BIOINFORMATICS

Vol. 19 no. 16 2003, pages 2088–2096
DOI: 10.1093/bioinformatics/btg287

A Bayesian missing value estimation method for gene expression profile data

Shigeyuki Oba[1], Masa-aki Sato[2,5], Ichiro Takemasa[3], Morito Monden[3], Ken-ichi Matsubara[4] and Shin Ishii[1,5,*]

[1]Graduate School of Information Science, Nara Institute of Science and Technology, 8916-5 Takayama, Ikoma 630-0192, Japan, [2]ATR Human Information Science Laboratories, 2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto, Japan, [3]Graduate School of Medicine, Osaka University, 2-2 Yamadaoka, Suita, Osaka, Japan, [4]DNA Chip Research Institute, 134 Kobecho, Hodogayaku, Yokohama, Japan and [5]CREST, Japan Science and Technology Corporation

Received on March 10, 2003; revised on May 6, 2003; accepted on May 9, 2003

ABSTRACT

Motivation: Gene expression profile analyses have been used in numerous studies covering a broad range of areas in biology. When unreliable measurements are excluded, missing values are introduced in gene expression profiles. Although existing multivariate analysis methods have difficulty with the treatment of missing values, this problem has received little attention. There are many options for dealing with missing values, each of which reaches drastically different results. Ignoring missing values is the simplest method and is frequently applied. This approach, however, has its flaws. In this article, we propose an estimation method for missing values, which is based on Bayesian principal component analysis (BPCA). Although the methodology that a probabilistic model and latent variables are estimated simultaneously within the framework of Bayes inference is not new in principle, actual BPCA implementation that makes it possible to estimate arbitrary missing variables is new in terms of statistical methodology.

Results: When applied to DNA microarray data from various experimental conditions, the BPCA method exhibited markedly better estimation ability than other recently proposed methods, such as singular value decomposition and K-nearest neighbors. While the estimation performance of existing methods depends on model parameters whose determination is difficult, our BPCA method is free from this difficulty. Accordingly, the BPCA method provides accurate and convenient estimation for missing values.

Availability: The software is available at http://hawaii.aist-nara.ac.jp/~shige-o/tools/

Contact: ishii@is.aist-nara.ac.jp

1 INTRODUCTION

Gene expression profiling, using DNA microarrays, provides high throughput investigation of gene expressions by simultaneously measuring the expression of thousands of genes under a certain experimental condition. Gene expression profiling has been used in numerous studies over a broad range of biological disciplines. In clinical studies on cancer classification, e.g. Golub *et al.* (1999) distinguished between two leukemia subtypes, acute myeloid leukemia and acute lymphoblastic leukemia, by comparing the expression of 'predictor genes'. Their methodologies on class discovery and class prediction have been applied in a number of studies examining expression changes underlying various clinical phenomena. Unknown effects of a specific therapy were estimated by comparing gene expression profiles before and after the therapy (Perou *et al.*, 2000). Gene expression profile analyses were also effective in cancer prognosis prediction, even when morphological or immunohistological study was difficult (Alizadeh *et al.*, 2000; Kihara *et al.*, 2001; Pomeroy *et al.*, 2002; Shipp *et al.*, 2002; van't Veer *et al.*, 2002). In addition, expression profile analyses successfully identified genes relevant to a certain diagnosis or therapy (Takemasa *et al.*, 2001; Muro *et al.*, 2003). In these studies, various multivariate analysis methods have played crucial roles. Clustering, e.g. hierarchical clustering, is a popular unsupervised classification analysis, which has mainly been applied to class discovery problems (Eisen *et al.*, 1998).

In order to extract underlying biological reality based on gene expression profile analyses, it is necessary to discard various artifacts, such as noise and fluctuations that occur through the acquisition and normalization of data. Suspicious values are usually regarded as missing values, because they may be detrimental to analyses further. Existing multivariate analyses for expression profile data, however, often have difficulty with the treatment of missing values. Different methods of treating missing values may lead to different results. Although the handling of missing values is thus very important, researchers have often been unaware of this issue.

As an example, hierarchical clustering (Eisen *et al.*, 1998) constructs gene clusters or sample clusters based on the distance between two gene expression profiles (vectors) or

*To whom correspondence should be addressed.

two sample expression vectors, respectively. The distance measurement on missing values, however, is problematic. Existing hierarchical clustering software, such as 'Cluster' (Eisen *et al.*, 1998), defines the distance between vectors with missing values, by just ignoring the missing dimensions. Since ignoring dimensions is identical to assuming that expression levels are the same in two vectors, the distance between vectors with missing values tends to be smaller than that between vectors without missing values. Therefore, a cluster of genes with a lot of missing values is often obtained as a result. Support vector machine (SVM) classifier (Brown *et al.*, 2000), a popular multivariate supervised classification method, also encounters a similar problem in defining distance. Moreover, many multivariate statistical analyses, like principal component analysis (PCA) (Raychaudhuri *et al.*, 2000) and singular value decomposition (SVD) (Alter *et al.*, 2000), cannot be applied to data with missing values. Thus, in order to avoid improper analyses, missing value estimation is an important preprocess.

There are several simple ways to deal with missing values such as deleting an expression vector with missing values from further analysis, imputing missing values to zero, or imputing missing values of a certain gene (sample) to the sample (gene) average (Alizadeh *et al.*, 2000). On the other hand, Troyanskaya *et al.* (2001) proposed two advanced estimation methods for missing values in expression profiles. One method is based on K-nearest neighbor (KNNimpute), and the other is based on SVD (SVDimpute). Troyanskaya *et al*. evaluated their performance using various microarray data sets and reported that the two advanced methods performed better than the above-mentioned simple methods. The estimation ability of these advanced methods depends on important model parameters, such as the K-value in KNNimpute and the number of eigenvectors in SVDimpute. There is no theoretical way, however, to determine these parameters appropriately.

In this paper, we propose a new missing value estimation method based on Bayesian PCA (BPCA) (Bishop, 1999). Although the methodology that a probabilistic model and latent variables are estimated simultaneously within the framework of Bayes inference is not new in principle, actual BPCA implementation that makes it possible to estimate arbitrary missing variables is new in terms of statistical methodology. We evaluated the method by comparing it to KNNimpute and SVDimpute (Troyanskaya *et al.*, 2001), using various microarray data sets, and showed marked improvement in estimation performance. In addition, the model parameter is automatically determined in this BPCA method. Therefore, our BPCA method can be easily used by medical and biological scientists to analyze gene expression data.

2 SYSTEM AND METHODS

A whole data set of gene expression profiles is represented by a numerical $(D \times N)$ matrix $\boldsymbol{Y}$, where N is the number of genes and D is the number of samples. $\boldsymbol{Y}$ is called an expression matrix. The (i, j) component of the matrix, y_{ij}, denotes the expression level of the j-th gene in the i-th sample, which is typically a logarithm of the expression ratio between the control and the objective samples, in the case of cDNA microarray data. The i-th row vector and the j-th column vector of the matrix are called the expression vector of the i-th sample and the expression vector of the j-th gene, respectively.

2.1 BPCA

The missing value estimation method based on BPCA consists of three elementary processes. They are (1) principal component (PC) regression, (2) Bayesian estimation, and (3) an expectation–maximization (EM)-like repetitive algorithm. Below, we describe each of these processes.

2.2 PC regression

For the time being, we consider a situation where there is no missing value. PCA represents the variation of D-dimensional gene expression vectors $\boldsymbol{y}$ as a linear combination of principal axis vectors $\boldsymbol{w}_l$ $(1 \leq l \leq K)$ whose number is relatively small $(K < D)$:

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

The linear coefficients $x_l (1 \leq l \leq K)$ are called factor scores. ϵ denotes the residual error. Using a specifically determined number K, PCA obtains x_l and $\boldsymbol{w}_l$ such that the sum of squared error $\|\epsilon\|^2$ over the whole data set $\boldsymbol{Y}$ is minimized.

When there is no missing value, x_l and $\boldsymbol{w}_l$ are calculated as follows. A covariance matrix $\boldsymbol{S}$ for the expression vectors $\boldsymbol{y}_i$ $(1 \leq i \leq N)$ is given by

$$\boldsymbol{S} = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{y}_i - \boldsymbol{\mu})(\boldsymbol{y}_i - \boldsymbol{\mu})^{\mathsf{T}},$$

where $\boldsymbol{\mu}$ is the mean vector of $\boldsymbol{y}$: $\boldsymbol{\mu} \stackrel{\text{def}}{=} (1/N) \sum_{i=1}^{N} \boldsymbol{y}_i$. T denotes the transpose of a vector or a matrix. For description convenience, $\boldsymbol{Y}$ is assumed to be row-wisely normalized by a preprocess, so that $\boldsymbol{\mu} = \boldsymbol{0}$ holds. With this normalization, the result by PCA is identical to that by SVD.

Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_D$ and $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_D$ denote the eigenvalues and the corresponding eigenvectors, respectively, of $\boldsymbol{S}$. We also define the l-th principal axis vector by $\boldsymbol{w}_l = \sqrt{\lambda_l}\boldsymbol{u}_l$. With these notations, the l-th factor score for an expression vector $\boldsymbol{y}$ is given by $x_l = (\boldsymbol{w}_l/\lambda_l)^{\mathsf{T}} \boldsymbol{y}$.

Now we assume the existence of missing values. In PC regression, the missing part $\boldsymbol{y}^{\text{miss}}$ in the expression vector $\boldsymbol{y}$ is estimated from the observed part $\boldsymbol{y}^{\text{obs}}$ by using the PCA result. Let $\boldsymbol{w}_l^{\text{obs}}$ and $\boldsymbol{w}_l^{\text{miss}}$ be parts of each principal axis $\boldsymbol{w}_l$, corresponding to the observed and missing parts, respectively, in $\boldsymbol{y}$. Similarly, let $\boldsymbol{W} = (\boldsymbol{W}^{\text{obs}}, \boldsymbol{W}^{\text{miss}})$ where $\boldsymbol{W}^{\text{obs}}$ or $\boldsymbol{W}^{\text{miss}}$ denotes a matrix whose column vectors are $\boldsymbol{w}_1^{\text{obs}}, \ldots, \boldsymbol{w}_K^{\text{obs}}$ or $\boldsymbol{w}_1^{\text{miss}}, \ldots, \boldsymbol{w}_K^{\text{miss}}$, respectively.

Factor scores $\boldsymbol{x} = (x_1, \ldots, x_K)$ for the expression vector $\boldsymbol{y}$ are obtained by minimization of the residual error:

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

This is a well-known regression problem, and the least square solution is given by

$$\boldsymbol{x} = (\boldsymbol{W}^{\mathrm{obs}\mathsf{T}} \boldsymbol{W}^{\mathrm{obs}})^{-1} \boldsymbol{W}^{\mathrm{obs}\mathsf{T}} \boldsymbol{y}^{\mathrm{obs}}.$$

Using $\boldsymbol{x}$, the missing part is estimated as

$$\boldsymbol{y}^{\mathrm{miss}} = \boldsymbol{W}^{\mathrm{miss}} \boldsymbol{x}. \tag{2}$$

In the PC regression above, $\boldsymbol{W}$ should be known beforehand. Later, we will discuss the way to determine the parameter.

2.3 Bayesian estimation

A parametric probabilistic model, which is called probabilistic PCA (PPCA), has been proposed recently (Tipping and Bishop, 1999). The probabilistic model is based on the assumption that the residual error ϵ and the factor scores x_l $(1 \leq l \leq K)$ in Equation (1) obey normal distributions:

$$p(\boldsymbol{x}) = \mathcal{N}_K(\boldsymbol{x}|\boldsymbol{0}, \mathbf{I}_K),$$
$$p(\epsilon) = \mathcal{N}_D(\epsilon|\boldsymbol{0}, (1/\tau)\mathbf{I}_D),$$

where $\mathcal{N}_K(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a K-dimensional normal distribution for $\boldsymbol{x}$, whose mean and covariance are $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, respectively. $\mathbf{I}_K$ is a $(K \times K)$ identity matrix and τ is a scalar inverse variance of ϵ. In this PPCA model, a complete log-likelihood function is written as:

$$\begin{aligned}\ln p(\boldsymbol{y}, \boldsymbol{x}|\boldsymbol{\theta}) &\equiv \ln p(\boldsymbol{y}, \boldsymbol{x}|\boldsymbol{W}, \boldsymbol{\mu}, \tau) \\ &= -\frac{\tau}{2}\|\boldsymbol{y} - \boldsymbol{W}\boldsymbol{x} - \boldsymbol{\mu}\|^2 - \frac{1}{2}\|\boldsymbol{x}\|^2 + \frac{D}{2}\ln\tau \\ &\quad - \frac{K+D}{2}\ln 2\pi,\end{aligned}$$

where $\boldsymbol{\theta} \equiv \{\boldsymbol{W}, \boldsymbol{\mu}, \tau\}$ is the parameter set. Since the maximum likelihood (ML) estimation of the PPCA is identical to PCA, PPCA is a natural extension of PCA to a probabilistic model.

We introduce here a Bayesian estimation method for PPCA, which was originally proposed by Bishop (1999). Bayesian estimation obtains the posterior distribution of $\boldsymbol{\theta}$ and $\boldsymbol{X}$, according to the Bayes theorem:

$$p(\boldsymbol{\theta}, \boldsymbol{X}|\boldsymbol{Y}) \propto p(\boldsymbol{Y}, \boldsymbol{X}|\boldsymbol{\theta})p(\boldsymbol{\theta}). \tag{3}$$

$p(\boldsymbol{\theta})$ is called a prior distribution, which denotes a priori preference for parameter $\boldsymbol{\theta}$. The prior distribution is a part of the model and must be defined before estimation.

We assume conjugate priors for τ and $\boldsymbol{\mu}$, and a hierarchical prior for $\boldsymbol{W}$, namely, the prior for $\boldsymbol{W}$, $p(\boldsymbol{W}|\tau, \boldsymbol{\alpha})$, is parameterized by a hyperparameter $\boldsymbol{\alpha} \in \mathbb{R}^K$.

$$p(\boldsymbol{\theta}|\boldsymbol{\alpha}) \equiv p(\boldsymbol{\mu}, \boldsymbol{W}, \tau|\boldsymbol{\alpha}) = p(\boldsymbol{\mu}|\tau)p(\tau)\prod_{j=1}^{K} p(\boldsymbol{w}_j|\tau, \alpha_j),$$
$$p(\boldsymbol{\mu}|\tau) = \mathcal{N}(\boldsymbol{\mu}|\overline{\boldsymbol{\mu}}_0, (\gamma_{\mu 0}\tau)^{-1}\mathbf{I}_m),$$
$$p(\boldsymbol{w}_j|\tau, \alpha_j) = \mathcal{N}(\boldsymbol{w}_j|\boldsymbol{0}, (\alpha_j\tau)^{-1}\mathbf{I}_m),$$
$$p(\tau) = \mathcal{G}(\tau|\bar{\tau}_0, \gamma_{\tau 0}).$$

$\mathcal{G}(\tau|\bar{\tau}, \gamma_\tau)$ denotes a Gamma distribution with hyperparameters $\bar{\tau}$ and γ_τ:

$$\mathcal{G}(\tau|\bar{\tau}, \gamma_\tau) \equiv \frac{(\gamma_\tau \bar{\tau}^{-1})^{\gamma_\tau}}{\Gamma(\gamma_\tau)} \exp\left[-\gamma_\tau \bar{\tau}^{-1}\tau + (\gamma_\tau - 1)\ln\tau\right],$$

where $\Gamma(\cdot)$ is a Gamma function.

The variables used in the above priors, $\gamma_{\mu 0}, \overline{\boldsymbol{\mu}}_0, \gamma_{\tau 0}$ and $\bar{\tau}_0$ are deterministic hyperparameters that define the prior. Their actual values should be given before the estimation. We set $\gamma_{\mu 0} = \gamma_{\tau 0} = 10^{-10}$, $\overline{\boldsymbol{\mu}}_0 = \boldsymbol{0}$ and $\bar{\tau}_0 = 1$, which corresponds to an almost non-informative prior.

Assuming the priors and given a whole data set $\boldsymbol{Y} = \{\boldsymbol{y}\}$, the type-II ML hyperparameter $\boldsymbol{\alpha}_{\mathrm{ML-II}}$ and the posterior distribution of the parameter, $q(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{\alpha}_{\mathrm{ML-II}})$, are obtained by Bayesian estimation.

The hierarchical prior $p(\boldsymbol{W}|\boldsymbol{\alpha}, \tau)$, which is called an automatic relevance determination (ARD) prior, has an important role in BPCA. The j-th principal axis $\boldsymbol{w}_j$ has a Gaussian prior, and its variance $1/(\alpha_j\tau)$ is controled by a hyperparameter α_j which is determined by type-II ML estimation from the data. When the Euclidian norm of the principal axis, $\|\boldsymbol{w}_j\|$, is small relatively to the noise variance $1/\tau$, the hyperparameter α_j gets large and the principal axis $\boldsymbol{w}_j$ shrinks nearly to be $\boldsymbol{0}$. Thus, redundant principal axes are automatically supressed.

2.4 EM-like repetitive algorithm

If we know the true parameter $\boldsymbol{\theta}_{\mathrm{true}}$, the posterior of the missing values is given by

$$q(\boldsymbol{Y}^{\mathrm{miss}}) = p(\boldsymbol{Y}^{\mathrm{miss}}|\boldsymbol{Y}^{\mathrm{obs}}, \boldsymbol{\theta}_{\mathrm{true}}),$$

which produces equivalent estimation to the PC regression. Here, $p(\boldsymbol{Y}^{\mathrm{miss}}|\boldsymbol{Y}^{\mathrm{obs}}, \boldsymbol{\theta}_{\mathrm{true}})$ is obtained by marginalizing the likelihood (3) with respect to the observed variables $\boldsymbol{Y}^{\mathrm{obs}}$. If we have the parameter posterior $q(\boldsymbol{\theta})$ instead of the true parameter, the posterior of the missing values is given by

$$q(\boldsymbol{Y}^{\mathrm{miss}}) = \int \mathrm{d}\boldsymbol{\theta}\, q(\boldsymbol{\theta})\, p(\boldsymbol{Y}^{\mathrm{miss}}|\boldsymbol{Y}^{\mathrm{obs}}, \boldsymbol{\theta}),$$

which corresponds to the Bayesian PC regression. Since we do not know the true parameter naturally, we conduct the BPCA. Although the parameter posterior $q(\boldsymbol{\theta})$ can be easily obtained by the Bayesian estimation when a complete data set $\boldsymbol{Y}$ is available, we assume that only a part of $\boldsymbol{Y}$, $\boldsymbol{Y}^{\mathrm{obs}}$, is observed

and the rest $\boldsymbol{Y}^{\mathrm{miss}}$ is missing. In that situation, it is required to obtain $q(\boldsymbol{\theta})$ and $q(\boldsymbol{Y}^{\mathrm{miss}})$ simultaneously.

We use a variational Bayes (VB) algorithm (Attias, 1999), in order to execute Bayesian estimation for both model parameter $\boldsymbol{\theta}$ and missing values $\boldsymbol{Y}^{\mathrm{miss}}$. Although the VB algorithm resembles the EM algorithm that obtains ML estimators for $\boldsymbol{\theta}$ and $\boldsymbol{Y}^{\mathrm{miss}}$, it obtains the posterior distributions for $\boldsymbol{\theta}$ and $\boldsymbol{Y}^{\mathrm{miss}}$, $q(\boldsymbol{\theta})$ and $q(\boldsymbol{Y}^{\mathrm{miss}})$, by a repetitive algorithm.

The VB algorithm is implemented as follows: (a) the posterior distribution of missing values, $q(\boldsymbol{Y}^{\mathrm{miss}})$, is initialized by imputing each of the missing values to gene-wise average; (b) the posterior distribution of the parameter $\boldsymbol{\theta}$, $q(\boldsymbol{\theta})$, is estimated using the observed data $\boldsymbol{Y}^{\mathrm{obs}}$ and the current posterior distribution of missing values, $q(\boldsymbol{Y}^{\mathrm{miss}})$; (c) the posterior distribution of the missing values, $q(\boldsymbol{Y}^{\mathrm{miss}})$, is estimated using the current $q(\boldsymbol{\theta})$; (d) the hyperparameter $\boldsymbol{\alpha}$ is updated using both of the current $q(\boldsymbol{\theta})$ and the current $q(\boldsymbol{Y}^{\mathrm{miss}})$; (e) repeat (b)–(d) until convergence.

The VB algorithm has been proved to converge to a locally optimal solution (Sato, 2001). Although the convergence to the global optimum is not guaranteed, the VB algorithm for BPCA almost always converges to a single solution practically. This is probably because the objective function of BPCA has a simple landscape. As a consequence of the VB algorithm, therefore, $q(\boldsymbol{\theta})$ and $q(\boldsymbol{Y}^{\mathrm{miss}})$ are expected to approach the global optimal posteriors.

Then, the missing values in the expression matrix are imputed to the expectation with respect to the estimated posterior distribution:

$$\hat{\boldsymbol{Y}}^{\mathrm{miss}} = \int \boldsymbol{Y}^{\mathrm{miss}} q(\boldsymbol{Y}^{\mathrm{miss}})\, \mathrm{d}\boldsymbol{Y}^{\mathrm{miss}}. \tag{4}$$

2.5 SVDimpute

With respect to the above three elementary processes, SVDimpute (Troyanskaya *et al.*, 2001) is a method incorporating the first process and the ML estimation based on the EM algorithm, because SVD is identical to standard PCA when applied to a matrix normalized so that the row-wise mean is zero. Therefore, the most important advance of BPCA, in comparison to SVDimpute, is the existence of the second process, i.e. the Bayesian estimation using the ARD prior. An SVD-based imputation method was also proposed and described in detail (Hastie *et al.*, 1999).

2.6 KNNimpute

In order to estimate a missing value y_{ih} in the i-th gene expression vector $\boldsymbol{y}_i$ by KNNimpute (Troyanskaya *et al.*, 2001), we first select K genes whose expression vectors are similar to $\boldsymbol{y}_i$. Next, the missing value is estimated as the average of the corresponding entries in the selected K expression vectors.

The similarity measure $s_i(\boldsymbol{y}_j)$ between two expression vectors $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ is defined by the reciprocal of the Euclidian distance calculated over observed components in $\boldsymbol{y}_i$. When there are other missing values in $\boldsymbol{y}_i$ and/or $\boldsymbol{y}_j$, their treatment requires some heuristics. Following (Troyanskaya *et al.*, 2001), we define the measure as follows:

$$1/s_i(\boldsymbol{y}_j) = \sum_{h \in O_i \cap O_j} (y_{ih} - y_{jh})^2, \tag{5}$$

$$O_i = \{h \mid \text{the } h\text{-th component of } \boldsymbol{y}_i \text{ is observed}\}.$$

The missing entry y_{ih} is estimated as average weighted by the similarity measure:

$$\hat{y_{ih}} = \frac{\sum_{j \in I_{Kih}} s_i(\boldsymbol{y}_j) y_{jh}}{\sum_{j \in I_{Kih}} s_i(\boldsymbol{y}_j)}, \tag{6}$$

where I_{Kih} is the index set of K-nearest neighbor genes of the i-th gene, and if y_{jh} is missing the j-th gene is excluded from I_{Kih}. Note that KNNimpute has no theoretical criteria for selecting the best K-value and the K-value has to be determined empirically.

3 RESULTS AND DISCUSSION

3.1 Data sets

Spellman *et al.* (1998) placed a cDNA microarray data set relevant to the yeast cell-cycle at the URL http://genome-www.stanford.edu/cellcycle/data/rawdata/ as a complement. This data set consists of three parts, which are relevant to alpha factor (A-part), elutriation (E-part), *cdc15*, and *cdc28* (C-part). We first used samples in the A-part (18 samples) and the E-part (14 samples) to prepare test data sets. Each sample represents relative expression levels of 6178 genes, and 4304 genes have no missing value in the A- and E-parts. Therefore, the complete expression matrix is composed of 4304 genes. We prepared three test data sets: (data A), (data E) and (data A + E), from the complete expression matrix. The C-part samples were used for examining the effects of additional samples (see Section 3.4).

We also prepared a test data set (data A + E + C) by adding the C-part samples to (data A + E).

Takemasa *et al.* (2001) obtained original cDNA microarray data relevant to human colorectal cancer (CRC). Clinical materials of the data consist of 205 primary CRCs that include 127 non-metastatic primary CRCs, 54 metastatic primary CRCs to the liver and 24 metastatic primary CRCs to distant organs exclusive of the liver, and 12 normal colonic epithelia that were histopathologically confirmed to be free of cancer. Each sample expression vector represents logarithm-transformed ratios between the expression levels in the objective sample and that in the control reference using cDNA microarrays specialized for CRC, by selecting genes that were preferentially expressed in colorectal carcinoma tissue. As members of the complete expression matrix, we selected 758 genes in 4608 genes, and a test data set (data I) was prepared.

Using these four data sets, (data A), (data E), (data A + E) and (data I), we examined the estimation ability for missing values.

In order to evaluate the performance of missing value estimation methods, we introduced artificial missing entries to a complete (i.e. without missing values) expression matrix. The artificial missing entries were introduced in two different ways:

Rate-based way Randomly select a specific percentage of the entries in the complete expression matrix, and remove them.

Histogram-based way Obtain a histogram of column-wise numbers of missing entries in the original expression matrix. Then, remove entries from the complete expression matrix so that the histogram of the artificial missing entries is similar to the histogram of the original missing entries.

When 5% artificial missing entries are introduced to (data I) in the rate-based way, the test data set is denoted by (data I, 5%).

The performance of the missing value estimation is evaluated by normalized root mean squared error (NRMSE):

$$\mathrm{NRMSE} = \sqrt{\frac{\mathrm{mean}[(y_{\mathrm{guess}} - y_{\mathrm{answer}})^2]}{\mathrm{variance}[y_{\mathrm{answer}}]}}, \qquad (7)$$

where the mean and the variance are calculated over missing entries in the whole matrix. We know y_{answer} because the missing entries are artificial. When the estimation is accurate, NRMSE approaches its minimum value 0.0. When the estimation is equivalent to a random guess, which occurs either when the estimation is too poor or when the noise involved is too large, NRMSE approaches a value of 1.0.

3.2 *K*-value selection

Both BPCA and SVDimpute depend on the number of principal axes (eigenvectors), K, and KNNimpute depends on the number of neighbors, K (see Section 2.6). Since these K-values describe similar parameters, we use the same symbol. In order to measure how the estimation ability depends on the value of K, we applied BPCA, SVDimpute and KNNimpute to the test data sets, (data A + E) and (data I), and calculated NRMSE with various K-values.

Figure 1 shows the results for (data A + E, 5%) and (data I, 5%). BPCA produces better results than KNNimpute or SVDimpute at the optimal K-value for each method. BPCA exhibits its best results with $K = D - 1$, where D is the number of samples. SVDimpute and BPCA show similar results when K is small, because they employ the same PC regression process. When K is larger, however, BPCA exhibits much better results than SVDimpute. This is due to the ARD prior, because the main difference of BPCA from SVDimpute is its existence. When $K = 0$, the imputation by BPCA or SVDimpute is identical to that based on gene-wise average, and therefore the results are poor.

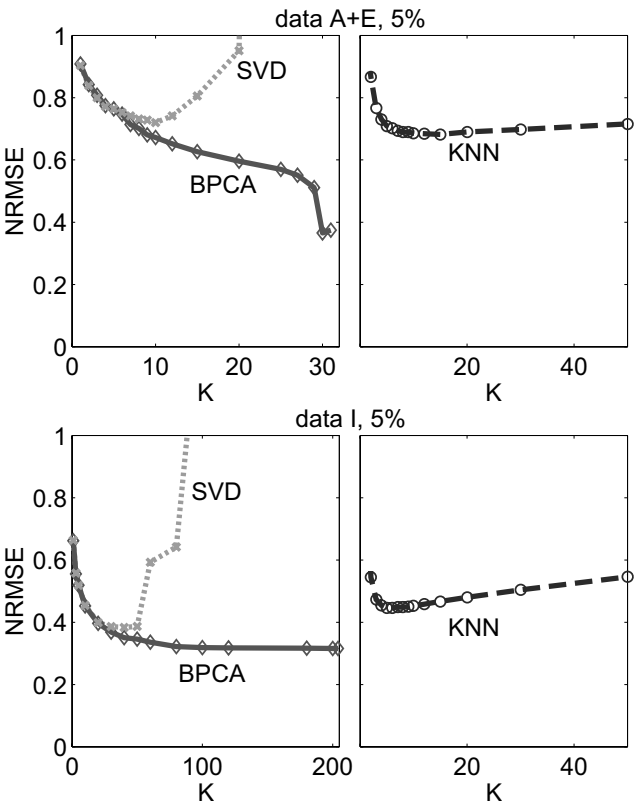

Fig. 1. Estimation ability (NRMSE) by BPCA, SVD and KNN with various K-values. (top panel): Application to (data A + E, 5%). (bottom panel): Application to (data I, 5%).

In Figure 1, we see that BPCA exhibits different NRMSE curves in the (data A + E) case and the (data I) case, with respect to the optimal K-value. For (data I), the NRMSE curve becomes almost flat between $K = 100$ and $K = 204$, because the principal axes corresponding to the eigenvalues exceeding $K = 100$ degenerated so that their lengths became almost zero. For (data A + E), on the other hand, almost none of the axes degenerated, except for the 31st one. From $K = 1$ to $K = 30$, therefore, each additional axis improved the estimation ability. Although, the improvement by adding the 30th axis was apparently large, we consider this is a special phenomenon for this data set. This phenomenon implies the importance of setting $K = D - 1$ if we do not have a priori knowledge on the data set.

Accordingly, we can safely use $K = D-1$ for every data set in BPCA. If the effective dimension of the data set is smaller than the K-value, the ARD prior automatically reduces the redundant principal axes. Therefore, in our BPCA method, there is no need to tune the K-value in advance.

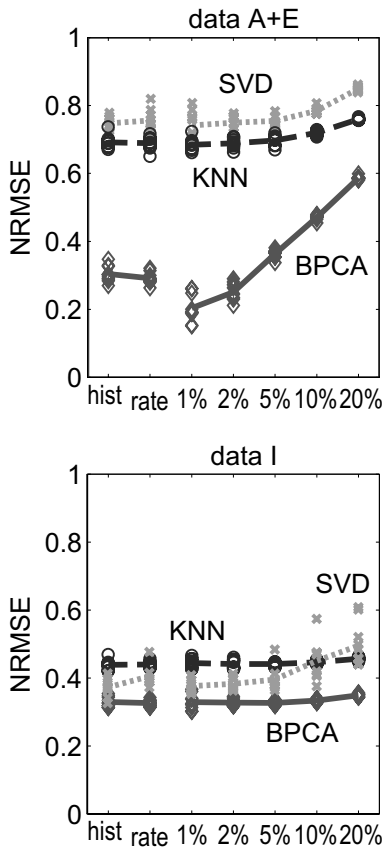

Fig. 2. Estimation ability (NRMSE) by BPCA, SVD and KNN for various percentages of missing entries. For KNNimpute and SVDimpute, the best results by tuning the parameter K are shown. (Upper): Application to (data A + E). (Lower): Application to (data I). For (data A + E), the histogram-based way introduced 2.96% missing entries ('hist') and the same percentage of missing entries were introduced in the rate-based way ('rate'). Similarly, the histogram- and the rate-based way introduced 3.04% missing entries to (data I). In addition, the rate-based way by introducing 1, 2, 5, 10 or 20% is evaluated. In each case, we repeated 10 times by varying the set of missing entries, and the figure shows those 10 results.

3.3 Type and percentage of missing entries

Figure 2 shows the results for various percentages of missing entries. The comparison of two types of missing entry introduction is also included. For SVDimpute, the K-value was set at $K = 10$ in (data A + E) and $K = 30$ in (data I), which exhibited the best results. For BPCA, we set $K = D - 1$, where $D = 32$ in (data A + E) and $D = 206$ in (data I). For KNNimpute, the K-value was set at the best value between 1 and 50 in each data condition.

By comparing 'hist' and 'rate' in Figure 2, we find that the missing value estimation accuracy does not much depend on the two types of the introduction way, if the percentage of missing entries is the same. By comparing the results for various percentages of missing entries, however, we find that the percentage of missing entries affects the estimation accuracy especially for (data A + E). For (data I), large amount of missing entries do not degrade the estimation performance, probably because there are a lot of samples in the data set. The performance advantage of the BPCA to the KNNimpute for (data I) was not so larger than for (data A + E), probably because there is difference in the levels of backgound noise between the both data sets.

The variance of NRMSE within 10 times repetition is smaller when the number of missing entries is large than when the number is small. This is attributed to the law of large numbers. SVDimpute, however, sometimes produced very poor results and then the variance is large.

3.4 Effect of additional information

Figure 3 shows the estimation performance for (data A, 5%) and (data E, 5%) when additional samples were available. From this figure, we find that SVDimpute and BPCA improved the estimation ability when utilizing the additional information, while KNNimpute degraded the performance in this case. It should be noted that the C-part data used in conditions 'A-AEC' and 'E-AEC' contains a lot of missing entries, and such 'dirty' data may be harmful for the selection process of K-nearest neighbors in KNNimpute.

Figure 4 shows the missing value estimation ability for various sample numbers: 10, 20, . . . , 200. The data sets were made by clipping certain numbers of samples from (data I, 5%). As the number of samples increased the information useful for the imputation increased, which is the reason for the improvement by SVDimpute and BPCA. The performance by KNNimpute, however, did not improve much, possibly because the similarity measure used in the method was not very suitable for cases with a large number of missing values. When the number of samples was small, KNNimpute exhibited better performance than the others.

Figure 5 shows the estimation ability when gene expression vectors with a lot of missing values were used as additional information. A fixed set of missing entries in (data A + E), 5%) or (data I, 5%) were estimated by KNNimpute, SVD impute and BPCA. Additional sets of gene expression vectors, 'A' including many (20%) missing values and 'B' including extremely many (40%) missing values, were prepared for (data A + E), 5%) and (data I, 5%) from the corresponding original expression matrices. By adding such 'dirty' gene expression vectors, the missing value estimation accuracy by BPCA improved or did not degrade. Although the estimation accuracy by SVDimpute did not degrade by additional gene expression vectors, the algorithm often diverged when K and/or the number of missing values was large. KNNimpute sometimes degraded by adding the extremely 'dirty' data set, possibly due to the similarity measure for determining KNN. If there is a gene whose expression levels are missing for all samples, for example, the similarity measure [Equation (5)]

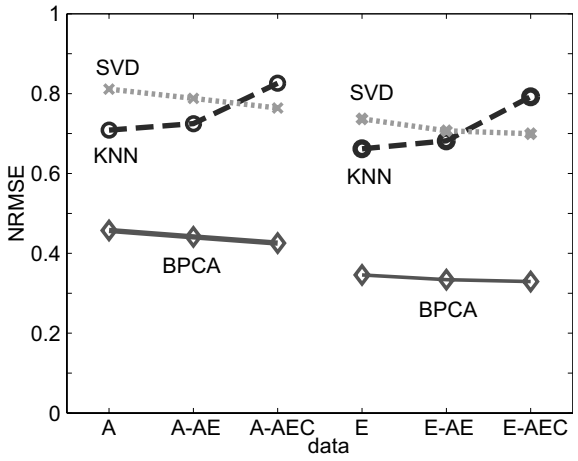

Fig. 3. Estimation ability (NRMSE) by BPCA, SVDimpute and KNNimpute for (data A, 5%) and (data E, 5%). The other data sets were used as additional information (samples). For example, in the condition 'A-AE', the missing values in (data A, 5%) were estimated by using (data E, 5%) as additional information. For KNNimpute and SVDimpute, the best results by tuning the parameter K are shown. In BPCA, $K = D - 1$ was used for every condition.

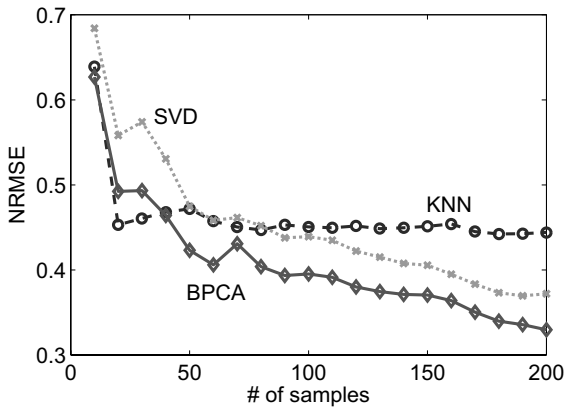

Fig. 4. Estimation ability (NRMSE) by BPCA, SVDimpute and KNNimpute. For SVDimpute and KNNimpute, the best results by tuning the parameter K are shown.

says that this gene is the nearest from every gene even though there is no information available for the imputation.

4 CONCLUSION

We found that BPCA performs much better than the existing methods for missing value estimation. This advantage is prominent especially when the number of samples is large (see Fig. 4). We assumed a PPCA model for expression profile data. The model assumption, however, may introduce a bias in the estimation as mentioned by (Troyanskaya *et al.*, 2001), because the assumed model may not account well for the actual data generation process. Since BPCA assumes

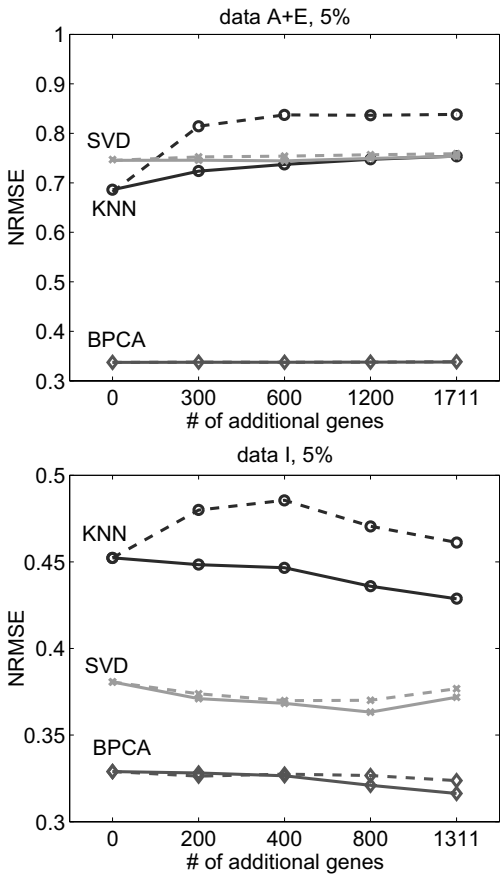

Fig. 5. Estimation ability (NRMSE) by BPCA, SVD and KNN with various numbers of additional 'dirty' genes expression vectors. Solid and dash lines denote that the additional genes have many (20%) or extremely many (40%) missing values, respectively.

only a global covariance structure, the estimation with BPCA may not be accurate if genes have dominant local similarity structures. In such a case, KNNimpute will be suitable. We consider, however, that any method, parametric or non-parametric, assumes a model explicitly or implicitly and thus cannot be completely free from introducing biases. Since the estimation error is small in BPCA, we suggest that the bias introduced by BPCA is small in comparison with those introduced by the existing methods.

Our study assumes that missing values in an expression matrix occur randomly and independently of other features in the matrix (Hastie *et al.*, 2001). This assumption may not be valid in gene expression profile data. When control, objective, or both types of expression levels are too small, the corresponding data are treated as missing. In this case, the reason for calling data missing is dependent on the character of the gene. Due to this dependence, the estimation performance might be either little or much worse for real missing values than for artificial missing values. Missing values can be classified according to the reasons why they are treated as missing.

In addition, there are cases where the existence of missing values is in itself meaningful. In clinical fields, the data may become missing by the dropout of a patient from a study (Shih, 2002). There are some characteristic types of flaws on DNA microarrays (Troyanskaya *et al.*, 2001). Since such missing data have a certain reason to be missing, the estimated missing value should correlate to the reason. Within a Bayesian framework, such reasons may be incorporated into a model as a prior distribution, enabling a more accurate estimation.

We recommend using the BPCA missing value estimation prior to further gene expression profile analyses. We also recommend the following considerations when using the BPCA estimation method:

- Do not neglect samples or genes with many missing values, before the missing value estimation. The BPCA method uses the entire information in a given data set, even though the information on a sample or a gene with many missing values is relatively small. The estimation ability is improved by including these samples or genes (see Figs 3, 4 and 5). After the missing value estimation, however, it might be prudent to omit unreliable samples and/or genes that originally contained many missing values.
- Do not normalize the expression matrix before the missing value estimation process. According to our study (data not shown), row-wise or column-wise normalization always degrades the missing value estimation ability.

Although our BPCA method was examined using cDNA microarray data, applications to oligonucleotide array data, reverse transcription–polymerase chain reaction data, and others are straightforward. Moreover, our method can be applied to various bioinformatics data.

ACKNOWLEDGEMENTS

This work was supported by Grant-in-Aid for Scientific Research on Priority Areas (C) 'Genome Information Science' from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

REFERENCES

Alizadeh,A.A., Eisen,M.B., Davis,R.E., Ma,C., Lossos,I.S., Rosenwald,A., Boldrick,J., Sabet,H., Tran,T., Yu,X. *et al*. (2000) Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, **403**, 503–511.

Alter,O., Brown,P.O. and Botstein,D. (2000) Singular value decomposition for genome-wide expression data processing and modeling. *Proc. Natl Acad. Sci. USA*, **97**, 10101–10106.

Attias,H. (1999) Inferring parameters and structure of latent variable models by variational Bayes. In *Proceedings of Uncertainty in Artificial Intelligence*. pp. 21–30.

Bishop,C.M. (1999) Variational principal components. In *IEE Conference Publication on Artificial Neural Networks*. pp. 509–514.

Brown,M.P., Grundy,W.N., Lin,D., Cristianini,N., Sugnet,C.W., Furey,T.S., Ares,M.J. and Haussler,D. (2000) Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl Acad. Sci. USA*, **97**, 262–267.

Eisen,M.B., Spellman,P.T., Brown,P.O. and Botstein,D. (1998) Cluster analysis and display of genome-wide expression patterns. *Proc. Natl Acad. Sci. USA*, **95**, 14863–14868.

Golub,T.R., Slonim,D.K., Tamayo,P., Huard,C., Gaasenbeek,M., Mesirov,J.P., Coller,H., Loh,M.L., Downing,J.R., Caligiuri,M.A., Bloomfield,C.D. and Lander,E.S. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531–537.

Hastie,T., Alter,O., Sherlock,G., Eisen,M., Tibshirani,R., Botstein,D. and Brown,P. (1999) Imputation of missing values in DNA microarrays. Technical report Stanford University Statistics Department.

Hastie,T., Tibshirani,M. and Friedman,J. (2001) *The Elements of Statistical Learning*. Springer-Verlag, New York, USA.

Kihara,C., Tsunoda,T., Tanaka,T., Yamana,H., Furukawa,Y., Ono,K., Kitahara,O., Zembutsu,H., Yanagawa,R., Hirata,K., Takagi,T. and Nakamura,Y. (2001) Prediction of sensitivity of esophageal tumors to adjuvant chemotherapy by cDNA microarray analysis of gene-expression profiles. *Cancer Res.*, **61**, 6474–6479.

Muro,S., Takemasa,I., Oba,S., Matoba,R., Ueno,N., Maruyama,C., Yamashita, R., Sekimoto,M., Yamamoto,H., Nakamori,S. *et al*. (2003) Identification of expressed genes linked to malignancy of human colorectal carcinoma by parameteric clustering of quantitative expression data. *Genome Biol.*, **4**, R21.

Perou,C.M., Sorlie,T., Eisen,M.B., van de Rijn, M., Jeffrey,S.S., Rees,C.A., Pollack,J.R., Ross,D.T., Johnsen,H., Akslen,L.A. *et al*. (2000) Molecular portraits of human breast tumours. *Nature*, **406**, 747–752.

Pomeroy,S.L., Tamayo,P., Gaasenbeek,M., Sturla,L.M., Angelo,M., McLaughlin,M.E., Kim,J.Y., Goumnerova,L.C., Black,P.M., Lau, C. *et al*. (2002) Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, **415**, 436–442.

Raychaudhuri,S., Stuart,J.M. and Altman,R. (2000) Principal components analysis to summarize microarray experiments: application to sporulation time series. *Pac. Symp. Biocomput.*, 455–466.

Sato,M. (2001) On-line model selection based on the variational Bayes. *Neural Comput.*, **13**, 1649–1681.

Shih,W. (2002) Problems in dealing with missing data and informative censoring in clinical trials. *Curr. Control Trials Cardiovasc. Med.*, **3**, 4.

Shipp,M.A., Ross,K.N., Tamayo,P., Weng,A.P., Kutok,J.L., Aguiar,R.C., Gaasenbeek,M., Angelo,M., Reich,M., Pinkus,O.S. *et al.* (2002) Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nat. Med.*, **8**, 68–74.

Spellman,P.T., Sherlock,G., Zhang,M.Q., Iyer,V.R., Anders,K., Eisen,M.B., Brown,P.O., Botstein,D. and Futcher,B. (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, **9**, 3273–3297.

Takemasa,I., Higuchi,H., Yamamoto,H., Sekimoto,M., Tomita,N., Nakamori, S., Matoba,R., Monden,M. and Matsubara,K. (2001) Construction of preferential cDNA microarray specialized for human colorectal carcinoma: molecular sketch of colorectal cancer. *Biochem. Biophys. Res. Commun.*, **285**, 1244–1249.

Tipping,M.E. and and Bishop,C.M. (1999) Mixtures of probabilistic principal component analysers. *Neural Comput.*, **11**, 443–482.

Troyanskaya,O., Cantor,M., Sherlock,G., Brown,P., Hastie,T., Tibshirani, R., Botstein,D. and Altman,R. (2001) Missing value estimation methods for DNA microarrays. *Bioinformatics*, **17**, 520–525.

van't Veer,L.J., Dai,H., van de Vijver,M.J., He,Y.D., Hart,A.A., Mao,M., Peterse,H.L., van der Kooy, K., Marton,M.J., Witteveen,A.T. *et al*. (2002) Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, **415**, 530–536.