every 10 minutes from 70 min to 250 min and every 20 minutes from 250 min to 290

min. Therefore, the CDC15 dataset contains the expression level of 6178 genes at 24 different time points which gives a matrix size of 6178×24. An example of the CDC15 dataset is shown in Table 2.

Table 2. Example of CDC15 dataset.

	10 min	30 min	50 min	70 min	...	290 min
Gene 1	−0.16	0.09	−0.23	0.03	...	−0.26
Gene 2	NaN	NaN	NaN	−0.58	...	NaN
Gene 3	−0.37	−0.22	-0.16	0.04	...	−0.41
Gene 4	NaN	NaN	NaN	−1.5	...	NaN
Gene 5	−0.43	−1.33	−1.53	−1.53	...	1.18

The CDC15 dataset had missing values, so genes that contained missing values were removed to get the ground truth. The ground truth was used to calculate the estimation error or NRMSE of each imputation methods. After removing genes that contained missing values, the size of the matrix became 4381×24. In the experiments for this dataset, r% of the observation values was set to be missing randomly where r = 1, 5, 10, 15, 20, 25 and 30. The estimation was repeated five times for each missing rate to generate the average result.

6.2. RNA-seq data

The RNA-seq dataset was gene expression data from the *Schizosaccharomyces pombe* or GSE150544 [24]. The technique of RNA sequencing was used to identify the differences between the gene expression levels of four different INO80 mutant strains, each with two replicates; this resulted in eight samples for each gene. The four strains were *wt* (control), *Nht*1, *Iec*1 and *Iec*5. The length for each gene, which indicates how many nucleotides are in that gene, was also included. In this experiment, only coding genes were observed. This dataset contained the expression of 5137 genes under nine different conditions , i.e., the length, *wt_rep*1, *wt_rep*2, *nht*1_*rep*1, *nht*1_*rep*2, *Iec*1_*rep*1, *Iec*2_*rep*2, *Iec*5_*rep*1 and *Iec*5_*rep*2, resulting in a matrix size of 5137×9. The data were not normalized to ensure the real expression of each gene and positive gene expression. An example of the GSE150544 dataset is shown in Table 3.

Table 3. Example of GSE150544 dataset.

	Length	wt_rep1	wt_rep2	nht1_rep1	nht1_rep2	Iec1_rep1	Iec1_rep2	Iec5_rep1	Iec5_rep2
Gene 1	669	18	16	8	2	4	15	17	19
Gene 2	993	46	50	45	25	33	34	25	29
Gene 3	3227	1623	1474	1655	1268	994	1870	1476	1849
Gene 4	868	258	322	215	200	138	284	278	286
Gene 5	2250	87	79	119	121	87	209	88	102
...	...	...	...	...	...	...	...	...	...
Gene 5137	546	0	0	0	0	0	0	0	0

A value of zero indicates that a gene was not detected because the gene was not expressed , or was minimally expressed; therefore the value of zero is not a missing value. Then, r% of the observation values was set to be missing randomly where r = 1, 5, 10, 15, 20, 25 and 30. The estimation was

repeated five times for each missing rate to generate the average result.

6.3. Imputation results

Our proposed imputation method was implemented in MATLAB. The parameters k and T_0 were estimated automatically using the integrated function in our algorithm. The estimation process was iterated five times for each test. We carried out five tests for each missing rate to obtain the most accurate and convergent results. The imputation results applying our proposed method to the microarray dataset can be seen in Table 4 and Figure 4 below, where *mr* denotes the missing rate in Tables 4 and 5.

Table 4. Imputation results of applying bi-BPCA-iLS to the CDC15 dataset.

NRMSE Bi-BPCA-iLS	mr 1%	mr 5%	mr 10%	mr 15%	mr 20%	mr 25%	mr 30%
Test 1	0.1851	0.3934	0.4766	0.5485	0.5681	0.6102	0.6369
Test 2	0.1685	0.4610	0.5101	0.5634	0.5717	0.6093	0.6269
Test 3	0.2065	0.3882	0.4888	0.5359	0.5887	0.6044	0.6206
Test 4	0.2170	0.3741	0.4892	0.5476	0.5839	0.6034	0.6300
Test 5	0.2371	0.3934	0.4811	0.5458	0.5801	0.6000	0.6203
Average	0.20284	0.40202	0.48916	0.54824	0.5785	0.60546	0.62694

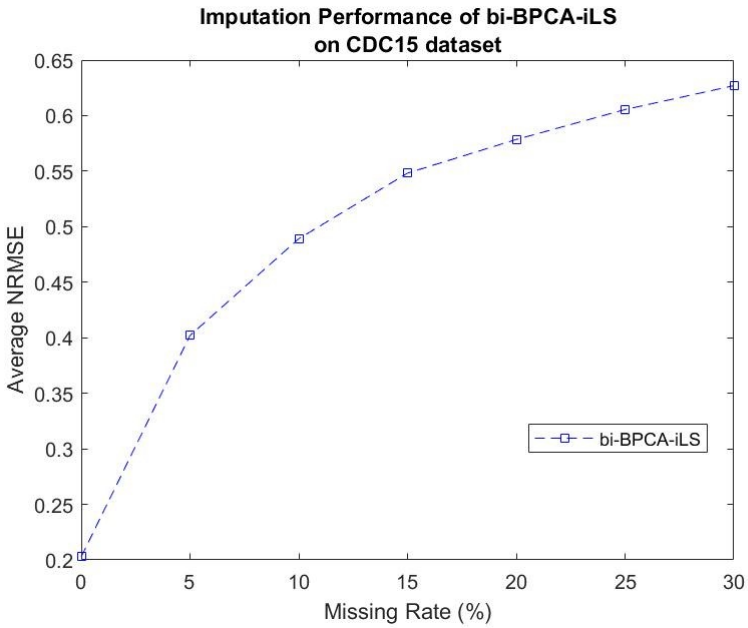

Figure 4. Imputation performance of bi-BPCA-iLS on CDC15 dataset.

The imputation results of applying our proposed method to the RNA-seq dataset can be seen in Table 5 and Figure 5 below.

Based on Table 5 and Figure 5, the average value of the NRMSE for a missing rate of 1% was 0.29626, for a missing rate of 5% was 0.23798 and for a missing rate of 10% was 0.24662. The lowest

estimation error was achieved when the missing rate was 5%; it was highest when the missing rate was 1%. The NRMSE values were predominantly below 0.3 at every missing rate, indicating that our proposed imputation method, bi-BPCA-iLS, performed well on the GSE150544 dataset.

Table 5. Imputation results of applying bi-BPCA-iLS to the GSE150544 dataset.

NRMSE Bi-BPCA-iLS	mr 1%	mr 5%	mr 10%	mr 15%	mr 20%	mr 25%	mr 30%
Test 1	0.3020	0.2593	0.2226	0.2351	0.2470	0.2595	0.2612
Test 2	0.1317	0.1737	0.2684	0.2527	0.2518	0.2515	0.2493
Test 3	0.3011	0.2814	0.2577	0.2378	0.2364	0.2234	0.2641
Test 4	0.3976	0.2838	0.2273	0.2791	0.2872	0.2357	0.2457
Test 5	0.4162	0.1917	0.2571	0.2595	0.2589	0.2485	0.3046
Average	0.30972	0.23798	0.24662	0.25284	0.25626	0.24372	0.26498

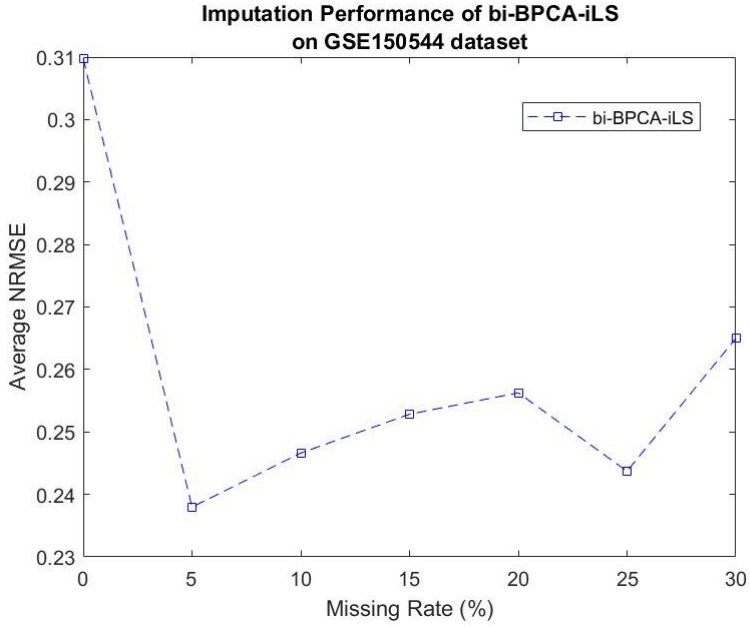

Figure 5. Imputation performance of bi-BPCA-iLS on GSE150544 dataset.

7. Evaluation and analysis

Two existing methods, LLS and bi-iLS, were compared to our proposed imputation method. This comparison entailed the use of the the average value of NRMSE and computational time generated from five trials for every missing rate. The difference between the NRMSE values of Method A and Method B divided by the NRMSE value of Method A shows the improvement of Method B relative to Method A. If the improvement value is positive, then Method B results in a higher imputation accuracy compared to Method A. If the improvement value is negative, then method B has a decrease in imputation accuracy compared to Method A.

7.1. CDC15 dataset

The averages of the improvement values across all missing rates for the CDC15 dataset are shown in Table 6 below. Based on these figures, the bi-iLS algorithm showed a significant overall improvement in NRMSE value (10.07%) relative to the LLS algorithm. Our proposed method, bi-BPCA-iLS, also showed a significant overall improvement in NRMSE value: 10.612% relative to LLS and 0.582% relative to bi-iLS.

Table 6. Performance of LLS, bi-iLS and bi-BPCA-iLS on CDC15 dataset.

Average value of NRMSE	NRMSE from LLS	NRMSE from Bi-iLS	NRMSE from Bi-BPCA-iLS	Improvement of bi-iLS relative to LLS	Improvement of bi-BPCA-iLS relative to LLS	Improvement of bi-BPCA-iLS relative to bi-iLS
Missing rate 1%	0.20938	0.20792	0.20284	0.697296781%	3.123507498%	2.443247403%
Missing rate 5%	0.51722	0.40444	0.40202	21.80503461%	22.27292061%	0.598358224%
Missing rate 10%	0.57694	0.49252	0.48916	14.63237078%	15.2147537%	0.682205799%
Missing rate 15%	0.61448	0.5499	0.54824	10.50969926%	10.77984637%	0.301873068%
Missing rate 20%	0.63408	0.5799	0.5785	8.544663134%	8.765455463%	0.241420935%
Missing rate 25%	0.65518	0.60512	0.60546	7.640648371%	7.588754235%	−0.05618720%
Missing rate 30%	0.6708	0.62610	0.62694	6.663685152%	6.538461538%	−0.13416387%
Average improvement				10.070%	10.612%	0.582%

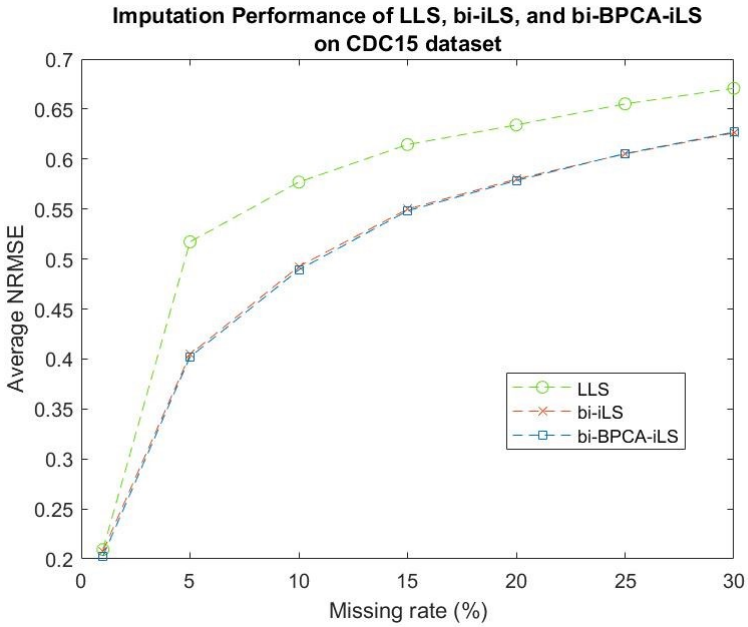

Figure 6. Imputation performance of bi-BPCA-iLS on GSE150544 dataset.

As shown in Table 6 and Figure 6, the imputation method that produced the lowest overall NRMSE across all missing rates for the CDC15 dataset was our proposed method, bi-BPCA-iLS.

After comparing the values of NRMSE, the computational times of the imputation methods were

also compared in MATLAB. Based on Table 7, bi-iLS is shown to add an overall average of 320.134 seconds of computational time compared to LLS. bi-BPCA-iLS is shown to add an overall average of 343.850 seconds of computational time relative to LLS and only 23.716 seconds relative to bi-iLS.

As shown in Figure 7, LLS displayed a consistent computational time for every missing rate, while bi-iLS and bi-BPCA-iLS had additional computational time following the increase in missing rates. In conclusion, the fastest imputation method was LLS; this is related to the high NRMSE it generated compared to the other methods. Regarding bi-iLS and bi-BPCA-iLS, there was no significant computational time difference between these two methods. If the goal is achieving a lower NRMSE, then one can use bi-BPCA-iLS instead of bi-iLS.

7.2. RNA-Seq

The average improvement values across all missing rates for the RNA-seq dataset (GSE150544) are shown in Table 8. We can see that the bi-iLS algorithm showed an overall improvement in NRMSE value of 5.12% relative to the LLS algorithm. Our proposed method, bi-BPCA-iLS, had an overall improvement in NRMSE value of 8.20% relative to LLS and 3.09% relative to bi-iLS.

The performances of LLS, bi-iLS, and bi-BPCA-iLS on the GSE1505544 data can be seen in Table 8. Bi-BPCA-iLS and bi-iLS had negative performances when the missing rate was 1% and 5%, so LLS performed well when the missing rate was below 5% in this dataset. But when the missing rate moved above 5%, the performance of bi-BPCA-iLS was superior to the other methods. As shown in Figure 8, the average NRMSE from bi-BPCA-iLS tended to be lower than those of the other methods.

After comparing the values of NRMSE, the computational times of the imputation methods were compared in MATLAB. Based on Table 9 and Figure 9, bi-iLS is shown to add an overall 117.200 seconds of computational time relative to LLS. While bi-BPCA-iLS is shown to add an overall 126.549 seconds of computational time relative to LLS and only 9.349 seconds relative to bi-iLS. There is no significant computational time difference between bi-BPCA-iLS and bi-iLS, only 9.349 seconds.

8. Conclusions

Early approaches toward missing-value imputation tended to consider all experimental conditions in measuring gene similarity. However, genes are only similar under certain experimental conditions. This meant that an bi-iLS algorithm for imputing missing values has to be developed. This algorithm uses the row average to obtain a temporary complete matrix, which has become to be considered as a flawed approach. The row average cannot reflect the real structure of the dataset because it only leverages the information of an individual row. Thus, in this study, we used BPCA to obtain a temporary complete matrix instead of using row average. The proposed algorithm is called bi-BPCA-iLS. After finding the temporary complete matrix using BPCA, the required parameters can be found. Our proposed algorithm performs clustering on genes and conditions alternately to find biclusters that consist of a subset of genes that are similar under a subset of conditions. After the biclusters related to the target genes are found, least squares estimation of the missing values can be performed while considering only related genes and conditions. This estimation process can be iterated to improve the selection of similar genes and conditions in every iteration, which improves the accuracy of the missing-value imputation.

Table 7. Computational times of LLS, bi-iLS and bi-BPCA-iLS on CDC15 dataset.

Average computational time	Computational time of LLS	Computational time of Bi-iLS	Computational time of Bi-BPCA-iLS	Additional time of bi-iLS relative to LLS	Additional time of bi-BPCA-iLS relative to LLS	Additional time of bi-BPCA-iLS relative to bi-iLS
Missing rate 1%	60.402	120.439	126.396	60.037	65.994	5.957
Missing rate 5%	76.419	255.455	275.087	179.036	198.668	19.632
Missing rate 10%	71.727	388.521	455.319	316.794	383.592	66.798
Missing rate 15%	59.123	378.844	444.281	319.721	385.158	65.437
Missing rate 20%	52.033	472.843	434.815	420.81	382.782	−38.028
Missing rate 25%	45.539	457.156	467.654	411.617	422.115	10.498
Missing rate 30%	61.047	593.972	629.689	532.925	568.642	35.717
Average additional computational time in seconds				320.134	343.850	23.716

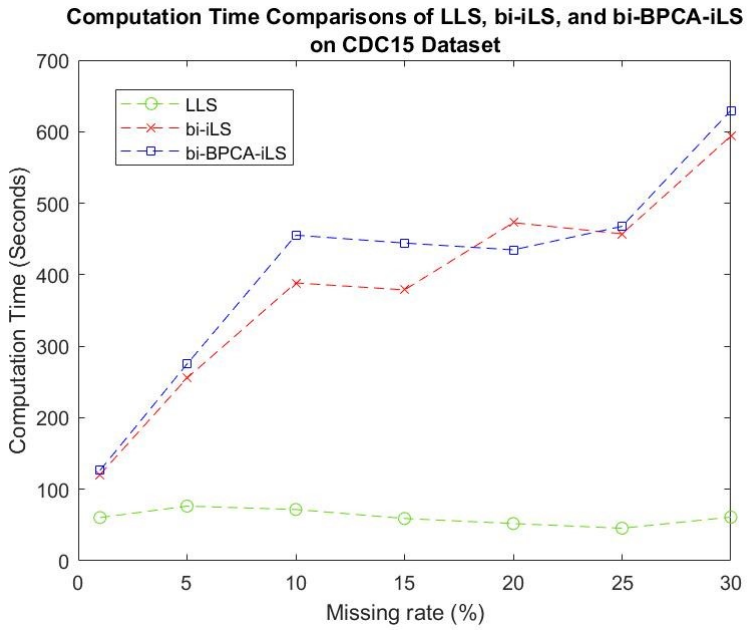

Figure 7. Computational time comparison for LLS, bi-iLS and bi-BPCA-iLS on CDC15 dataset.

Table 8. Performance of LLS, bi-iLS, and bi-BPCA-iLS on GSE150544 dataset.

Average value of NRMSE	NRMSE from LLS	NRMSE from Bi-iLS	NRMSE from Bi-BPCA-iLS	Improvement of bi-iLS relative to LLS	Improvement of bi-BPCA-iLS relative to LLS	Improvement of bi-BPCA-iLS relative to bi-iLS
Missing rate 1%	0.29318	0.32288	0.30972	−10.1303%	−5.64159%	4.075818%
Missing rate 5%	0.23604	0.25204	0.23798	−6.77851%	−0.82189%	5.57848%
Missing rate 10%	0.26032	0.25436	0.24662	2.28949%	5.262754%	3.042931%
Missing rate 15%	0.27980	0.26908	0.25284	3.831308%	9.635454%	6.03538%
Missing rate 20%	0.28308	0.25652	0.25626	9.382507%	9.474354%	0.101357%
Missing rate 25%	0.29156	0.24558	0.24372	15.77034%	16.40829%	0.757391%
Missing rate 30%	0.34442	0.27056	0.26498	21.44475%	23.06486%	2.062389%
Average improvement				5.12%	8.20%	3.09%

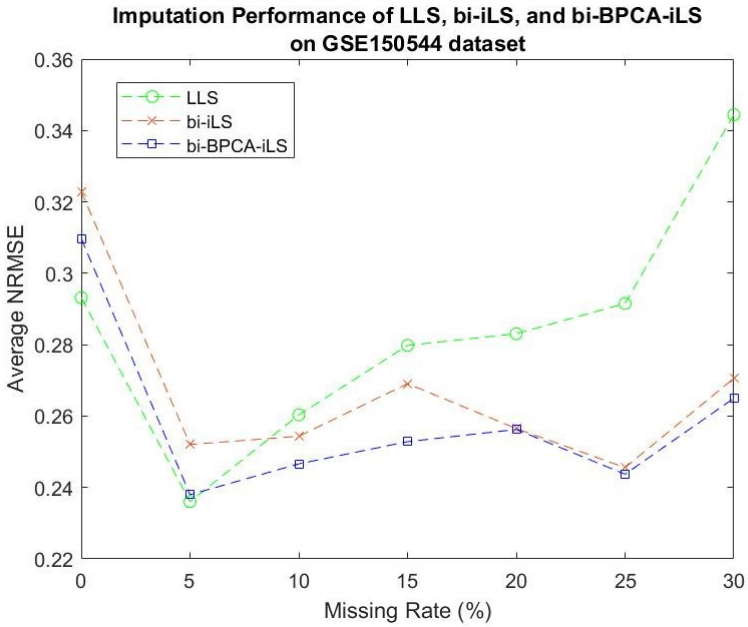

Figure 8. Imputation performances of LLS, bi-iLS and bi-BPCA-iLS on GSE150544 dataset.

Table 9. Computational time of LLS, bi-iLS, and bi-BPCA-iLS on GSE150544 dataset.

Average computational time	Computational time of LLS	Computational time of Bi-iLS	Computational time of Bi-BPCA-iLS	Additional time of bi-iLS relative to LLS	Additional time of bi-BPCA-iLS relative to LLS	Additional time of bi-BPCA-iLS relative to bi-iLS
Missing rate 1%	28.418	79.678	82.420	51.26	54.002	2.742
Missing rate 5%	22.080	107.653	109.090	85.573	87.01	1.437
Missing rate 10%	18.678	130.136	142.0635	111.458	123.3855	11.9275
Missing rate 15%	25.831	155.458	158.179	129.627	132.348	2.721
Missing rate 20%	22.429	160.494	165.366	138.065	142.937	4.872
Missing rate 25%	20.513	152.178	189.796	131.665	169.283	37.618
Missing rate 30%	25.018	197.769	201.895	172.751	176.877	4.126
Average additional computational time in seconds				117.200	126.549	9.349

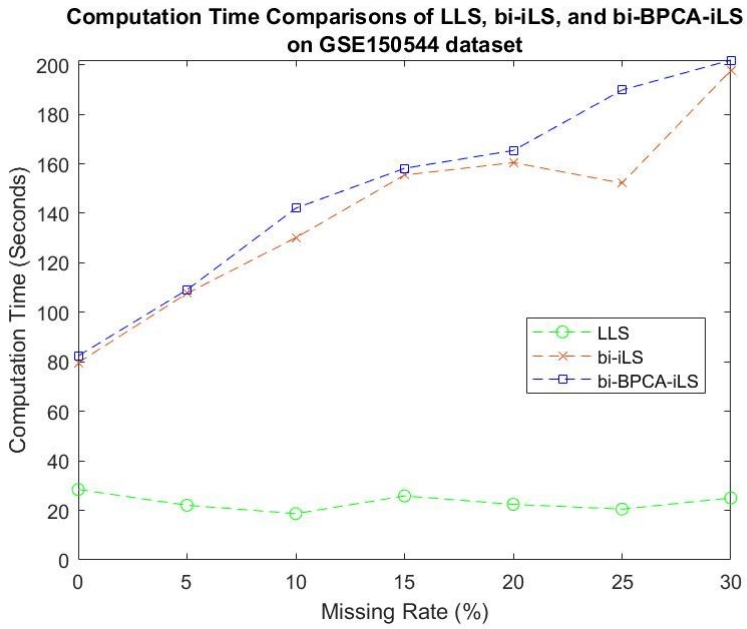

Figure 9. Computational time comparison for LLS, bi-iLS, and bi-BPCA-iLS on GSE150544 dataset.

Experiments were conducted on two gene expression datasets: a microarray dataset for *Saccharomyces cerevisiae* (CDC15) and an RNA-seq dataset for *Schizosaccharomyces pombe* (GSE150544). The results show that our proposed method is best suited to impute missing values in microarray datasets and RNA-seq datasets based on the NRMSE, compared to preceding imputation methods such as LLS and bi-iLS. Significant NRMSE improvements of 10.612% for CDC15 and 8.20% for GSE150544 were observed when using bi-BPCA-iLS instead of LLS, indicating the importance of using biclustering and iterative frameworks. Also, bi-BPCA-iLS showed NRMSE improvements of 0.582% for CDC15 and 3.09% for GSE150544 relative to bi-iLS, indicating that the temporary complete matrix is better obtained with BPCA rather than via the row average. The additional computational time of bi-BPCA-iLS compared to bi-iLS was only 23.716 seconds for CDC15 and 9.349 seconds for GSE150544, which can be concluded as not significant. These experimental results show that our proposed method outperforms the other two existing methods. Thus, our proposed method is applicable to other datasets that fit our assumption.

The missing-value imputation method bi-BPCA-iLS outperformed other methods such as LLS and bi-iLS in selected microarray and RNA-seq datasets in terms of the NRMSE. The improvement relative to LLS indicates the importance of using biclustering and iterative framework in the imputation, while the improvement relative to bi-iLS indicates that the temporary complete matrix is better obtained with BPCA rather than via the row average.

Acknowledgments

Universitas Indonesia funded this research with grant number NKB-030/UN2.F3/HKP.05.00/2021.

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. T. Siswantining, A. Bustamam, S. Puspa, Z. Rustam, F. Zubedi, Biclustering of diabetic nephropathy and diabetic retinopathy microarray data using a similarity-based biclustering algorithm, *Int. J. Bioinf. Res. Appl.*, **17** (2021), 343–362. https://doi.org/10.1504/ijbra.2021.117934
2. B. Pontes, R. Girldez, J. Aguilar-Ruiz, Quality measures for gene expression biclusters, *PloS One*, **10** (2015), e0115497. https://doi.org/10.1371/journal.pone.0115497
3. S. Madeira, A. Oliveira, Biclustering algorithms for biological data analysis: A survey, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **1** (2004), 24–45. https://doi.org/10.1109/TCBB.2004.2
4. K. Cheng, N. Law, W. Siu, Iterative bicluster-based least square framework for estimation of missing values in microarray gene expression data, *Pattern Recognit.*, **45** (2012), 1281–1289. https://doi.org/10.1016/j.patcog.2011.10.012
5. F. Shi, D. Zhang, J. Chen, H. Karimi, Missing value estimation for microarray data by Bayesian principal component analysis and iterative local least squares, *Math. Prob. Eng.*, **2013** (2013), 1–5. https://doi.org/10.1155/2013/162938

6. D. Rubin, Inference and missing data, *Biometrika*, **63** (1976), 581–592. https://doi.org/10.1093/biomet/63.3.581
7. S. Christopher, T. Siswantining, D. Sarwinda, A. Bustaman, Missing value analysis of numerical data using fractional hot deck imputation, in *2019 3rd International Conference On Informatics And Computational Sciences (ICICoS)*, (2019), 1–6. https://doi.org/10.1109/icicos48119.2019.8982412
8. A. G. De Brevern, S. Hazout, A. Malpertuy, Influence of microarrays experiments missing values on the stability of gene groups by hierarchical clustering, *BMC Bioinf.*, **5** (2004), 1–12. https://doi.org/10.1186/1471-2105-5-114
9. M. Celton, A. Malpertuy, G. Lelandais, A. G. De Brevern, Comparative analysis of missing value imputation methods to improve clustering and interpretation of microarray experiments, *BMC Genomics*, **11** (2010), 1–16. https://doi.org/10.1186/1471-2164-11-15
10. T. Siswantining, T. Anwar, D. Sarwinda, H. Al-Ash, A novel centroid initialization in missing value imputation towards mixed datasets, *Commun. Math. Biol. Neurosci.*, **11** (2021), 1–36. https://doi.org/10.28919/cmbn/5344
11. C. Mack, Z. Su, D. Weistreich, L. Research, *Managing Missing Data in Patient Registries: Addendum to Registries for Evaluating Patient Outcomes: A User's Guide*, Agency for Healthcare Research and Quality (US), 2018.
12. P. Berkhin, A survey of clustering data mining techniques, in *Grouping Multidimensional Data*, Springer, (2006), 25–71. https://doi.org/10.1007/3-540-28349-8_2
13. T. Siswantining, A. Aminanto, D. Sarwinda, O. Swasti, Biclustering analysis using plaid model on gene expression data of colon cancer, *Austrian J. Stat.*, **50** (2021), 101–114. https://doi.org/10.17713/ajs.v50i5.1195
14. H. Zhao, A. Liew, D. Wang, H. Yan, Biclustering analysis for pattern discovery: Current techniques, comparative studies and applications, *Curr. Bioinf.*. **7** (2012), 43–55. https://doi.org/10.2174/157489312799304413
15. A. Tanay, R. Sharan, R. Shamir, Biclustering algorithms: A survey. Handbook of computational molecular biology, **9** (2005), 122–124. https://doi.org/10.1201/9781420036275.ch26
16. H. Kim, G. Golub, H. Park, Missing value estimation for DNA microarray gene expression data: Local least squares imputation, *Bioinformatics*, **21** (2004), 187–198. https://doi.org/10.1093/bioinformatics/bth499
17. T. H. Bø, B. Dysvik, I. Jonassen, LSimpute: Accurate estimation of missing values in microarray data with least squares methods, *Nucleic Acids Res.*, **32** (2004), e34. https://doi.org/10.1093/nar/gnh026
18. L. Bras, J. Menezes, Dealing with gene expression missing data, *IEE Proc. Syst. Biol.*, **153** (2006), 105. https://doi.org/10.1049/ip-syb:20050056
19. S. Oba, M. Sato, I. Takemasa, M. Monden, K. Matsubara, S. Ishii, A Bayesian missing value estimation method for gene expression profile data, *Bioinformatics*, **19** (2003), 2088–2096. https://doi.org/10.1093/bioinformatics/btg287

20. G. Brock, J. Shaffer, R. Blakesley, M. Lotz, G. Tseng, Which missing value imputation method to use in expression profiles: A comparative study and two selection schemes, *BMC Bioinf.*, **9** (2008), 1–12. https://doi.org/10.1186/1471-2105-9-12

21. O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, et al., Missing value estimation methods for DNA microarrays, *Bioinformatics*, **17** (2001), 520–525. https://doi.org/10.1093/bioinformatics/17.6.520

22. A. Bustamam, S. Formalidin, T. Siswantining, Z. Rustam, Finding correlated biclusters from microarray data using the modified lift algorithm based on new residue score, *Int. J. Data Mining Bioinf.*, **24** (2020), 326. https://doi.org/10.1504/ijdmb.2020.113691

23. P. Spellman, G. Sherlock, M. Zhang, V. Iyer, K. Anders, M. Eisen, et al., Comprehensive identification of cell cycle-regulated genes of the YeastSaccharomyces cerevisiaeby microarray hybridization, *Mol. Biol. Cell*, **9** (1998), 3273–3297. https://doi.org/10.1091/mbc.9.12.3273

24. C. Shan, C. Bao, J. Diedrich, X. Chen, C. Lu, J. Yates, et al., The INO80 complex regulates epigenetic inheritance of heterochromatin, *Cell Rep.*, **33** (2020), 108561. https://doi.org/10.1016/j.celrep.2020.108561

AIMS Press