A Bicluster-Based Bayesian Principal Component Analysis Method for Microarray Missing Value Estimation

Fanchi Meng, Cheng Cai, and Hong Yan

***Abstract*—Data generated from microarray experiments often suffer from missing values. As most downstream analyses need full matrices as input, these missing values have to be estimated. Bayesian principal component analysis (BPCA) is a well-known microarray missing value estimation method, but its performance is not satisfactory on datasets with strong local similarity structure. A bicluster-based BPCA (bi-BPCA) method is proposed in this paper to fully exploit local structure of the matrix. In a bicluster, the most correlated genes and experimental conditions with the missing entry are identified, and BPCA is conducted on these biclusters to estimate the missing values. An automatic parameter learning scheme is also developed to obtain optimal parameters. Experimental results on four real microarray matrices indicate that bi-BPCA obtains the lowest normalized root-mean-square error on 82.14% of all missing rates.**

***Index Terms*—Bayesian principal component analysis (BPCA), biclustering, microarray missing value estimation.**

I. Introduction

DNA microarray technology is an effective tool in studying various biology processes such as cancer classification [1], specific therapy identification [2], drug mechanism investigation [3], etc. By adopting microarray technology, the mRNA levels of thousands of genes under different experimental conditions can be investigated simultaneously. The data generated from microarray experiments are usually in the form of large matrices. Generally, a row in the matrix represents a gene, and a column represents an experimental condition. A typical microarray data contains about 1000–20 000 genes and 5–100 experimental conditions. However, missing values are inevitable due to limitations in microarray experiments such as spotting problems, background noise in the scanned image, and various other technical reasons [4]. Although the genes that contain missing values can be entirely ignored, or the missing values in the microarray can be replaced with simple numbers such as zero or the row's average, these simple methods have been proven to be of little help to the downstream analyses [5], [6]. Another strategy is to repeat the experiment, but as an expensive method, it has been used in the validation of microarray analysis algorithms, rather than getting the missing values [7].

Current microarray missing value estimation methods can be divided into four categories [6]: 1) global approach, 2) local approach, 3) hybrid approach, and 4) knowledge assisted approach. Well-known global approaches include singular value decomposition (SVD) [8] and Bayesian principal Component Analysis (BPCA) [9]. SVD estimates the missing value j in gene i by first regressing this gene against K eigengenes (i.e., a set of mutually orthogonal expression patterns that can be linearly combined to approximate the expression of all genes) and use the coefficients of the regression to reconstruct value j from a linear combination of the K eigengenes. BPCA estimates the target gene (i.e., a gene that contains missing values) by a linear combination of K principal axis vectors (see (1) in Section II), where the parameters are identified by a Bayesian estimation method. Local approach is a large category which includes many methods. The most well-studied method in this category is local least squares (LLS) [10]. LLS uses a multiple regression model to impute the missing values from K nearest neighbor (KNN) genes of the target gene. Due to the simplicity and effectiveness of LLS, various LLS-derived methods have been proposed, including iterated local least squares (iLLS) [11], sequential local least squares [12], weighted local least squares [13], and iterative bicluster-based least squares (bi-iLS) [14]. Other famous local approaches include KNN [8], least squares (LS) [15], Gaussian mixture clustering [16], and a recently proposed autoregressive model-based least-squares (ARLS) method [17]. Local approaches are superior than global approaches in the presence of data with dominant local similarity (in this case, the data are heterogeneous), but in the presence of more homogenous data, global approaches may perform better [6]. To choose the optimal estimation tool for different types of data, hybrid methods were proposed with the aim of capturing both global and local correlations in the data. LinCmb [18] and EMDI [19] are two typical hybrid methods. Both the two methods estimate the missing values by a combination of other estimation methods from global approaches and local approaches. In the knowledge assisted category, domain knowledge or external information is integrated into the estimation process. For example, projection onto convex sets (POCS) [20] is a typical knowledge assisted method which designs a series of convex sets, taking into consideration

Manuscript received October 8, 2012; revised April 26, 2013; accepted September 30, 2013. Date of publication October 11, 2013; date of current version May 1, 2014. This work was supported in part by the National Natural Science Foundation of China under Project 61202188 and in part by the City University of Hong Kong under Project 7002843.

F. Meng and C. Cai are with the Department of Computer Science, College of Information Engineering, Northwest A&F University, Yangling 712100, China (e-mail: mpbchina@gmail.com; cheney.chengcai@gmail.com).

H. Yan is with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: h.yan@cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JBHI.2013.2284795

of the structure among genes, arrays and the biological phenomenon of synchronization loss in microarray experiments. POCS successively projects the solution onto these convex sets until reaching a point at the intersection of all convex sets.

Biclusters are coherent clusters consisting of correlated genes (rows) under correlated experimental conditions (columns). In this paper, these correlated experimental conditions are the columns that are correlated with the missing entry. The concept of biclustering was introduced early, but it did not become popular until 2000 when Cheng and Church applied it in the gene expression matrices [21]. Gan *et al*. introduced a geometrical biclustering method [22], where biclusters embedded in a matrix can be regarded as points distributed on special linear structures in high-dimensional space, and the Hough transform is applied to find these linear patterns in the high-dimensional space so that biclusters can be recognized. Other biclustering methods are proposed based on distance measures [23], probability models [24], and hypergraph-based geometry [25].

Recently, there have been studies about integrated framework of missing value estimation and bicluster analysis, such as bicluster-based impute (BIC) [26] and bi-iLS [14]. In BIC, the missing values are estimated by minimizing the coherence of subsets in gene expression matrix. Bi-iLS identifies biclusters for every individual missing value in the expression matrix, and applies iLLS to estimate the missing values in biclusters. In this study, we propose a local approach bicluster-based BPCA (bi-BPCA) to capture the local structure by biclustering, and estimate the missing values by imputing the missing values in biclusters using BPCA. The proposed method overcomes the shortcoming of BPCA that is incapable of handling local structure of the data, and can reduce the estimation error. The paper is organized as follows: Section II gives a brief review of the BPCA method. Section III introduces the bi-BPCA method. Then the proposed method is evaluated and compared with some existing methods in Section IV. Discussions about the proposed method are given in Section V. Finally, we conclude this paper in Section VI.

II. Review of Bayesian Principal Component Analysis

BPCA [9] regards that the D-dimensional microarray expression vector y can be represented as a linear combination of K $(K < D)$ principal axis vectors $\boldsymbol{w}_l$ $(1 \leq l \leq K)$:

$$\boldsymbol{y} = \sum_{l=1}^{K} x_l \boldsymbol{w}_l + \varepsilon \tag{1}$$

where the coefficient x_l is called a factor score and $\boldsymbol{\varepsilon}$ denotes the residual error. Principal axis vector $\boldsymbol{w_l} = \sqrt{\lambda_l}\boldsymbol{u_l}$, where λ_l and u_l are the lth eigenvalue and the corresponding eigenvector of the covariance matrix of the dataset $\boldsymbol{Y}$, respectively. The principal axis vectors are separated into two parts as $\boldsymbol{W} = (\boldsymbol{W}^{\mathrm{obs}}, W^{\mathrm{miss}})$, corresponding to the observed part and missing part, respectively. Factor scores $\boldsymbol{x} = (x_1, x_2, \ldots, x_K)$ are obtained by minimizing the residual error of the observed part of the dataset Y:

$$\mathrm{err} = \left\| \boldsymbol{y}^{\mathrm{obs}} - \boldsymbol{W}^{\mathrm{obs}} \boldsymbol{x} \right\|^2 \tag{2}$$

and the missing values are estimated as follows:

$$\boldsymbol{y}^{\mathrm{miss}} = \boldsymbol{W}^{\mathrm{miss}} \boldsymbol{x}. \tag{3}$$

In the aforementioned principal component regression model, $\boldsymbol{W}$ is unknown beforehand, but the factor scores $\boldsymbol{x}$ and the residual error $\boldsymbol{\varepsilon}$ are regarded to obey normal distributions in a probabilistic PCA model [27]:

$$\boldsymbol{x} \sim N_K(0, \boldsymbol{I}_K) \tag{4}$$

$$\varepsilon \sim N_D(0, (1/\tau)\boldsymbol{I}_D) \tag{5}$$

where $N_K(\mu, \boldsymbol{\Sigma})$ denotes a K-dimensional normal distribution whose mean and covariance are μ and $\boldsymbol{\Sigma}$, respectively, τ is a scalar inverse variance of ε, and I_k denotes a $K \times K$ identity matrix. BPCA assumes that only a part of the dataset $\boldsymbol{Y}$, $\boldsymbol{Y}^{\mathrm{obs}}$ is observed, and the rest $\boldsymbol{Y}^{\mathrm{miss}}$ is missing. The posterior distribution of the parameter set $\theta \equiv \{\boldsymbol{W}, \mu, \tau\}$ and $\boldsymbol{Y}^{\mathrm{miss}}$ is estimated by a variational Bayes algorithm [28] simultaneously.

There is only one parameter for the BPCA method, i.e., the number of principal axis vectors. The parameter is set to be D-1 by default, where D is the number of columns (experimental conditions) of the data matrix. An automatic relevance determination is used first to suppress redundant axes [9].

Though $\boldsymbol{W}$ and $\boldsymbol{Y}^{\mathrm{miss}}$ are gained by a Bayesian estimation method, BPCA is essentially a linear regression in low-dimensional space of the microarray matrix, which is based on the global feature of the whole matrix. In that the local similarity structure is not utilized, the performance of BPCA deteriorates on datasets that exhibit strong local similarity structure [29]. To make the best use of the local similarity structure of the matrix, we propose the bi-BPCA method in next section.

III. Bicluster-Based BPCA

Although the local similarity can be characterized easily by finding KNN of the target gene, as in LLS [10], uncorrelated columns may also be included in the neighborhood, which affects the accuracy of the estimation. As a result, we adopt biclusters to handle the local similarity structure of the matrix, where only the most correlated rows and columns with the missing entry are chosen to estimate the missing value. The bi-BPCA method is described as follows:

Step 1: The incomplete matrix is estimated by BPCA, to get a complete matrix. The reason for getting an initial complete matrix is that when finding a bicluster for a missing entry in the next step, we need KNN of the target gene. When most genes contain missing values, the distances between the target gene and other genes cannot be measured. In LLS, to measure the distances between the target gene and other genes, the missing entries are initially filled with the average value of nonmissing entries in the row to which they belong. However, row-average is a poor reflection of the real structure of the dataset as it does not utilize the correlation structure of the data [7], so we choose BPCA to get an initial complete matrix.

Step 2: Find a bicluster for every individual missing value. First, for every target gene, a set of KNN genes is identified from the initial complete matrix in the first step, according to the euclidean distances between the target gene and all the other

genes. The genes that have the k shortest distances are chosen to be the KNN genes. The KNN genes $(g_{s1}, g_{s2}, \ldots, g_{sk})^T$ are stacked up and rearranged as follows:

$$\begin{pmatrix} \boldsymbol{g}_{s1} \\ \boldsymbol{g}_{s2} \\ \vdots \\ \boldsymbol{g}_{sk} \end{pmatrix} = \begin{pmatrix} \boldsymbol{B} & \boldsymbol{A} \end{pmatrix} = \begin{pmatrix} B_{1,1} & B_{1,2} & \cdots & B_{1,p} & A_{1,1} & A_{1,2} & \cdots & A_{1,n-p} \\ B_{2,1} & B_{2,2} & \cdots & B_{2,p} & A_{2,1} & A_{2,2} & \cdots & A_{2,n-p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ B_{k,1} & B_{k,2} & \cdots & B_{k,p} & A_{k,1} & A_{k,2} & \cdots & A_{k,n-p} \end{pmatrix} \tag{6}$$

where p is the number of missing values in the target gene, matrix $\boldsymbol{B}$ consists of the p columns in the k neighbor genes corresponding to the p missing positions of the target gene, and matrix A consists of the $n-p$ columns in the k neighbor genes corresponding to the $n-p$ nonmissing positions of the target gene.

In biclustering, we consider that every individual condition has its own correlation with other conditions. In other words, conditions i and j $(1 \leq i, j \leq p, i \neq j)$ have different correlations with other $n-p$ conditions. To take into account the correlation among the $n-p$ different conditions for the p missing entries in the target gene, we introduce a matrix $\boldsymbol{R}$ in (7) as in bi-iLS [14]:

$$\boldsymbol{R} = \boldsymbol{B}^{\mathrm{T}} \boldsymbol{A} \tag{7}$$

then a set of k neighbor genes for the jth missing value of the target gene is reselected from the complete matrix in Step 1. The distances between the target gene and other genes for the jth missing value are calculated by a weighted euclidean distance (8) as in [14]:

$$d_j(\boldsymbol{g}_t, \boldsymbol{g}_s) = \sqrt{\sum_{v=p+1}^{n} r_j(v-p)^2 [\boldsymbol{g}_t(v) - \boldsymbol{g}_s(v)]^2} \Bigg/ \sqrt{\sum_{v=1}^{n-p} r_j(v)^2} \tag{8}$$

where $1 \leq j \leq p$ and $\boldsymbol{g}_t$ denote the target gene, $\boldsymbol{g}_s$ denotes one of the other genes in the matrix, and $r_j(v)$ denotes the (j, v)th element of matrix $\boldsymbol{R}$. In (8), $r_j(v)$ serves as a weight for calculating the distance between $\boldsymbol{g}_t$ and $\boldsymbol{g}_s$ in the vth position. The genes corresponding to the k smallest weighted euclidean distances are chosen to be the reselected neighbor genes for the jth missing value. In this case, the selected genes are considered to be the most correlated genes with the jth missing value of the target gene. The most correlated experimental conditions also have to be selected from the most correlated genes, this is determined by the value of $r_j(v)$. If $|r_j(v)| \geq T_0 r_{j,\max}$, then the vth experimental condition is considered to be correlated with the jth missing value, where $r_{j,\max} = \max_{v \in \{1,2,\ldots n-p\}} |r_j(v)|$, and T_0 is a preset threshold. After the uncorrelated genes and experimental conditions are removed, we get a subset A_j. The rows and columns of A_j are the most correlated genes and experimental conditions with the jth missing value of the target gene, respectively. The bicluster for the jth missing value is in this form:

$$\mathbf{bicluster}_j = \begin{pmatrix} \alpha_j & \boldsymbol{w}_j \\ \boldsymbol{b}_j & \boldsymbol{A}_j \end{pmatrix} \tag{9}$$

where α_j denotes the jth missing value of the target gene, $\boldsymbol{b}_j$ is the column in α_j's position in the most correlated genes, $\boldsymbol{w}_j$ denotes the nonmissing values in the most correlated locations with α_j, and $\boldsymbol{A}_j$ is the subset we found above. In a bicluster, the only unknown value is α_j, which is the jth missing value of the target gene.

Step 3: Conduct BPCA for a second time on biclusters. For a target gene containing p missing values, conduct BPCA in $\boldsymbol{bicluster}_j$ $(1 \leq j \leq p)$ until the p missing values are estimated, and we can get a complete gene vector.

There are two parameters in the bi-BPCA method, the number of nearest neighbors k and the value of the preset threshold T_0. The two parameters are determined by estimating simulated missing entries in the complete set. The number of neighbors k is estimated first and then is used to determine T_0. First, only the complete rows are chosen from the original matrix to get a complete matrix, then a number of entries are randomly removed based on the dataset's missing rate, and we obtain an artificial missing matrix. To determine the best k value, for every artificial target gene, we construct a series of subsets, these subsets consist of the target gene and its KNN, the range of k is from 1 to $N-1$, where N is the number of rows in the simulated missing matrix. The $N-1$ subsets are estimated by BPCA, and the missing entries in the artificial target gene have $N-1$ sets of estimated values for the individual target gene. When all the artificial target genes are considered, there are $N-1$ sets of estimated values for all the missing entries in the simulated missing matrix. As the real values of these artificial missing entries are actually known, the estimation error rates of the $N-1$ sets can be calculated. The k value corresponding to the lowest error rate is chosen to be an optimal parameter. After k is determined, for every missing entry in the artificial missing matrix, a series of biclusters are constructed with different T_0 values. These biclusters are estimated by BPCA and the error rates are evaluated in a similar way to the determination of k. The optimal value of T_0 is then determined as the value corresponding to the lowest error rate.

IV. Experiments and Results

For simplicity, we call the proposed bicluster-based BPCA method bi-BPCA. Bi-BPCA is evaluated on four real microarray datasets and compared with LLS [10], BPCA [9], and bi-iLS [14]. Some details of the four datasets are given in Table I.

The first dataset, Infection, comes from an infection time series study [30], which can be downloaded from http://genome-www.stanford.edu/listeria/gut/. The second dataset, Ronen, consists of two time series of yeast in study [31], which is available in http://ncbi.nlm.nih.gov/Projects/geo/query/acc.cgi?acc=GSE4158. The third dataset, Ogawa, is a nontime genome DNA microarray series from yeast [32], it is downloaded from

TABLE I
TESTING DATASETS

Dataset	Original size	Complete size	Overall missing rate	Type
Infection	16839×39	6851×39	7.21%	Time series
Ronen	10749×26	5342×26	18.12%	Time series
Ogawa	6013×8	5783×8	0.77%	Non-time series
Yoshi	6616×24	4380×24	3.82%	Mixed series

http: //smd.stanford.edu/cgi-bin//publication/viewPublication.pl?pub_no=68. The last dataset, Yoshi, composes both time-course and nontime-course data, it belongs to a study of gene expression in yeast [33] and can be downloaded from http://www.stanford.edu/group/cyert/microarray.html. Note that dataset Infection was used to evaluate the performance of LS in [15], Ronen and Ogawa were both used to test bi-iLS in [14], and dataset Yoshi was also used in survey [5] to evaluate the performances of different methods.

The original datasets all contain missing values. To assess the error rates for the methods, we randomly remove a number of entries to obtain missing matrices at certain missing rates. As the real values of these entries are known, the estimation error can be calculated. The same testing method was also used in LLS [10], BPCA [9], bi-iLS [14], and surveys [5], [6], [29].

The accuracy of the estimation result is evaluated by normalized root-mean-square error (NRMSE):

$$\text{NRMSE} = \sqrt{\sum_{j=1}^{N}(y_j - \hat{y}_j)^2 \Big/ N} \Big/ \sigma_y \tag{10}$$

where y_j is the real value, $\hat{y}_j$ is the estimated value, and σ_y is the standard deviation for the N actual values of the missing entries. A smaller NRMSE represents a higher accuracy. The same evaluation criterion is also employed in other papers [5], [6], [9], [10], [14], [29].

All parameters of LLS and bi-iLS are obtained automatically by the methods' heuristic parameter selection strategy, and the iterations of bi-iLS are set to be ten because the NRMSE does not change much after ten iterations in our experiments. The only parameter of BPCA is set to be the default value, i.e., $D - 1$, where D is the number of columns (experimental conditions) of the data matrix. The parameters k and T_0 of the proposed bi-BPCA method are determined from estimating simulated missing entries as mentioned in Section III. As the proposed method includes the BPCA procedure, the parameter of the BPCA procedure is also set to be the default value.

Fig. 1 shows the NRMSE against different missing rates (from 1% to 30%), on the four datasets: (a) Infection, (b) Ronen, (c) Ogawa, and (d) Yoshi. All the experiments are conducted five times to show the general estimation ability of the testing methods, and the NRMSE in Fig. 1 represents averaged values.

As can be seen from Fig. 1(a), on dataset Infection, the NRMSE of bi-BPCA is lower than that of LLS, BPCA, and bi-iLS, from missing rates 1% to 25%. The NRMSE of bi-iLS is the second lowest when the missing rate is between 5% and 15%, but when the missing rate is higher than 20%, the NRMSE of bi-iLS increases dramatically. On dataset Ronen in Fig. 1(b), the NRMSE of bi-BPCA is the lowest one from missing rates 10% to 30%. The NRMSE of bi-iLS is lower than that of LLS and BPCA when the missing rate is lower than 20%, which conforms to the experiment in [14]. On dataset Ogawa in Fig. 1(c), the NRMSE of bi-BPCA is the lowest one at all missing rates. Bi-iLS still outperforms LLS and BPCA when the missing rate is low (below 20%), but shows high NRMSE when the missing rate is higher than 20%. On the last dataset Yoshi in Fig. 1(d), the NRMSE of bi-BPCA is lower than all the other methods from missing rates 10% to 30%. Bi-iLS shows the lowest error rate at missing rates 1% and 5%, but when the missing rate increases, it is higher than that of bi-BPCA.

We can see from the four line graphs in Fig. 1 that bi-BPCA outperforms BPCA on 89.29% (25 out of 28) of all missing rates, which shows the advantage of adopting the bicluster scheme. The number of biclusters is the same with the number of missing values, and in a bicluster, the rows and columns that are uncorrelated with the missing entry are suppressed. The performance of bi-iLS deteriorates when the missing rate is high. This is probably due to the fact that bi-iLS identifies the *KNN* for the target gene in an initial complete matrix which is imputed by row-average. As the missing rate increases, the structure of the row-average imputed matrix is far from the real one, which affects the identification of biclusters. In that the proposed bi-BPCA method gets the initial complete matrix by BPCA, such deterioration can be avoided. However, the proposed bi-BPCA gets a relatively high NRMSE value at missing rates 1% and 5% on datasets Infection, Ronen, and Yoshi. This is because that the LLS-based methods (LLS and bi-iLS) use only complete rows to find neighbor genes when the missing rate is low (for LLS, if there are more than 400 complete rows in the missing matrix, it will not use the temporary full matrix that is filled by row-average) to highlight the original information of the matrix. Whereas for bi-BPCA, we are always using the temporary full matrix that is estimated by BPCA, these imputed values in the temporary full matrices still have large gaps between the true values.

Tables II–V provide the average NRMSE values and their standard deviations on dataset Infection, Ronen, Ogawa, and Yoshi, respectively, where the lowest NRMSE is denoted by a bold number. As can be seen from Tables II–V, among all 28 (7×4) missing rates, bi-BPCA obtains the lowest NRMSE in 23 missing rates, which accounts for 82.14% of all missing rates. The standard deviations of all the four methods are relatively small. For bi-BPCA, 92.86% (26 out of 28 missing rates) standard deviations are lower than 0.01, whereas for LLS, BPCA, and bi-iLS, this rate is 82.14% (23 out of 28 missing rates), 82.14% (23 out of 28 missing rates), and 89.29% (25 out of 28 missing rates), respectively.

Tables VI–IX show the average computation time and their standard deviations of the above four methods, on dataset Infection, Ronen, Ogawa, and Yoshi, respectively. The experiments are carried out using Matlab R2011b on a 64-bit Windows 7 computer with 3.4-GHz quad-core processor and 16-GB internal memory. Compared with LLS and BPCA, the computation time and standard deviations of bi-iLS and bi-BPCA are significantly larger, because both bi-iLS and bi-BPCA need to

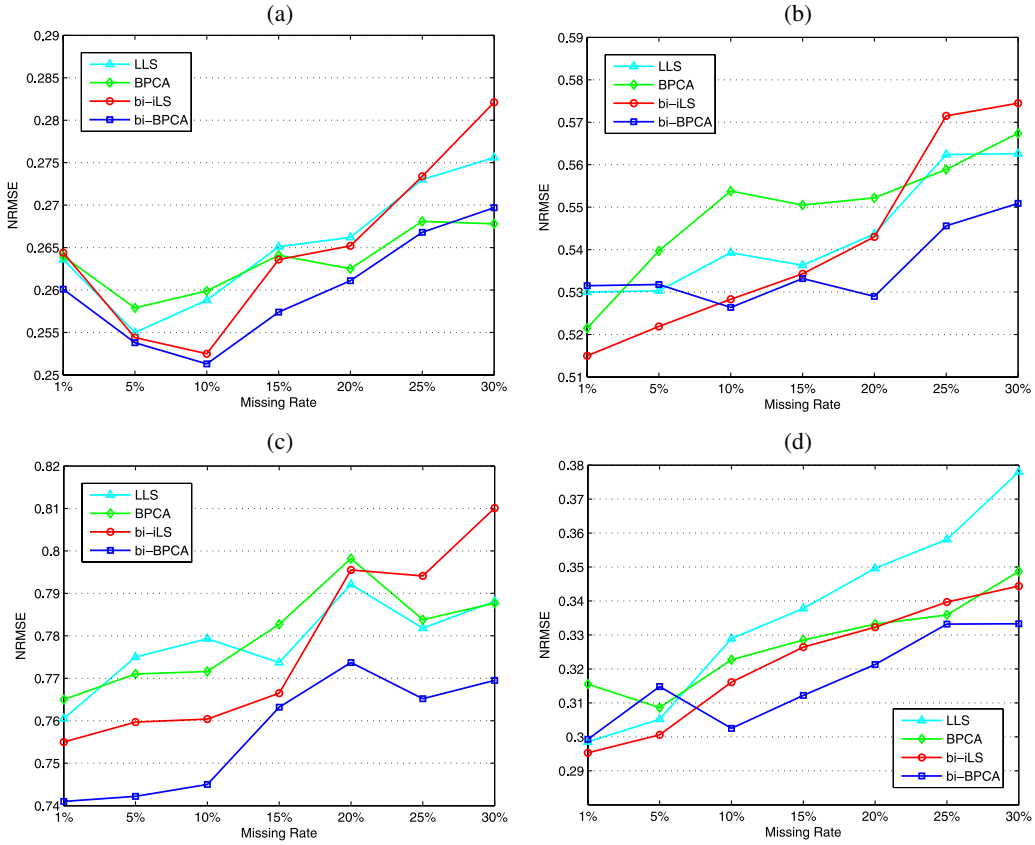

Fig. 1. NRMSE on four testing datasets: (a) Infection, (b) Ronen, (c) Ogawa, and (d) Yoshi.

TABLE II
NRMSE ON INFECTION

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	0.2636 ±0.0029	0.2640 ±0.0033	0.2644 ±0.0029	**0.2601** ±0.0031
5%	0.2550 ±0.0037	0.2579 ±0.0040	0.2544 ±0.0042	**0.2538** ±0.0035
10%	0.2588 ±0.0034	0.2599 ±0.0037	0.2525 ±0.0037	**0.2513** ±0.0033
15%	0.2651 ±0.0023	0.2641 ±0.0030	0.2636 ±0.0043	**0.2574** ±0.0054
20%	0.2662 ±0.0036	0.2625 ±0.0045	0.2652 ±0.0039	**0.2611** ±0.0070
25%	0.2730 ±0.0035	0.2681 ±0.0041	0.2734 ±0.0038	**0.2668** ±0.0042
30%	0.2756 ±0.0042	**0.2678** ±0.0070	0.2821 ±0.0076	0.2697 ±0.0052

TABLE III
NRMSE ON RONEN

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	0.5300 ±0.0015	0.5215 ±0.0013	**0.5150** ±0.0085	0.5315 ±0.0055
5%	0.5303 ±0.0016	0.5397 ±0.0016	**0.5219** ±0.0091	0.5318 ±0.0064
10%	0.5393 ±0.0107	0.5538 ±0.0110	0.5283 ±0.0270	**0.5264** ±0.0057
15%	0.5363 ±0.0035	0.5505 ±0.0040	0.5343 ±0.0078	**0.5332** ±0.0095
20%	0.5437 ±0.0097	0.5522 ±0.0110	0.5430 ±0.0022	**0.5290** ±0.0090
25%	0.5624 ±0.0039	0.5589 ±0.0042	0.5715 ±0.0037	**0.5456** ±0.0085
30%	0.5626 ±0.0057	0.5674 ±0.0064	0.5745 ±0.0097	**0.5509** ±0.0077

find a bicluster for every missing value. The search of parameters is also a high-computational task.

The computational time of bi-BPCA is comparable with that of bi-iLS on datasets Infection and Yoshi, but it is shorter than that of bi-iLS on datasets Ronen and Ogawa. When the computational time is long (in the presence of large matrices such as Infection and Ronen), the standard deviation of bi-BPCA's computational time is significantly larger than that of bi-iLS.

V. DISCUSSIONS

Brock *et al.* proposed an entropy-based method to evaluate the data's complexity in [29], the complexity of a dataset $\boldsymbol{D}$ can be represented by its entropy $e(\boldsymbol{D})$:

$$e(\boldsymbol{D}) = -\frac{\sum_{i=1}^{k} p_i \log p_i}{\log(k)} \tag{11}$$

TABLE IV
NRMSE ON OGAWA

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	0.7605 ±0.0088	0.7650 ±0.0065	0.7550 ±0.0066	**0.7410** ±0.0074
5%	0.7750 ±0.0138	0.7710 ±0.0108	0.7597 ±0.0075	**0.7422** ±0.0088
10%	0.7793 ±0.0076	0.7716 ±0.0061	0.7604 ±0.0065	**0.7450** ±0.0050
15%	0.7737 ±0.0103	0.7827 ±0.0067	0.7665 ±0.0079	**0.7632** ±0.0057
20%	0.7921 ±0.0173	0.7982 ±0.0164	0.7955 ±0.0139	**0.7737** ±0.0098
25%	0.7818 ±0.0070	0.7838 ±0.0076	0.7941 ±0.0117	**0.7652** ±0.0110
30%	0.7881 ±0.0073	0.7877 ±0.0093	0.8101 ±0.0089	**0.7695** ±0.0115

TABLE V
NRMSE ON YOSHI

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	0.2985 ±0.0063	0.3155 ±0.0065	**0.2953** ±0.0045	0.2993 ±0.0067
5%	0.3052 ±0.0079	0.3086 ±0.0089	**0.3006** ±0.0039	0.3148 ±0.0071
10%	0.3289 ±0.0117	0.3227 ±0.0129	0.3161 ±0.0079	**0.3025** ±0.0073
15%	0.3378 ±0.0042	0.3285 ±0.0038	0.3264 ±0.0064	**0.3122** ±0.0081
20%	0.3496 ±0.0029	0.3332 ±0.0034	0.3323 ±0.0062	**0.3213** ±0.0083
25%	0.3581 ±0.0030	0.3359 ±0.0055	0.3397 ±0.0054	**0.3332** ±0.0076
30%	0.3780 ±0.0066	0.3487 ±0.0053	0.3444 ±0.0076	**0.3333** ±0.0086

TABLE VI
COMPUTATION TIME (SECONDS) ON INFECTION

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	504.27 ±188.56	14.06 ±0.66	2451.426 ± 135.47	3199.45 ±2463.22
5%	968.14 ±192.70	38.69 ±0.78	8300.06 ± 143.75	5765.83 ±2663.60
10%	909.15 ±168.90	43.71 ±0.46	14444.26 ±192.69	20138.67 ±4295.28
15%	832.76 ±113.48	43.70 ±0.42	17929.94 ±266.96	14039.46 ±4438.31
20%	805.80 ±103.76	44.10 ±0.66	23514.74 ±145.45	34870.15 ±4946.69
25%	640.94 ±153.09	46.82 ±0.60	26650.35 ± 205.36	33486.18 ±3550.95
30%	571.09 ±136.33	54.57 ±0.56	31969.37 ±169.07	38262.30 ±4620.98

TABLE VII
COMPUTATION TIME (SECONDS) ON RONEN

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	263.21 ±5.72	5.80 ±0.46	842.28 ± 29.40	710.04 ±100.23
5%	276.28 ±6.52	24.17 ±0.22	2609.17 ± 31.95	1488.87 ±115.28
10%	256.61 ±8.09	30.92 ±0.41	4559.05 ±28.49	2887.71 ±184.83
15%	236.78 ±7.46	31.42 ± 0.36	6217.86 ±43.08	4140.42 ±123.28
20%	222.52 ±12.11	40.62 ±0.24	7776.83 ±52.90	5257.71 ±214.09
25%	174.75 ±6.99	42.07 ±0.29	9060.95 ± 37.52	5307.33 ±119.28
30%	161.15 ±10.10	49.98 ±0.33	10662.09 ±40.69	6621.50 ±199.46

TABLE VIII
COMPUTATION TIME (SECONDS) ON OGAWA

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	77.84 ±2.43	3.29 ±0.40	628.70 ±15.88	536.19 ±14.33
5%	74.15 ±2.64	14.10 ±0.42	1109.08 ±12.43	784.45 ±12.13
10%	56.16 ±2.30	22.26 ±0.39	1666.88 ±7.8	1064.03 ±22.16
15%	90.86 ±3.16	35.16 ±0.37	2186.75 ±14.49	1289.09 ± 11.93
20%	87.47 ±4.28	45.96 ±0.69	2674.52 ±11.47	1512.80 ±12.30
25%	72.21 ±2.81	60.62 ±0.39	3150.50 ± 13.46	1645.41 ±12.03
30%	71.32 ±3.29	71.70 ±0.54	3619.05 ±9.64	2055.38 ± 17.23

TABLE IX
COMPUTATION TIME (SECONDS) ON YOSHI

Missing rate	LLS	BPCA	Bi-iLS	Bi-BPCA
1%	145.13 ± 1.80	5.93 ± 1.23	557.36 ± 40.51	683.89 ±60.75
5%	192.38 ± 1.91	18.91 ± 1.00	1683.81 ± 37.07	1569.88 ±59.12
10%	180.91 ±2.15	24.37 ±0.97	2896.22 ±42.03	3000.62 ±49.08
15%	169.01 ±1.83	26.24 ±1.15	3961.41 ±49.84	3927.29 ±52.57
20%	157.00 ±3.40	25.37 ±1.81	4869.94 ±26.84	4242.74 ±91.16
25%	129.50 ±1.87	35.35 ±1.07	5722.46 ±43.45	4939.94 ±55.84
30%	116.58 ±2.77	34.21 ±1.39	6571.75 ±34.43	6195.83 ± 70.12

where $p_i = \sqrt{\lambda_i} \Big/ \sum_{l=1}^{k} \sqrt{\lambda_l}$, and λ_i denotes the ith eigenvalue of the covariance matrix of D, and k is the rank of the covariance matrix. The entropy in (11) actually evaluates the data's complexity by mapping the data to a low-dimensional space. Low entropy indicates that the entries in the matrix are strongly correlated so that it can be reduced to a low-dimensional space, and high entropy indicates a data with strong local similarity substructure [29]. We plot entropies of biclusters, as well as entropy of the whole matrix. The entropy of the whole matrix is calculated only in the complete set to reflect the real structure of the dataset. To find biclusters of a matrix, 10% artificial missing entries are removed from the complete dataset. Fig. 2 shows the entropies of the four datasets: 1) Infection, 2) Ronen, 3) Ogawa, and 3) Yoshi.

As can be seen from Fig. 2(a), for dataset Infection, all entropies of biclusters are larger than that of the whole matrix. Similarly, Fig. 2(b) and (d) demonstrates that for Ronen and Yoshi, more than 90% entropies of biclusters are larger than that of the whole matrix. Although high entropy reveals that

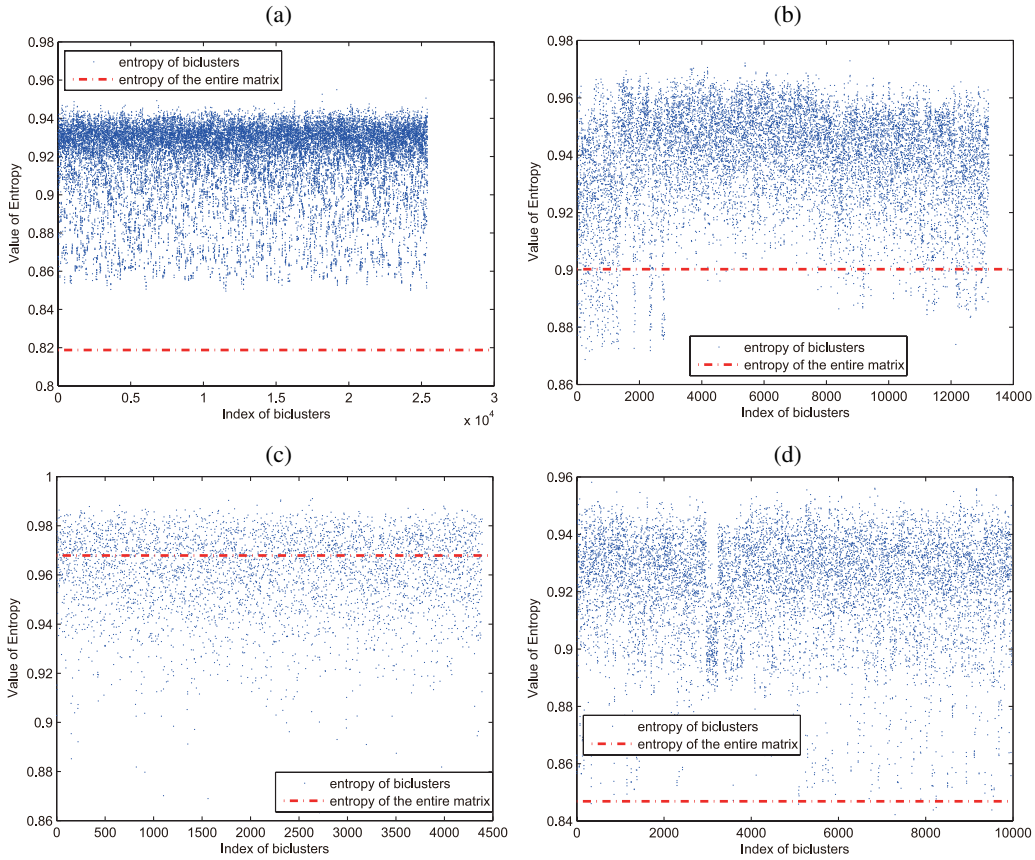

Fig. 2. Entropies of biclusters and entropy of the whole matrix: (a) Infection, (b) Ronen, (c) Ogawa, and (d) Yoshi.

the data cannot be easily mapped to a low-dimensional space, in this case, almost all principal axis vectors are used to construct the target gene in BPCA because the parameter k is set to $D-1$ (see Section II). As a result, the increment of entropy has only trivial influence on the BPCA procedure. Actually, further SVD experiment in the next paragraph has shown that the "average" low-dimensional structure of biclusters is still similar to that of the whole matrix. Fig. 2(c) shows the entropy for dataset Ogawa. Here, 61.66% entropies of biclusters are smaller than that of the whole matrix. This reveals that in dataset Ogawa, the data can be more effectively mapped to a low-dimensional space in biclusters.

To further evaluate the low-dimensional structure of the whole dataset and biclusters, we compute SVD of the whole matrix and its biclusters. In Fig. 3, the percentage of the ith singular value $\lambda_i/\sum_{j=1}^{n}\lambda_j$ is plotted in the left y-axis, and the accumulated percentages from the first singular value to the ith one $\sum_{1}^{i}(\lambda_i/\sum_{j=1}^{n}\lambda_j)$ is plotted in the right y-axis. To reflect the real structure of the microarray matrix, SVD is computed only in the complete part of the matrix (simulated 10% missing entries are only recorded for finding biclusters). The two blue lines refer to the percentages (left y-axis) and accumulated percentages (right y-axis) of singular values in biclusters. Since there are many biclusters according to the number of missing entries, the percentages and accumulated percentages are the average values. The red lines indicate the percentages (left y-axis) and accumulated percentages (right y-axis) of singular values of the whole matrix.

As shown in Fig. 3, for datasets Infection, Ronen, and Yoshi, the distributions of average percentages and accumulated percentages of singular values in biclusters do not vary much from that of the whole matrix. This reveals that the "average" low-dimensional structure of biclusters is still similar to that of the whole matrix although the entropies of biclusters of the three datasets have increased. For dataset Ogawa, the average percentages of singular values of biclusters decline faster than the percentage of the whole matrix, especially in the first two positions. Also, the average accumulated percentages of singular values of biclusters in Ogawa are always larger than that of the whole matrix, from the first position to the second last one. This reveals that more energy is concentrated on the first few singular values in biclusters of Ogawa. Thus, in dataset Ogawa, the low-dimensional structure is enhanced in biclusters, which helps BPCA to get a higher accuracy.

Fig. 3(a), (b), and (d) reveals that in datasets Infection, Ronen, and Yoshi, the "average" low-dimensional structure of biclusters is not significantly changed from that of the whole matrix. Fig. 3(c) demonstrates that the low-dimensional structure is enhanced in biclusters of Ogawa. In addition to the fact that the local similarity structure is fully exploited in biclusters, it is not surprising that bi-BPCA obtains the lowest NRMSE in 82.14% missing rates on the four datasets.

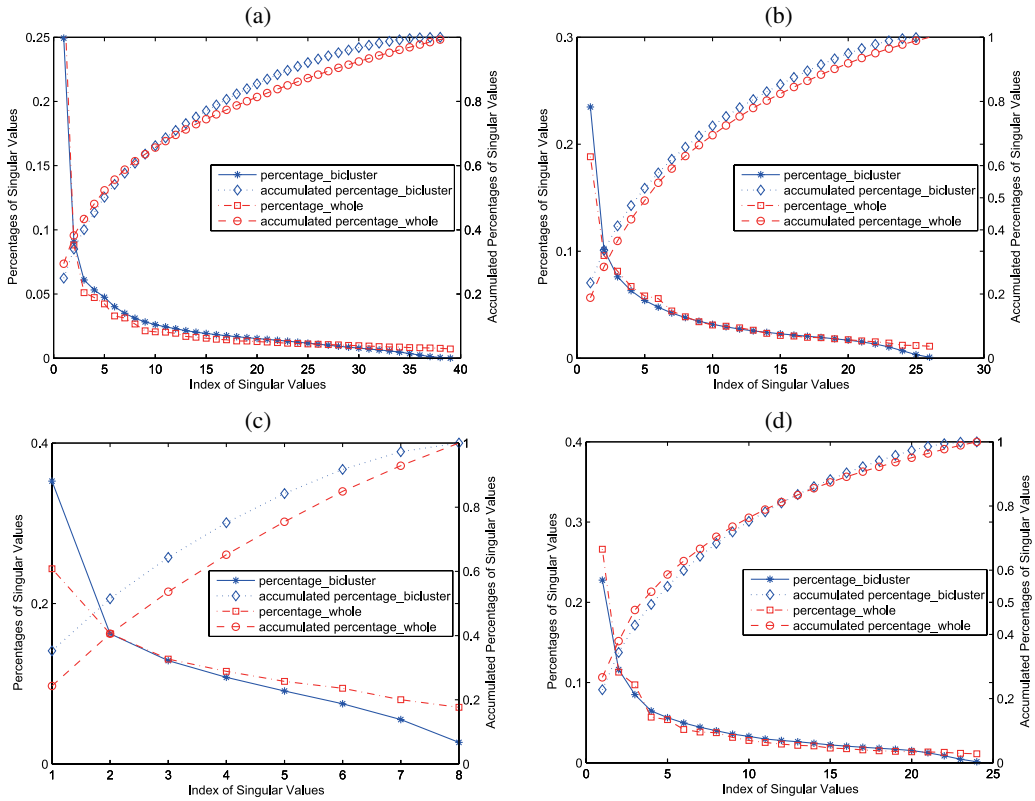

Fig. 3. Percentages and accumulated percentages of singular values: (a) Infection, (b) Ronen, (c) Ogawa, and (d) Yoshi.

VI. Conclusion

In this bi-BPCA method, BPCA is conducted twice, first on the original missing matrix, and second on biclusters for every missing value. A bicluster consists of the most correlated genes and experimental conditions for the missing value, where the uncorrelated rows and columns are removed. The adoption of biclusters overcomes the drawback of handling insufficient local similarity of BPCA. An automatic parameter learning strategy is used to get optimal parameters for bi-BPCA.

Experimental results on four real microarray datasets have shown that bi-BPCA produces the lowest estimation error rates on 82.14% of the overall missing rates. Validation experiments have also revealed that the increment of entropies in biclusters has litter impact on the "average" low-dimensional structure of these biclusters. Furthermore, the low-dimensional structure is enhanced in biclusters on a test dataset Ogawa. The drawback of bi-BPCA lies in its computational cost. Although the computational time is comparable with that of bi-iLS, it is still significantly longer than that of LLS and BPCA. Thus, the bicluster-based methods, e.g., bi-iLS and bi-BPCA are especially useful on more powerful computers.

References

[1] S. A. Armstrong, J. E. Staunton, L. B. Silverman, R. Pieters, M. L. den Boer, M. D. Minden, S. E. Sallan, E. S. Lander, T. R. Golub, and S. J. Korsmeyer, "MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nat. Genet.*, vol. 30, no. 1, pp. 41–47, 2002.

[2] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, and M. A. Caligiuri, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, no. 5439, pp. 531–537, 1999.

[3] S. Kim, E. R. Dougherty, Y. Chen, K. Sivakumar, P. Meltzer, J. M. Trent, and M. Bittner, "Multivariate measurement of gene expression relationships," *Genomics*, vol. 67, no. 2, pp. 201–209, 2000.

[4] R. Jörnsten, H. Wang, W. J. Welsh, and M. Ouyang, "DNA microarray data imputation and significance analysis of differential expression," *Bioinformatics*, vol. 21, no. 22, pp. 4155–4161, 2005.

[5] L. P. Bras and J. C. Menezes, "Dealing with gene expression missing data," *Syst. Biol. (Stevenage)*, vol. 153, no. 3, pp. 105–119, May 2006.

[6] A. W. Liew, N. F. Law, and H. Yan, "Missing value imputation for gene expression data: Computational techniques to recover missing data from available information," *Brief Bioinform.*, vol. 12, no. 5, pp. 498–513, Sep. 2011.

[7] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman, "Missing value estimation methods for DNA microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.

[8] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman, "Missing value estimation methods for DNA microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, Jun. 2001.

[9] S. Oba, M. A. Sato, I. Takemasa, M. Monden, K. Matsubara, and S. Ishii, "A Bayesian missing value estimation method for gene expression profile data," *Bioinformatics*, vol. 19, no. 16, pp. 2088–2096, Nov. 1, 2003.

[10] H. Kim, G. H. Golub, and H. Park, "Missing value estimation for DNA microarray gene expression data: Local least squares imputation," *Bioinformatics*, vol. 21, no. 2, pp. 187–198, Jan. 15, 2005.

[11] Z. Cai, M. Heydari, and G. Lin, "Iterated local least squares microarray missing value imputation," *J. Bioinform. Comput. Biol.*, vol. 4, no. 5, pp. 935–957, Oct. 2006.

[12] X. Zhang, X. Song, H. Wang, and H. Zhang, "Sequential local least squares imputation estimating missing value of microarray data," *Comput. Biol. Med.*, vol. 38, no. 10, pp. 1112–1120, 2008.

[13] W. K. Ching, L. Li, N. K. Tsing, C. W. Tai, T. W. Ng, A. Wong, and K. W. Cheng, "A weighted local least squares imputation method for missing value estimation in microarray gene expression data," *Int. J. Data Mining Bioinform.*, vol. 4, no. 3, pp. 331–347, 2010.
[14] K. O. Cheng, N. F. Law, and W. C. Siu, "Iterative bicluster-based least square framework for estimation of missing values in microarray gene expression data," *Pattern Recog.*, vol. 45, no. 4, pp. 1281–1289, 2012.
[15] T. H. Bø, B. Dysvik, and I. Jonassen, "LSimpute: Accurate estimation of missing values in microarray data with least squares methods," *Nucl. Acids Res.*, vol. 32, no. 3, pp. e34.1–e34.8, 2004.
[16] M. Ouyang, W. J. Welsh, and P. Georgopoulos, "Gaussian mixture clustering and imputation of microarray data," *Bioinformatics*, vol. 20, no. 6, pp. 917–923, Apr. 12, 2004.
[17] M. K. Choong, M. Charbit, and H. Yan, "Autoregressive-model-based missing value estimation for DNA microarray time series data," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, no. 1, pp. 131–137, Jan. 2009.
[18] R. Jornsten, H. Y. Wang, W. J. Welsh, and M. Ouyang, "DNA microarray data imputation and significance analysis of differential expression," *Bioinformatics*, vol. 21, no. 22, pp. 4155–4161, 2005.
[19] X. Pan, Y. Tian, Y. Huang, and H. Shen, "Towards better accuracy for missing value estimation of epistatic miniarray profiling data by a novel ensemble approach," *Genomics*, vol. 97, no. 5, pp. 257–264, 2011.
[20] X. Gan, A. W. C. Liew, and H. Yan, "Microarray missing data imputation based on a set theoretic framework and biological knowledge," *Nucl. Acids Res.*, vol. 34, no. 5, pp. 1608–1619, 2006.
[21] Y. Cheng and G. M. Church, "Biclustering of expression data," in *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 2000, pp. 93–103.
[22] X. Gan, A. W. Liew, and H. Yan, "Discovering biclusters in gene expression data based on high-dimensional linear geometries," *BMC Bioinform.*, vol. 9, no. 1, pp. 209–223, 2008.
[23] W. C. Tjhi and L. Chen, "A partitioning based algorithm to fuzzy co-cluster documents and words," *Pattern Recog. Lett.*, vol. 27, no. 3, pp. 151–159, 2006.
[24] S. Das and S. M. Idicula, "Application of cardinality based grasp to the biclustering of gene expression data," *Int. J. Comput. Appl.*, vol. 1, no. 18, pp. 47–54, 2010.
[25] Z. Wang, C. W. Yu, R. C. C. Cheung, and H. Yan, "Hypergraph based geometric biclustering algorithm," *Pattern Recog. Lett.*, vol. 33, no. 12, pp. 1656–1665, 2012.
[26] R. Ji, D. Liu, and Z. Zhou, "A bicluster-based missing value imputation method for gene expression data," *J. Comput. Inf. Syst.*, vol. 7, no. 13, pp. 4810–4818, 2011.
[27] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Comput.*, vol. 11, no. 2, pp. 443–482, 1999.
[28] H. Attias, "Inferring parameters and structure of latent variable models by variational Bayes," in *15th. Proc. Conf Uncertainty Artif. Intell*, 1999, pp. 21–30.
[29] G. N. Brock, J. R. Shaffer, R. E. Blakesley, M. J. Lotz, and G. C. Tseng, "Which missing value imputation method to use in expression profiles: A comparative study and two selection schemes," *BMC Bioinform.*, vol. 9, no. 1, pp. 12–24, 2008.
[30] D. N. Baldwin, V. Vanchinathan, P. O. Brown, and J. A. Theriot, "A gene-expression program reflecting the innate immune response of cultured intestinal epithelial cells to infection by Listeria monocytogenes," *Genome Biol.*, vol. 4, no. 1, pp. R2.1–R2.14, 2003.
[31] M. Ronen and D. Botstein, "Transcriptional response of steady-state yeast cultures to transient perturbations in carbon source," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 103, pp. 389–394, 2006.
[32] N. Ogawa, J. DeRisi, and P. O. Brown, "New components of a system for phosphate accumulation and polyphosphate metabolism in Saccharomyces cerevisiae revealed by genomic expression analysis," *Mol. Biol. Cell*, vol. 11, no. 12, pp. 4309–4321, 2000.
[33] H. Yoshimoto, K. Saltsman, A. P. Gasch, H. X. Li, N. Ogawa, D. Botstein, P. O. Brown, and M. S. Cyert, "Genome-wide analysis of gene expression regulated by the calcineurin/Crz1p signaling pathway in Saccharomyces cerevisiae," *J. Biol. Chem.*, vol. 277, no. 34, pp. 31079–31088, Aug. 23, 2002.

Fanchi Meng was born in Zhengzhou, Henan Province, China, in 1986. He received the B.Eng. degree from the College of Information Engineering, Northwest A&F University, Yangling, Shaanxi Province, China, in 2010, where he is currently working toward the Master degree at the Department of Computer Science under the supervision of Prof. S. Li and Dr. C. Cai.

His major is computer application technology, and his research interest includes bioinformatics with focus on microarray missing value estimation.

Cheng Cai received the B.S. degree in information engineering, and the M.E. and Ph.D. degrees in information and communication engineering all from Xi'an Jiaotong University, China, in 2001, 2003, and 2008, respectively.

From March 2004 to December 2005, he was a Research Assistant in the Multimedia Center, Hong Kong Polytechnic University. From June 2008 to July 2008, he was a Visiting Scientist in the Department of Computer Science, Oldenburg University, Germany. From July 2009 to September 2009, he was a Research Associate in the Multimedia Center, Hong Kong Polytechnic University. From July 2010 to August 2010, he was a Senior Research Associate in the Department of Electronic Engineering, City University of Hong Kong. From December 2012 to March 2013, he was a Postdoctoral Fellow in the Institute of Computer Graphics and Algorithms, Vienna University of Technology, Vienna, Austria. He joined the Department of Computer Science, Northwest A&F University, China, as a Lecturer in June 2008, and is currently an Associate Professor. His research interests include pattern recognition, signal processing, and bioinformatics. He has authored more than 30 papers in journals and conferences, and has served as a reviewer for many journals and conferences. He organized a special session on APSIPA ASC 2010 Singapore.

Dr. Cai is a member of the Asia-Pacific Signal and Information Processing Association.

Hong Yan photograph and biography not available at the time of publication.