An Imputation-Consistency Algorithm for High-Dimensional Missing Data Problems and Beyond

Faming Liang, Bochao Jia, Jingnan Xue, Qizhai Li, Ye Luo*

November 6, 2016

Abstract

Missing data are frequently encountered in high-dimensional problems, but they are usually difficult to deal with using standard algorithms, such as the expectation-maximization (EM) algorithm and its variants. To tackle this difficulty, some problem-specific algorithms have been developed in the literature, but there still lacks a general algorithm. This work is to fill the gap: we propose a general algorithm for high-dimensional missing data problems. The proposed algorithm works by iterating between an imputation step and a consistency step. At the imputation step, the missing data are imputed conditional on the observed data and the current estimate of parameters; and at the consistency step, a consistent estimate is found for the minimizer of a Kullback-Leibler divergence defined on the pseudo-complete data. The consistency of the averaged estimate for the true parameter can be established under quite general conditions. The proposed algorithm is illustrated using high-dimensional Gaussian graphical models, high-dimensional variable selection, and a random coefficient model. The proposed algorithm has strong implications for big data computing: Based on it, we propose a general strategy to improve Bayesian computation for big data problems. The proposed algorithm also facilitates data integration from multiple sources, which plays an important role in big data analysis.

Keywords: EM Algorithm; Gaussian Graphical Model; Gibbs Sampler; Random Coefficient Model; Variable Selection.

1 Introduction

Missing data are frequently encountered in both low and high-dimensional data, where low and high refer to that the number of variables is smaller or larger than the sample size, respectively. For example, the microarray data is usually considered as high-dimensional, where the number of genes can be much larger than the number of samples. Missing values can appear in microarray data due to various factors.
such as scratches on slides, spotting problems, experimental errors, etc. In some microarray experiments, missing values can occur for more than 90% of the genes (Ouyang et al., 2004). Simply deleting the samples or genes for which missing values occur can lead to a significant loss of information of the data. How to deal with missing data has been a long-standing problem in statistics.

For low-dimensional problems, the missing data can be dealt with using the EM algorithm (Dempster et al., 1977) or its variants. Let $X_{\text{obs}} = (X_{1\text{obs}}, X_{2\text{obs}}, \ldots, X_{n\text{obs}})$ denote the observed incomplete data, where $n$ denotes the sample size. Let $X_{\text{mis}} = (X_{1\text{mis}}, X_{2\text{mis}}, \ldots, X_{n\text{mis}})$ denote the missing data, and let $X = (X_{\text{obs}}, X_{\text{mis}})$ denote the complete data. Let $\theta$ denote the vector of unknown parameters, and let $f(X|\theta)$ denote the likelihood function of the complete data. Then the maximum likelihood estimate (MLE) of $\theta$ can be determined by maximizing the marginal likelihood of the observed data,

$$f(X_{\text{obs}}|\theta) = \int f(X_{\text{obs}}, X_{\text{mis}}|\theta) h(x_{\text{mis}}|\theta, X_{\text{obs}}) dx_{\text{mis}},$$

where $h(x_{\text{mis}}|\theta, X_{\text{obs}})$ denotes the predictive density of the missing data. The EM algorithm seeks to maximize the marginal likelihood function by iteratively applying the following two steps:

- **E-step**: Calculate the expected value of the log-likelihood function with respect to the predictive distribution of the missing data given the current estimate of $\theta^{(t)}$, i.e.,

$$Q(\theta|\theta^{(t)}) = \int \log f(X_{\text{obs}}, x_{\text{mis}}|\theta) h(x_{\text{mis}}|\theta^{(t)}, X_{\text{obs}}) dx_{\text{mis}}.$$

- **M-step**: Find a value of $\theta$ that maximizes the quantity $Q(\theta|\theta^{(t)})$, i.e., set

$$\theta^{(t+1)} = \arg \max_\theta Q(\theta|\theta^{(t)}).$$

Dempster et al. (1977) showed that the marginal likelihood value increases with each iteration and, under fairly general conditions, it converges to a stationary value—a local or global maximum of the marginal likelihood. A rigorous study for the convergence is given by Wu (1983).

Both the E and M-steps of the EM algorithm can be rather complicated or even intractable. Meng and Rubin (1993) found that in many cases, the M-step is relatively simple when conditioned on some function of the parameters under estimation. Motivated by this observation, they introduced the expectation-conditional maximization (ECM) algorithm, which is to replace the M-step by a number of computationally simpler conditional maximization steps. Later, the EM algorithm was further speeded up by some other variants, such as the ECME algorithm (Liu and Rubin, 1994, He and Liu, 2012) and the PX-EM algorithm (Liu et al., 1998). When the E-step is analytically intractable, Wei and Tanner (1990) introduced the Monte Carlo EM algorithm, which is to simulate missing values from the predictive distribution $h(x_{\text{mis}}|\theta^{(t)}, X_{\text{obs}})$ at the $(t+1)th$ iteration, and then maximize the approximate conditional expectation of the complete-data loglikelihood

$$\hat{Q}(\theta|\theta^{(t)}) = \frac{1}{m} \sum_{j=1}^{m} \log f(X_{\text{obs}}, X_j^{\text{mis}}|\theta),$$
which converges to $Q(\theta | \theta^{(t)})$ as $m \to \infty$, where $X^{\text{mis}}_1, \ldots, X^{\text{mis}}_m$ denote the missing values simulated from $h(x^{\text{mis}} | \theta^{(t)}, X^{\text{obs}})$. When the dimension of $X^{\text{mis}}$ is high, the Monte Carlo approximation can be rather expensive. An alternative algorithm to deal with the intractable E-step is the stochastic EM (SEM) algorithm (Celeux and Diebolt, 1985). In this algorithm, the E-step is replaced by an imputation step, where the missing data are imputed with plausible values conditional on the observed data and the current estimate of the parameters. At the M-step, the MLE of the parameters is computed based on the pseudo-complete data. Unlike the deterministic EM algorithm, the imputation-step and M-step of the SEM algorithm generate a Markov chain which converges to a stationary distribution whose mean is close to the MLE and whose variance reflects the information loss due to the missing data (Nielsen, 2000).

Although EM and its variants work well for low-dimensional problems, see McLachlan and Krishnan (2008) for an overview, they essentially fail for high-dimensional problems. For the latter, the MLE can be non-unique or inconsistent. To address this issue, some problem-specific algorithms have been proposed, see e.g., misgLasso (Städler and Bühlmann, 2012), misPALasso (Städler et al., 2014), and matrix completion algorithms (Cai et al. 2010; Mazumder et al., 2010). The matrix completion algorithm deals with large incomplete matrices, which is to learn a low-rank approximation for a large-scale matrix with missing entries. MisgLasso is specifically designed for estimating Gaussian graphical models in presence of missing data. MisPALasso is similar to misgLasso, which also deals with multivariate Gaussian data in presence of missing data. However, there still lacks a general algorithm for high-dimensional missing data problems.

This work is to fill the gap: we propose a general algorithm for dealing with high-dimensional missing data problems. The proposed algorithm consists of two steps, an imputation step and a consistency step, and is thus called an imputation-consistency (IC) algorithm. The imputation step is to impute the missing data with plausible values conditioned on the observed data and the current estimate of parameters. The consistency step is to find a consistent estimate for the minimizer of a Kullback-Leibler divergence defined on the pseudo-complete data. For high dimensional problems, the consistent estimate is suggested to be found by a regularization approach. Like the SEM algorithm, the IC algorithm generates a Markov chain which converges to a stationary distribution. Under mild conditions, we show that the mean of the stationary distribution converges to the true value of the parameters in probability as the sample size becomes large. The IC algorithm is general, which, in principle, can be applied to any missing data problems, regardless the dimension and distribution of the data. For the low-dimensional problems, the SEM algorithm can be viewed as a special case of the IC algorithm. The IC algorithm has strong implications for big data computing: Based on it, we propose a general strategy to improve Bayesian computation for big data. The IC algorithm also facilitates data integration from multiple sources, which plays an important role in big data analysis.

The remainder of this paper is organized as follows. Section 2 describes the IC algorithm with the theoretical development deferred to the Appendix. Section 3 applies the IC algorithm to high-dimensional Gaussian graphical models. Section 4 applies the IC algorithm to high-dimensional variable selection. Section 5 applies the IC algorithm to a random coefficient model and discusses its potential use for big data analysis.
2 The Imputation-Consistency Algorithm

2.1 The IC Algorithm

Let \( X_1, \ldots, X_n \) denote a random sample drawn from the distribution \( f(x|\theta) \) (also denoted by \( f_\theta(x) \) depending on convenience), where \( \theta \) is a vector of parameters. Let \( X_i = (X_{i}^{\text{obs}}, X_{i}^{\text{mis}}), i = 1, \ldots, n \), where \( X_{i}^{\text{obs}} \) is observed and \( X_{i}^{\text{mis}} \) is missed. Let \( X = (X_1, \ldots, X_n), X^{\text{obs}} = (X_1^{\text{obs}}, \ldots, X_n^{\text{obs}}) \) and \( X^{\text{mis}} = (X_1^{\text{mis}}, \ldots, X_n^{\text{mis}}) \). To indicate the dependence of the dimension of \( \theta \) on the sample size \( n \), we also write \( \theta \) as \( \theta_n \) and denote by \( \theta_n^{(t)} \) the estimate of \( \theta \) obtained at the \( t^{th} \) iteration of the IC algorithm.

The IC algorithm works as follows: Starting with an initial guess \( \theta_n^{(0)} \) and then iterating between the imputation and consistency steps:

- **I-step:** Draw \( \tilde{X}^{\text{mis}} \) from the predictive distribution \( h(x_{\text{mis}}|X^{\text{obs}}, \theta_n^{(t)}) \) given \( X^{\text{obs}} \) and the current estimate \( \theta_n^{(t)} \).

- **C-step:** Based on the pseudo-complete data \( \tilde{X} = (X^{\text{obs}}, \tilde{X}^{\text{mis}}) \), find an updated estimate \( \theta_n^{(t+1)} \) which forms a consistent estimate of

\[
\theta_n^{(t)} = \arg \max_{\theta} E_{\theta_n^{(t)}} \log f_\theta(\tilde{x}), \tag{1}
\]

where \( E_{\theta_n^{(t)}} \log f_\theta(\tilde{x}) = \int \log(f(x^{\text{obs}}, \tilde{x}^{\text{mis}}|\theta))f(x^{\text{obs}}|\theta^*)h(\tilde{x}^{\text{mis}}|x^{\text{obs}}, \theta_n^{(t)})dx^{\text{obs}}d\tilde{x}^{\text{mis}}, \theta^* \) denotes the true value of the parameters, and \( f(x^{\text{obs}}|\theta^*) \) denotes the marginal density function of \( x^{\text{obs}} \).

To find a consistent estimate of \( \theta_n^{(t)} \), which is the minimizer of the Kullback-Leibler divergence from \( f(\tilde{x}|\theta) \) to the joint density \( f(x^{\text{obs}}|\theta^*)h(\tilde{x}^{\text{mis}}|x^{\text{obs}}, \theta_n^{(t)}) \), we suggest a regularization approach: It works through maximizing a penalized likelihood function by setting

\[
\theta_n^{(t+1)} = \arg \max_{\theta} \left[ \log f(X^{\text{obs}}, \tilde{X}^{\text{mis}}|\theta) - \lambda P(\theta) \right], \tag{2}
\]

where \( P(\theta) \) denotes the penalty function of \( \theta \), \( \lambda \) is an appropriately tuned regularization parameter, and \( \tilde{X}^{\text{mis}} \) denotes the imputed data based on the current estimate \( \theta_n^{(t)} \). Note that (2) has included the maximum a posteriori (MAP) estimate as a special case. In Theorem 1 (see Appendix), we prove that \( \theta_n^{(t+1)} \) obtained in (2) is a consistent estimate of \( \theta_n^{(t)} \) under quite general conditions. For this reason, the parameter estimation step of the algorithm is called the consistency step. For low-dimensional problems, the consistent estimator of \( \theta_n^{(t)} \) can be obtained by maximizing the pseudo-complete likelihood function by setting \( \lambda = 0 \). In this sense, the SEM algorithm can be viewed as a special case of the IC algorithm.

The IC algorithm is general: In principle, it can be applied to any missing data problems, regardless the dimension and distribution of the data.
For the IC algorithm, it is easy to see that by simulating new independent missing values at each iteration, the sequence of estimates, \( \{ \theta_n^{(t)} \} \), forms a time-homogeneous Markov chain. Also, the imputed values at different iterations form a Markov chain. The two Markov chains are interleaved and share many properties, such as irreducibility, aperiodicity and ergodicity. Refer to Nielsen (2000) for more discussions on this issue. In Theorem 2 and Theorem 3 (see Appendix), we prove that the Markov chain \( \{ \theta_n^{(t)} \} \) has a stationary distribution and, furthermore, the mean of the stationary distribution forms a consistent estimate of \( \theta^* \), the true parameter vector.

Finally, we note that many of the conditions used in proving above theorems are quite regular. For example, we assume that \( \log f_\theta(\tilde{x}) \) is continuous function of \( \theta \) for each \( \tilde{x} \in X \) and a measurable function of \( \tilde{x} \) for each \( \theta \), and restrict the parameter space \( \Theta \) to be compact. These conditions have often been used in studying the property of MLE for misspecified models, see e.g., Jennrich (1969) and White (1982). The compactness assumption of \( \Theta \) is not much restrictive for applications of the IC algorithm. For example, we can set \( \Theta \) to be very large, say, \([-10^{100}, 10^{100}]^p \) which, as a practical matter, is equivalent to set \( \Theta = \mathbb{R}^p \). Here \( p \) denotes the dimension of \( \theta \), which can grow at \( O(e^{\alpha n}) \) for some \( \alpha < 1/2 \). For the penalty function \( P(\theta) \), we assume that it converges to zero uniformly for all \( \theta \) in \( \Theta \). This condition can be satisfied by imposing some sparsity conditions on \( \theta \). For example, for high-dimensional regression, one can first apply a sure independence screening procedure (Fan and Lv, 2008; Fan and Song, 2010) to reduce the number of variables to \( O(n / \log(n)) \) and then conduct variable selection with an appropriate penalty function. Since \( \Theta \) is bounded, it is easy to see that the popular penalty functions, such as Lasso (Tibshirani, 1996), SCAD (Fan and Li, 2001) and MCP (2010), all satisfy the condition if used in conjunction with a sure screening procedure.

### 2.2 An Extension of the IC Algorithm

Like the EM algorithm, the IC algorithm is attractive only when the consistent estimate of \( \theta_s^{(t)} \) can be easily obtained at each C-step. We found that for many problems, similar to the ECM algorithm (Meng and Rubin, 1993), the consistent estimate of \( \theta_s^{(t)} \) can be easily obtained with a number of conditional consistency steps. That is, we can partition the parameter \( \theta \) into a number of blocks and then find the consistent estimator for each block conditional on the current estimates of other blocks. Note that for many problems, e.g., the example studied in Sections 4 and 5, the partitioning of \( \theta \) is natural.

Suppose that \( \theta = (\theta^{(1)}, \ldots, \theta^{(k)}) \) has been partitioned into \( k \) blocks. The imputation-conditional consistency (ICC) algorithm can be described as follows:

- **I-step.** Draw \( X^{\text{mis}} \) from the conditional distribution \( h(x^{\text{mis}}|X^{\text{obs}}, \theta_n^{(t,1)}, \ldots, \theta_n^{(t,k)}) \) given \( X^{\text{obs}} \) and the current estimate \( \theta_n^{(t)} = (\theta_n^{(t,1)}, \ldots, \theta_n^{(n,k)}) \).

- **CC-step.** Based on the pseudo-complete data \( \hat{X} = (X^{\text{obs}}, X^{\text{mis}}) \), do the following step:
Theorem 5 (see Appendix), we prove that the Markov chain of the Markov chain can be studied under similar conditions as the IC algorithm. In Theorem 4 and furthermore, the mean of the stationary distribution forms a consistent estimate of the GGM is equivalent to identify the non-zero elements of the concentration matrix (i.e., the inverse of a multivariate Gaussian distribution $N$ and improves accuracy of statistical analysis.

Therefore, it overcomes the weakness of the “one-time” imputation methods, missing values for microarray data. The IC algorithm iteratively impute missing values based on the up-the missing values cannot be properly accounted for. In this section, we apply the IC algorithm to handle statistical analysis based on the “one-time” imputed data is potentially biased, because the uncertainty of missing values independent of the models under consideration, they are often ineffective. Moreover, the analysis (BPCA) imputation (Oba et al., 2003), have been proposed. Since these methods impute the in microarray data, many imputation methods, such as single value decomposition (SVD) imputation (Troyanskaya et al., 2001), least-square imputation (Bo et al., 2004), and Bayesian principal component analysis (BPCA) imputation (Oba et al., 2003), have been proposed. Since these methods impute the missing values independent of the models under consideration, they are often ineffective. Moreover, the statistical analysis based on the “one-time” imputed data is potentially biased, because the uncertainty of the missing values cannot be properly accounted for. In this section, we apply the IC algorithm to handle missing values for microarray data. The IC algorithm iteratively impute missing values based on the updated parameter estimate. Therefore, it overcomes the weakness of the “one-time” imputation methods, and improves accuracy of statistical analysis.

Let $X = (x_1, \ldots, x_n)^T$ denote a microarray dataset of $n$ samples and $p$ genes, where $x_i$ is assumed to follow a multivariate Gaussian distribution $N_p(\mu, \Sigma)$. According to the theory of GGMs, estimation of the GGM is equivalent to identify the non-zero elements of the concentration matrix (i.e., the inverse of

(k) Conditional on $(\theta_n^{(t+1,1)}, \ldots, \theta_n^{(t,k-1)})$, find $\theta_n^{(t+1,k)}$ which forms a consistent estimate of

$$
\theta_n^{(t,k)} = \arg \max_{\theta_n^{(t,k)}} E_{\theta_n^{(t+1,1)}, \ldots, \theta_n^{(t+1,k-1)}, \theta_n^{(t,k)}} \log f(\tilde{x} | \theta_n^{(t+1,1)}, \ldots, \theta_n^{(t+1,k-1)}, \theta_n^{(t,k)}).
$$

It is easy to see that the estimate sequence $\{\theta_n^{(t,1)}, \ldots, \theta_n^{(t,k)}\}$ forms a Markov chain. The convergence of the Markov chain can be studied under similar conditions as the IC algorithm. In Theorem 4 and Theorem 5 (see Appendix), we prove that the Markov chain $\{\theta_n^{(t)}\}$ has a stationary distribution and, furthermore, the mean of the stationary distribution forms a consistent estimate of $\theta^*$.

3 Learning High-Dimensional Gaussian Graphical Models in Presence of Missing Data

Gaussian graphical models (GGMs) have often been used in learning gene regulatory networks from microarray data, see e.g., Dobra et al. (2004) and Friedman et al. (2008). As mentioned in the Introduction, missing values can appear in microarray data due to many factors. To deal with missing values in microarray data, many imputation methods, such as single value decomposition (SVD) imputation (Troyanskaya et al., 2001), least-square imputation (Bo et al., 2004), and Bayesian principal component analysis (BPCA) imputation (Oba et al., 2003), have been proposed. Since these methods impute the missing values independent of the models under consideration, they are often ineffective. Moreover, the statistical analysis based on the “one-time” imputed data is potentially biased, because the uncertainty of the missing values cannot be properly accounted for. In this section, we apply the IC algorithm to handle missing values for microarray data. The IC algorithm iteratively impute missing values based on the updated parameter estimate. Therefore, it overcomes the weakness of the “one-time” imputation methods, and improves accuracy of statistical analysis.

Let $X = (x_1, \ldots, x_n)^T$ denote a microarray dataset of $n$ samples and $p$ genes, where $x_i$ is assumed to follow a multivariate Gaussian distribution $N_p(\mu, \Sigma)$. According to the theory of GGMs, estimation of the GGM is equivalent to identify the non-zero elements of the concentration matrix (i.e., the inverse of
the covariance matrix $\Sigma$) or to identify non-zero partial correlation coefficients for different pairs of genes. During the recent years, a couple of methods have been proposed to estimate high-dimensional GGMs, e.g., graphical Lasso (Yuan and Lin, 2007; Friedman et al., 2008), node-wise regression (Meinshausen and Bühlmann, 2006), and $\psi$-learning (Liang et al., 2015). Graphical Lasso is to estimate the concentration matrix using a $L_1$-penalty. Node-wise regression is developed based on the relationship between the partial correlation coefficients and regression coefficients, and it is to select the non-zero regression coefficients for each gene regressed with all other genes. The $\psi$-learning algorithm is developed based on an equivalent measure of the partial correlation coefficient, which is evaluated with a reduced conditional set of genes and thus feasible for high-dimensional problems. However, none of the methods can be directly applied in presence of missing data.

3.1 The IC Algorithm

To apply the IC algorithm to learn GGMs in presence of missing data, we choose the $\psi$-learning algorithm as the consistent estimator used in the C-step. For GGMs, $\theta$ corresponds to the concentration matrix, which can be uniquely determined from the network structure using the algorithm given in Hastie et al. (2009, p.634). Under mild conditions, Liang et al. (2015) showed that the $\psi$-learning algorithm provides a consistent estimator for Gaussian graphical networks. Refer to the Supplementary Material for a brief review of the algorithm. As explained there, the $\psi$-learning algorithm works based on the sure screening technique and thus the condition on the penalty function is satisfied. Other than the $\psi$-learning algorithm, node-wise regression and graphical Lasso can also be applied here. They both belong to the class of regularization methods and are consistent in Gaussian graphical network estimation.

The Gaussian graphical network specifies the dependence between different genes, according to which the missing values can be imputed. For convenience, we let $A = (a_{jk})$ denote the adjacency matrix of a Gaussian graphical network, where $a_{jk} = 1$ if an edge exists between node $j$ and node $k$ and 0 otherwise. For microarray data, a node corresponds to a gene. Let $x_{ij}$ denote a missing entry, and let $\omega(j) = \{k : a_{jk} = 1\}$ denote the neighborhood of node $j$. According to the faithfulness property of GGMs, conditional on the neighboring genes in $\omega(j)$, gene $j$ is independent of all other genes. Therefore, $x_{ij}$ can be imputed conditional on the expression values of the neighboring genes. Mathematically, we have

$$
\begin{pmatrix}
x_{ij} \\
x_{i\omega}
\end{pmatrix} \sim N \left( \begin{pmatrix} \mu_j \\ \mu_\omega \end{pmatrix}, \begin{pmatrix} \sigma_j^2 & \Sigma_{j\omega} \\ \Sigma_{j\omega}^T & \Sigma_{\omega\omega} \end{pmatrix} \right)
$$

(3)

where $x_{i\omega} = \{x_k : k \in \omega(j)\}$, and $\mu_j, \mu_\omega, \sigma_j^2, \Sigma_{j\omega}$ and $\Sigma_{\omega\omega}$ denote the corresponding mean and variance components. The mean and variance of $x_{ij}$ conditional on $x_{i\omega}$ is thus given by

$$
\begin{align*}
\mu_{ij|\omega} &= \mu_j + \Sigma_{j\omega} \Sigma_{\omega\omega}^{-1} (x_{i\omega} - \mu_\omega), \\
\sigma_{ij|\omega}^2 &= \sigma_j^2 - \Sigma_{j\omega} \Sigma_{\omega\omega}^{-1} \Sigma_{j\omega}^T.
\end{align*}
$$

(4)

As shown in Liang et al. (2015), for each variable, the neighborhood size can be upper bounded by $\lceil n/\log(n) \rceil$, where $\lceil z \rceil$ denotes the smallest integer not smaller than $z$. Hence, in practice, $\sigma_j^2, \Sigma_{j\omega}$ and
\[ \Sigma_{\omega\omega} \] can be directly estimated from the data. Let \( s_j^2, S_{j\omega}, S_{\omega\omega}, \bar{x}_j \) and \( \bar{x}_\omega \) denote the respective sample estimate of \( \sigma_j^2, \Sigma_{j\omega}, \Sigma_{\omega\omega}, \mu_j \) and \( \mu_\omega \). Then, at each iteration, \( x_{ij} \) can be imputed by sampling from the distribution

\[
X_{ij|\omega} \sim N(\bar{x}_j + S_{j\omega} S_{\omega\omega}^{-1}(x_{i\omega} - \bar{x}_\omega), s_j^2 - S_{j\omega} S_{\omega\omega}^{-1} S_{\omega j}^T).
\]

(5)

In this way, the exact evaluation of the concentration matrix can be skipped. In summary, we have the following algorithm for learning GGMs in presence of missing data:

- **(Initialization)** Replace each missing entry by the median of the observations of the corresponding variable, and then iterates between the C- and I-steps.
- **(C-step)** Apply the \( \psi \)-learning algorithm to learn the structure of the Gaussian graphical network.
- **(I-step)** Impute missing values according to (5) based on the network learned in the C-step.

This algorithm outputs a series of Gaussian graphical networks. To integrate/average these networks into a single network, we adopt the \( \psi \)-score averaging approach suggested by Liang et al. (2015). Let \( \psi_{ij}^{(t)} \) denote the \( \psi \)-scores obtained at iteration \( t \) of the algorithm. The \( \psi \)-score is obtained from the \( \psi \)-partial correlation coefficient via a Fisher’s transformation. Refer to Liang et al. (2015) for the detail. Let \( \tilde{\psi}_{ij} = \sum_{t=1}^{T} \psi_{ij}^{(t)} / T, i, j = 1, 2, \ldots, p \) and \( i \neq j \), denote the averaged \( \psi \)-score for gene \( i \) and gene \( j \). Then the averaged network can be obtained by applying a multiple hypothesis approach to threshold the averaged \( \psi \)-scores; if an averaged \( \psi \)-score is greater than the threshold value, we set the corresponding element of the adjacency matrix to 1 and 0 otherwise. The multiple hypothesis test can be done using the method of Liang and Zhang (2008), which can be viewed as a generalized empirical Bayesian method by Efron (2004). The significance level of the multiple hypothesis test can be specified in terms of Storey’s \( q \)-value (Storey, 2002). In this paper, we set it to 0.05.

### 3.2 A Simulated Example

We consider an autoregressive process of order two with the concentration matrix given by

\[
C_{i,j} = \begin{cases} 
0.5, & \text{if } |j - i| = 1, i = 2, \ldots, (p - 1), \\
0.25, & \text{if } |j - i| = 2, i = 3, \ldots, (p - 2), \\
1, & \text{if } i = j, i = 1, \ldots, p, \\
0, & \text{otherwise}.
\end{cases}
\]

(6)

This example has been used by multiple authors, e.g., Yuan and Lin (2007), Mazumder et al. (2012), and Liang et al. (2015) to illustrate different GGM methods. In this paper, we generated multiple datasets with \( n = 200 \) and different values of \( p=100, 200, 300 \) and 400. For each dataset, we randomly delete 10% of the observations as missing values. To evaluate the performance of the IC algorithm, the precision-recall curves are drawn by varying the threshold value of \( \psi \)-scores. Refer to the Supplementary Material for the definition of precision and recall.
For each dataset, the IC algorithm was run for 50 iterations. The resulting precision-recall curves are shown in Figure 1 (for \( p = 100 \) and 400) and Figure 1 of the Supplementary Material (for \( p = 200 \) and 300), where “IC-Ave” refers to the curve obtained from the \( \psi \)-scores averaged over last 20 iterations, “IC-Last” refers to the curve obtained from the \( \psi \)-score obtained at the last iteration, and “True” refers to the curve obtained from the \( \psi \)-score calculated from the complete data. For comparison, we also tried three existing methods, “Median” and BPCA, and misgLasso. “Median” is to replace each missing value by the median of the observations of the corresponding variable, and BPCA is to replace the missing values by its BPCA estimate (Oba et al., 2003). As mentioned previously, BPCA is a method specially designed for imputing missing values of the microarray data. Under the Bayesian framework, this method estimates the probabilistic principal components of the data using an EM-like iterative algorithm, and then imputes the missing values based on the principal component regression. For both the “Median” and BPCA methods, the GGM will be learned using the \( \psi \)-learning algorithm after the missing values are imputed. The misgLasso algorithm is a combination of the gLasso and EM algorithms, which is to integrate out the missing data as in the EM algorithm (see e.g., Städler and Bühlmann, 2012) and then learn the GGM using the gLasso algorithm. The misgLasso algorithm has been implemented in the R package spaceExt (He, 2011).

Figure 1: Precision-recall curves for the GGM with missing data (the left panel is for \( p = 100 \) and the right for \( p = 400 \)): “IC-Ave” refers to the curve obtained from the \( \psi \)-scores averaged over last 20 iterations, “IC-Last” refers to the curve obtained from the \( \psi \)-score generated in the last iteration, “True” refers to the curve obtained from the \( \psi \)-score calculated using the complete data, “Median” refers to the curve obtained from the \( \psi \)-score calculated with the missing entry replaced by the median expression value of the corresponding gene, “BPCA” refers to the curve obtained from the \( \psi \)-score calculated with the missing entries replaced by the BPCA estimate, and “misgLasso” refers to the misgLasso algorithm.

The comparison indicates that the IC algorithm outperforms all other algorithms. As shown in Figure 1, the “IC-Aver” curve is always the one closest to the “True” curve. It is interesting to note that although the “IC-Last” curve is also based on one-time imputation, it is much better than the “Median” and BPCA
curves. This suggests that for microarray data, the model-based imputation method is potentially more accurate than other one-time imputation methods. All other methods shown in the plot outperform the misgLasso algorithm. This does not mean that the EM algorithm does not work for dealing with the missing data, but the gLasso algorithm does not work well for this example. This is consistent with Liang et al. (2015), where it is shown that the \( \psi \)-learning algorithm works much better than gLasso for the complete data version of this example.

3.3 Yeast Cell Expression Data

Gasch et al. (2000) explored genomic expression patterns in the yeast *Saccharomyces cerevisiae* responding to diverse environmental changes. The whole dataset has a missing rate of 3.01% and is available at http://genome-www.stanford.edu/yeast-stress/. Our numerical results for a subset of 1000 genes, reported in the Supplementary Material, indicate that the IC algorithm works reasonably well for this example with a few hub genes successfully identified, which are expected to play an important role for yeast cells in response to environmental changes.

4 High-Dimensional Variable Selection in Presence of Missing Data

This problem is also motivated by microarray data analysis, but the goal has been shifted to selection of genes relevant to a particular phenotype. To be more general, we let \( \mathbf{Y} = (Y_1, \ldots, Y_n)^T \) denote the response vector for \( n \) observations, and let \( \mathbf{X} = (X_1, \ldots, X_n)^T \) denote the matrix of covariates, where each \( X_i \) is a \( p \)-dimensional vector and \( p \) can be much larger than \( n \) (a.k.a. small-\( n \)-large-\( p \)). The response variable and covariates are linked through the regression,

\[
\mathbf{Y} = (\mathbf{1}_n, \mathbf{X})\mathbf{\beta} + \mathbf{\epsilon},
\]  

(7)

where \( \mathbf{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^T \) denotes the vector of regression coefficients, and \( \mathbf{\epsilon} \sim N(0, \sigma^2 I_n) \) denotes the vector of random errors.

Variable selection for the model (7) with complete data has been extensively studied in the recent literature. Methods have been developed from both frequentist and Bayesian perspectives, see e.g., Tibshirani (1996) and Song and Liang (2015). For incomplete data, Garcia et al. (2010) proposed to conduct variable selection by maximizing the penalized likelihood function of the incomplete data. However, when \( p \) is large and the covariates \( X_i \)'s are generally correlated, the incomplete data likelihood function can be intractable, rendering failure of their method. Zhao and Long (2013) showed through numerical studies that for the high dimensional data the standard multiple imputation approach performs poorly, while the imputation method based on Bayesian Lasso often works better. However, since Bayesian Lasso tends to over-shrink the non-zero regression coefficients, its consistency in variable selection is hard to be justified when \( p \) is much greater than \( n \) (Castillo et al., 2015). Quite recently, Long and Johnson (2015) proposed
to combine Bayesian Lasso imputation and stability selection (Meinshausen and Bühlmann, 2010). Again, the consistency of this method is hard to be achieved due to the inconsistency of Bayesian Lasso.

### 4.1 The ICC Algorithm

In what follows, we consider a general setting of (7), where the covariates follow a multivariate Gaussian distribution $\mathbf{X} \sim N(\mu, \Sigma)$. Under this general setting, the parameter vector $\mathbf{\theta}$ consists of three natural blocks $\mathbf{\beta}$, $\sigma^2_\epsilon$ and the concentration matrix $\mathbf{C} = \Sigma^{-1}$. Since $n$ has been assumed to be smaller than $p$, we further assume the sparsity for both the regression coefficients $\mathbf{\beta}$ and the concentration matrix $\mathbf{C}$.

To apply the ICC algorithm to this problem, we choose the SIS-MCP algorithm as the consistent estimator of $\mathbf{\beta}$. That is, the variables are first subject to a sure independence screening procedure, and then the survived variables are selected using the MCP method (Zhang, 2010). This algorithm has been implemented in the R-package SIS. Given an estimates of $\mathbf{\beta}$, $\sigma^2_\epsilon$ can be estimated by $\hat{\sigma}^2_\epsilon = \frac{\sum_{i=1}^{n} \hat{\epsilon}_i^2}{(n - |\hat{\mathbf{\beta}}| - 1)}$, where $\hat{\epsilon}_i$ denotes the residual of sample $i$, and $|\hat{\mathbf{\beta}}|$ denotes the number of nonzero elements included in the estimate $\hat{\mathbf{\beta}}$. Given the consistency of $\hat{\mathbf{\beta}}$, the consistency of $\hat{\sigma}^2_\epsilon$ is easy to be justified. To estimate the concentration matrix $\mathbf{C}$, we choose the $\psi$-learning algorithm. As mentioned previously, the $\psi$-learning algorithm provides a consistent estimate for the Gaussian graphical network, based on which a consistent estimate of the concentration matrix can be uniquely determined by the algorithm given in Hastie et al. (2009, p.634). Note that SIS-MCP does not make use of the dependency among the covariates. Given the structure of the ICC algorithm, some other variable selection algorithms which have made use of the dependency among the covariates, e.g., Yu and Liu (2016), can also be applied here.

Next, we consider the imputation step. Suppose that the value of $x_{hk}$ is missed in $\mathbf{X}$. Section 4 of the Supplementary Material presents the conditional distributions of $X_{hk}$ given $\mathbf{Y}$ and the rest elements of $\mathbf{X}$ under different scenarios. Based on the conditional distributions, $x_{hk}$ can be easily imputed by sampling from the respective samplized conditional distributions. Here the samplized conditional distribution refers to the distribution with its population parameters replaced by their respective estimates calculated from samples. For example, $\beta_i$’s are replaced by their SIS-MCP estimates, $\sigma^2_\epsilon$ is replaced by $\hat{\sigma}^2_\epsilon$, etc. In summary, the ICC algorithm works as follows:

- **(Initialization)** Replace each missing entry of $\mathbf{X}$ by the median of the corresponding column, and then iterates between the CC- and I-steps.

- **(CC-step)** (i) Apply the SIS-MCP algorithm to estimate the regression coefficients $\mathbf{\beta}$; (ii) estimate $\sigma^2_\epsilon$ conditional on the estimate of $\mathbf{\beta}$; and (iii) apply the $\psi$-learning algorithm to learn the structure of the Gaussian graphical network.

- **(I-step)** Impute missing values according to the conditional distributions (given in the Supplemental Material) based on the regression model and network structure learned in the CC-step.
4.2 A Simulated Example

The datasets were simulated from the model (7) with \( n = 100 \) and \( p=200 \) and 500. The covariates \( \mathbf{X} \) were generated under two settings: (i) the covariates are mutually independent, where \( x_i \sim N(0,2I_n) \) for \( i = 1, \ldots, n \); and (ii) the covariates are generated according to the concentration matrix (6). For both settings, we set \( (\beta_0, \beta_1, \ldots, \beta_5) = (1, 1, 2, -1.5, -2.5, 5) \) and \( \beta_6 = \cdots = \beta_p = 0 \), and random error \( \epsilon \sim N(0, I_n) \). For each pair of \( (n, p) \), we simulated 10 datasets independently. For each dataset, we considered two missing rates, randomly deleting 5% and 10% entries of \( \mathbf{X} \) as missing values. The performance of different methods was measured using three criteria:

\[
\text{err}_\beta^2 = \| \hat{\beta} - \beta \|^2, \quad \text{fsr} = \frac{| s \backslash s^* |}{| s |}, \quad \text{nsr} = \frac{| s^* \backslash s |}{| s^* |},
\]

where \( \| \cdot \| \) denotes the Euclidean norm, \( \hat{\beta} \) denotes the estimate of \( \beta \), \( s^* \) denotes the set of true covariates, and \( s \) denotes the set of selected covariates.

The ICC algorithm was first applied to this example with the results summarized in Table 1 and 2. For comparison, we also tried the one-time imputation methods, “Median” and BPCA. As explained previously, “Median” is to replace each missing value by the median of the observations of the corresponding variable, and BPCA is to impute the missing values based on the principal component regression. Then the variables are selected using the SIS-MCP method.

Table 1: Comparison of the ICC algorithm with the “Median” and BPCA methods for high-dimensional variable selection with independent covariates. "True" denotes the results obtained by the MCP method from the complete data. The values in the table are obtained by averaging over 10 independent datasets with the standard deviation reported in the parentheses.

<table>
<thead>
<tr>
<th>( p )</th>
<th>Missing Rate</th>
<th>BPCA</th>
<th>Median</th>
<th>ICC</th>
<th>True</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>5%</td>
<td>err^2_\beta</td>
<td>0.257(0.267)</td>
<td>0.262(0.261)</td>
<td>0.042(0.041)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fsr</td>
<td>0.119(0.143)</td>
<td>0.082(0.092)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>nsr</td>
<td>0(0)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>err^2_\beta</td>
<td>0.903(0.396)</td>
<td>0.856(0.421)</td>
<td>0.065(0.087)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fsr</td>
<td>0.310(0.159)</td>
<td>0.308(0.178)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>nsr</td>
<td>0(0)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td>500</td>
<td>5%</td>
<td>err^2_\beta</td>
<td>0.339(0.214)</td>
<td>0.350(0.206)</td>
<td>0.029(0.034)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fsr</td>
<td>0.249(0.225)</td>
<td>0.266(0.237)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>nsr</td>
<td>0(0)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>err^2_\beta</td>
<td>1.532(1.071)</td>
<td>1.354(0.895)</td>
<td>0.044(0.022)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fsr</td>
<td>0.470(0.265)</td>
<td>0.420(0.255)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>nsr</td>
<td>0.033(0.070)</td>
<td>0.017(0.053)</td>
<td>0(0)</td>
</tr>
</tbody>
</table>
Table 2: Comparison of the ICC algorithm with the “Median” and BPCA methods for high-dimensional variable selection with dependent covariates. “True” denotes the results obtained by the MCP method from the complete data.

<table>
<thead>
<tr>
<th>p</th>
<th>Missing Rate</th>
<th>BPCA</th>
<th>Median</th>
<th>ICC</th>
<th>True</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$\hat{\sigma}^2_{\beta}$</td>
<td>$\hat{\sigma}^2_{\beta}$</td>
<td>$\hat{\sigma}^2_{\beta}$</td>
<td>$\hat{\sigma}^2_{\beta}$</td>
</tr>
<tr>
<td>200</td>
<td>5%</td>
<td>0.580(0.413)</td>
<td>0.548(0.140)</td>
<td>0.118(0.097)</td>
<td>0.071(0.050)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.262(0.204)</td>
<td>0.263(0.200)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.017(0.052)</td>
<td>0.017(0.052)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>1.604(0.666)</td>
<td>1.575(0.974)</td>
<td>0.424(0.461)</td>
<td>0.071(0.050)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.247(0.229)</td>
<td>0.273(0.238)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.100(0.086)</td>
<td>0.083(0.088)</td>
<td>0.033(0.070)</td>
<td>0(0)</td>
</tr>
<tr>
<td>500</td>
<td>5%</td>
<td>0.669(0.366)</td>
<td>0.717(0.358)</td>
<td>0.172(0.195)</td>
<td>0.096(0.083)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.262(0.202)</td>
<td>0.289(0.236)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.017(0.053)</td>
<td>0.017(0.053)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>2.752(2.306)</td>
<td>2.896(2.601)</td>
<td>0.578(0.587)</td>
<td>0.096(0.083)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.297(0.230)</td>
<td>0.327(0.224)</td>
<td>0(0)</td>
<td>0(0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.133(0.070)</td>
<td>0.133(0.070)</td>
<td>0.050(0.081)</td>
<td>0(0)</td>
</tr>
</tbody>
</table>

The comparison indicates that the ICC algorithm works extremely well for this example. For the case of independent covariates, its results are almost as good as those obtained from the complete data. In both cases, the ICC algorithm significantly outperforms the one-time imputation methods.

4.3 A Real Data Example

We analyzed one real gene expression dataset about Bardet-Biedl syndrome (Scheetz et al., 2006). The complete dataset contains 120 samples, where the expression level of the gene TRIM32 works as the response variable and the expression levels of 200 other genes work as the predictors. The dataset is available in the R package flare. We generate 10 incomplete datasets from the complete one by randomly deleting 5% observations. For each incomplete dataset, we ran the ICC algorithm for 30 iterations and averaged the estimates of $\hat{\beta}$ obtained in the last 10 iterations as the final estimate. For comparison, the “Median” and BPCA methods were also applied to this example. Table 3 summarizes the estimation errors of $\hat{\beta}$ (with respect to $\hat{\beta}_c$, the estimate of $\beta$ from the complete data) produced by the three methods for the 10 incomplete datasets.

We have also explored the results of variable selection. The complete data model selects 5 variables: v.153, v.180, v.185, v.87 and v.200. For the ICC, “Median” and BPCA models, we count the selection frequency of each variable for the 10 incomplete datasets. For the ICC models, the top 5 variables in selection frequency are v.153, v.185, v.180, v.87 and v.200, which are the same (ignoring the order) as
Table 3: Estimation errors of $\hat{\beta}$ (with respect to $\beta_c$) produced by ICC, “Median” and BPCA for the Bardet-Biedl syndrome example, where $\text{err}^2_{\beta}$ is calculated by averaging $\|\hat{\beta} - \beta_c\|^2$ over 10 incomplete datasets, and “s.d.” represents the standard deviation of $\text{err}^2_{\beta}$.

<table>
<thead>
<tr>
<th>Method</th>
<th>BPCA</th>
<th>Median</th>
<th>ICC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{err}^2_{\beta}$</td>
<td>0.428</td>
<td>0.397</td>
<td>0.187</td>
</tr>
<tr>
<td>s.d.</td>
<td>0.091</td>
<td>0.086</td>
<td>0.040</td>
</tr>
</tbody>
</table>

the complete data model. For the “Median” models, the top 5 variables are v.153, v.185, v.62, v.200 and v.54. For the BPCA models, the top 5 variables are v.153, v.87, v.185, v.62 and v.200. Both the results of $\beta$ estimation and variable selection indicate the superiority of the ICC algorithm over the one-time imputation methods.

5 A Random Coefficient Linear Model

To further illustrate the use of the ICC algorithm, we consider a random coefficient linear model. Such a model often arises, for instance, in recommendation systems where the customers rate different items, e.g., products or service. Specifically, we simulate the data from the following model

$$y_{ij} = x_{ij}^T \beta + z_i^T \lambda_i + w_j^T \gamma_j + e_{ij},$$

$$e_{ij} \sim N(0, \sigma^2), \quad \lambda_i \sim N(0, \Lambda), \quad \gamma_j \sim N(0, \Gamma),$$

(8)

where $y_{ij}$ represents the response for customer $i$ on item $j$. Assuming that there are $I$ customers and each customer responds to $J$ items. Thus, the dataset consists of a total of $n = IJ$ observations. The vector $x_{ij}$ represents the covariates that characterize the customers and items, e.g., how and how long the customer has purchased the item; $z_i$ represents customer-specific covariates such as gender, education and demographics; and $w_j$ represents item-specific covariates, e.g., the manufacturer and category of the item. The vector $\lambda_i$ represents the customer-specific (random) coefficients and $\gamma_j$ represents the item-specific (random) coefficients. This model can be easily extended to the case where each customer responds to only a subset of items. For this model, we treat the random coefficients $\lambda_i$’s and $\gamma_j$’s as missing data, and are interested in the estimation of $\beta$. For simplicity, we assume that $\beta$ is low-dimensional, although the whole dataset can be big when $I$ and/or $J$ become large. Under this assumption, the ICC algorithm is essentially reduced to the stochastic EM algorithm for this example. Instead of using the ICC algorithm in this straightforward way, we propose to use it under the Bayesian framework. This extends the applications of the ICC algorithm to Bayesian computation.

To conduct Bayesian analysis for the model, we assume the following semiconjugate priors:

$$\beta \sim N(\mu_\beta, \Sigma_\beta), \quad \sigma^2 \sim IG(a, b), \quad \Lambda \sim IW(\rho_\Lambda, R_\Lambda), \quad \Gamma \sim IW(\rho_\Gamma, R_\Gamma),$$

(9)
where \( IG(\cdot, \cdot) \) denotes the inverted Gamma distribution, \( IW(\cdot, \cdot) \) denotes the inverted Wishart distribution, and \( \mu_\beta, \Sigma_\beta, a, b, \rho_\Lambda, R_\Lambda, \rho_F, \) and \( R_F \) are hyperparameters to be specified by the user. Each of these priors is individually conjugate to the normal likelihood function, given the other parameters, although the joint prior is not conjugate. Given these priors, the full conditional posterior distributions are derived in Section 5 of the Supplementary Material. Since, under the low-dimensional setting, the mode of the full conditional posterior distribution provides a consistent estimator for the corresponding parameter, the ICC algorithm can work as follows:

- **(Initialization)** Initialize \( \lambda_i \)'s, \( \gamma_j \)'s, and all parameters by some random numbers.
- **(CC-step)** Estimate the parameters \( \beta, \Lambda, \Gamma, \) and \( \sigma^2 \) by the mode of their respective full conditional posterior distributions.
- **(I-step)** Impute the values of \( \lambda_i \)'s and \( \gamma_j \)'s according to their respective full conditional posterior distributions.

Under the Bayesian framework, the ICC algorithm works in a similar way to the Gibbs sampler except that it is to replace the posterior samples of the parameters by their respective full conditional posterior modes. Figure 2 (a)&(b) show the sampling paths of \( \beta_0 \) and \( \beta_1 \) for a simulated example. Refer to the supplementary material for more details. As expected, the samples generated by the ICC algorithm tend to have smaller variations than those generated by the Gibbs sampler. Figure 2 (c)&(d) show the autocorrelation plots of the ICC samples and the Gibbs samples, respectively. The plots indicate that the ICC algorithm can converge faster than the Gibbs sampler.

We are aware that the accuracy of the ICC estimates is achieved at the price that we sacrifice the variance information of the posterior distribution contained in posterior samples. As pointed out by Nielsen (2000),
the variance of the ICC samples reflects the information loss due to the missing data. However, for the random coefficient example, the variance information can be obtained from the full conditional posterior distributions (given in the Supplementary Material) by simply plugging the parameter estimates into their variances. This observation suggests a general strategy to improve simulations of the Gibbs sampler: At each iteration, we only need to draw samples for the components for which the posterior variance is not analytically available and also of interest to us, and the other components can be replaced by the mode of the respective full conditional posterior distributions. Since the mode can be found with a subset of samples, which can be much cheaper than sampling from the full data conditional posterior, we expect that the proposed strategy can significantly facilitate Bayesian computation for big data problems. A further study of this proposed strategy will be reported elsewhere.

6 Discussion

In this paper, we have proposed the imputation-consistency algorithm, or the IC algorithm in short, as a general algorithm for dealing with high-dimensional missing data problems. Under quite general conditions, we show that the IC algorithm can lead to a consistent estimate for the parameters. We have also extended the IC algorithm to the case of multiple block parameters, which leads to the imputation-conditional consistency (ICC) algorithm. We illustrate the proposed algorithms using the high-dimensional Gaussian graphical models, high-dimensional variable selection, and a random coefficient model.

The ICC algorithm have strong implications for big data computing. Based on the ICC algorithm, we proposed a general strategy to improve Bayesian computation under big data scenario; that is, we can replace posterior samples by posterior modes in Gibbs iterations to accelerate simulations, where the posterior modes can be calculated with a subset of samples. In addition, the proposed algorithms facilitate data integration from multiple sources, which is very important for big data analysis. For example, we have often multiple microarray datasets measured for the same disease but in different studies, and expect that a proper integration of the multiple datasets can significantly improve the accuracy of statistical analysis. As mentioned in the Introduction, due to many factors, the proportion of the missing data can be significant for each dataset. A simple deletion of the related gene or samples can lead to much loss of the data information. Suppose that we are interested in learning a gene regulatory network from the multiple microarray datasets. Based on the IC algorithm, data integration can be done for this problem in the following procedure:

- (Initialization) Impute each missing entry by the median expression level of the corresponding gene, and then iterate between the following steps.

- (C-step) Apply the ψ-learning algorithm to learn a Gaussian graphical network for each dataset.

- (Integration) Integrate the multiple Gaussian graphical networks learned in the C-step into a single network using the network integration algorithm prescribed in Liang et al. (2015).
(I-step) Impute missing values for each dataset based on the integrated network.

As we can see, the above procedure converts the original problem of data integration in presence of missing values to the problem of data integration for complete data. Similarly, for high-dimensional variable selection, the data integration in presence of missing values can be done based on the ICC algorithm and the complete data integration technique developed by Chen and Xie (2014).

Like the EM algorithm for low-dimensional data, we expect that the IC/ICC algorithm can have many applications for high-dimensional data. Also, with the IC/ICC algorithm, many problems can be much simplified, e.g., variable selection for high-dimensional mixed effect models (Fan and Li, 2012). Under the framework provided by the IC/ICC algorithm, variable selection for mixed effect models is much like the problem of variable selection for a high-dimensional regression with fixed designs.

In this paper, we consider only one type of consistent estimator, which is given in (2). Although this estimator has been very general, including the regularized estimator and the MAP estimator, it is still of great interest to find more types of estimators that work for the IC/ICC algorithm.

Acknowledgement

Liang’s research was support in part by the grants DMS-1545202, DMS-1612924 and DMS/NIGMS R01-GM117597.

Appendix

1. Proof of consistency of \( \theta_n^{(t+1)} \) Define \( \Theta_n \) as the parameter space of \( \theta \), where the subscript \( n \) indicates the dependence of the dimension of \( \theta \) on the sample size \( n \), i.e., the dimension of \( \theta \) can increase with \( n \). Without possible confusion, we will often refer to \( \Theta_n \) as \( \Theta \).

Let \( \tilde{x} = (x_{\text{obs}}, \tilde{x}_{\text{mis}}) \) and define

\[
G_n(\theta | \theta_n^{(t)}) = E_{\theta_n^{(t)}} \log f_\theta(\tilde{x}) = \int \log f_\theta(\tilde{x}) f(x_{\text{obs}} | \theta) h(\tilde{x}_{\text{mis}} | \theta_n^{(t)}) d\tilde{x},
\]

\[
\hat{G}_n(\theta | \tilde{x}, \theta_n^{(t)}) = \log f(x_{\text{obs}}, \tilde{x}_{\text{mis}} | \theta) = \frac{1}{n} \sum_{i=1}^{n} \log f(x_{i_{\text{obs}}}, \tilde{x}_{i_{\text{mis}}}) | \theta).
\]

Then \( \theta_n^{(t+1)} = \arg \max_{\theta \in \Theta_n} \hat{G}_n(\theta | \tilde{x}, \theta_n^{(t)}) - \frac{1}{n} P(\theta) \), and \( \theta_*^{(t)} = \arg \max_{\theta \in \Theta_n} G_n(\theta | \theta_n^{(t)}) \). Our goal is to show that \( \theta_n^{(t+1)} \rightarrow_p \theta_*^{(t)} \) as \( n \rightarrow \infty \), and we require the following set of conditions:

(A1) \( \log f_\theta(\tilde{x}) \) is a continuous function of \( \theta \) for each \( \tilde{x} \in \mathcal{X} \) and a measurable function of \( \tilde{x} \) for each \( \theta \).

(A2) \( \Theta_n \) is compact.

(A3) There exists a function \( m_n(\tilde{x}) \) such that \( \sup_{\theta \in \Theta_n, \tilde{x} \in \mathcal{X}} | \log f_\theta(\tilde{x}) | \leq m_n(\tilde{x}) \).
(A4) The function \( P(\theta)/n \), where \( P(\theta) \) is the penalty function or the log-prior density function, converges to 0 uniformly for all \( \theta \in \Theta_n \).

(A5) \( G_n(\theta|\theta_n^{(t)}) \) has a unique maximum at \( \theta_n^{(t)} \) for all \( \theta_n^{(t)} \in \Theta_n \).

Define \( \hat{G}_n(\theta|\theta_n^{(t)}) = \frac{1}{n} \sum_{i=1}^{n} \log f(x_i^{\text{obs}}, z_i^{\text{mis}}|\theta)h(z_i^{\text{mis}}|x_i^{\text{obs}}, \theta_n^{(t)})d\tilde{x}^{\text{mis}} \), and \( \hat{m}_n(x^{\text{obs}}, \theta_n^{(t)}) = \int m_n(\tilde{x})h(\tilde{x}^{\text{mis}}|x^{\text{obs}}, \theta_n^{(t)})d\tilde{x}^{\text{mis}} \). It is obvious that

\[
\hat{G}_n(\theta|\tilde{x}, \theta_n^{(t)}) - G_n(\theta|\theta_n^{(t)}) = \{ \hat{G}_n(\theta|\tilde{x}, \theta_n^{(t)}) - \hat{G}_n(\theta|\theta_n^{(t)}) \} + \{ \hat{G}_n(\theta|\theta_n^{(t)}) - G_n(\theta|\theta_n^{(t)}) \}.
\]

There are two error terms in the above decomposition. The first one comes from the imputation of missing data, and the second one comes from the observed data itself. First, we show that under certain conditions, the second term converges to 0 uniformly. To achieve this goal, we further assume the following condition.

(A6) [Conditions for Glivenko-Cantelli theorem]

(a) Assume that there exists \( m_n^*(x^{\text{obs}}) \) such that \( 0 \leq \hat{m}_n(x^{\text{obs}}, \theta_n^{(t)}) \leq m_n^*(x^{\text{obs}}) \) for all \( \theta_n^{(t)} \), \( E[m_n^*(x^{\text{obs}})] < \infty \), and \( \sup_{n} E[m_n^*(x^{\text{obs}})1(x^{\text{obs}} \geq \zeta)] \to 0 \) as \( \zeta \to \infty \). In addition, \( \sup_{n} \sup_{x, \theta} s_{n, \Theta_n} |\int \hat{m}_n(\tilde{x})1(m_n(\tilde{x}) > \zeta)h(\tilde{x}^{\text{mis}}|x, \theta)d\tilde{x}^{\text{mis}}| \to 0 \) as \( \zeta \to \infty \).

(b) Define \( \mathcal{F}_n = \{ \int \log f(x_i^{\text{obs}}, z_i^{\text{mis}}|\theta)h(z_i^{\text{mis}}|x_i^{\text{obs}}, \theta_n^{(t)})d\tilde{x}^{\text{mis}}|\theta, \theta_n^{(t)} \in \Theta_n \} \), and \( \mathcal{G}_{n,M} = \{ q(1(m_n^*(x^{\text{obs}}) \leq M))q \in \mathcal{F}_n \} \). Suppose for any fixed \( M, \epsilon \), \( \log N(\epsilon, \mathcal{G}_{n,M}, L_1(\mathbb{P}_n)) = o_p(n) \), where \( \mathbb{P}_n \) is the empirical measure of \( x^{\text{obs}} \), \( L_1(\mathbb{P}_n) \) denotes the \( L_1 \) space of the empirical measure, and \( N(\epsilon, \mathcal{G}_{n,M}, L_1(\mathbb{P}_n)) \) denotes the minimum number of balls \( \{ g : \| g - q \| \leq \epsilon \} \) of radius \( \epsilon \) needed to cover the set \( \mathcal{G}_{n,M} \).

Assume that all elements in \( \cup_{n \geq 1} \mathcal{F}_n \) are uniformly \( L - 1 \) Lipschitz. If the dimensionality of \( \Theta \), denoted as \( p \), is growing at \( O(e^{\alpha n}) \) for some \( \alpha < \frac{1}{2} \), and there exists a constant \( M_0 \) such that \( \Theta_n \subset [-M_0, M_0]^p \), then \( \log N(\epsilon, \mathcal{G}_{n,M}, L_1(\mathbb{P}_n)) = o_p(n) \) holds. Given the compactness of \( \Theta_n \), the above condition is a little bit redundant. This condition would allow \( p \) being potentially much larger than \( n \).

**Lemma 1 (ULLN of \( \hat{Q}_n \)).** Given conditions A1-A3 and A6,

\[
\sup_{\theta_n^{(t)} \in \Theta_n} \sup_{\theta \in \Theta_n} |\hat{G}_n(\theta|\theta_n^{(t)}) - G_n(\theta|\theta_n^{(t)})| \to_p 0.
\]

**Proof.** We follow the proof of Theorem 2.4.3 of Van der Vaart and Wellner (1996). By the symmetrization Lemma 2.3.1 of Van der Vaart and Wellner (1996),

\[
E^* \sup_{\theta, \theta_n^{(t)} \in \Theta_n} |\hat{G}_n(\theta|\theta_n^{(t)}) - G_n(\theta|\theta_n^{(t)})| \leq 2E_{x^{\text{obs}}}E_\epsilon \sup_{q(x) \in \mathcal{F}_n} \| \frac{1}{n} \sum_{i=1}^{n} \epsilon_i q(x_i^{\text{obs}})\| + 2E^*[m_n^*(x^{\text{obs}})1(m_n^*(x^{\text{obs}}) > M)],
\]

18
where $\epsilon_i$ are i.i.d. Rademacher random variables, and $E^*$ denotes the outer expectation.

By assumption (A6)(a), $2E^*[m_n^*(x_{\text{obs}})1(m_n^*(x_{\text{obs}}) > M)] \to 0$.

Fix $x_{\text{obs}}^1, \ldots, x_{\text{obs}}^n$, and let $\mathcal{H}$ be a $\epsilon$-net over $\mathcal{G}_{n,M}$, then

$$E_\epsilon \sup_{q(x) \in \mathcal{H}} \| \frac{1}{n} \sum_{i=1}^n \epsilon_i q(x_{\text{obs}}^i) \| \leq E_\epsilon \sup_{q(x) \in \mathcal{G}_{n,M}} \| \frac{1}{n} \sum_{i=1}^n \epsilon_i q(x_{\text{obs}}^i) \| + \epsilon .$$

By maximal inequality (Lemma 2.2.2 of Van der Vaart and Wellner (1996)),

$$E_\epsilon \sup_{q(x) \in \mathcal{H}} \| \frac{1}{n} \sum_{i=1}^n \epsilon_i q(x_{\text{obs}}^i) \| \leq K \sqrt{1 + \log N(\epsilon, \mathcal{G}_{n,M}, L_1(\mathbb{P}_n)) \sqrt{6/nM} + \epsilon \to P^* \epsilon ,}$$

where $K$ is a generic constant, and $P^*$ denotes outer probability. With assumption A6(b), the conclusion holds. $\square$

Define $B_r(\Theta) = \{ \theta' \| \theta' - \theta \|_2 < r \}$. Thus, $\Theta_n \setminus B_r(\Theta)$ is compact, and $\sup_{\theta \in \Theta_n \setminus B_r(\Theta)} G_n(\theta | \theta_n^{(t)})$ always exists. Define $r_n(\eta | \theta_n^{(t)}) = \inf \{ r | G_n(\theta_n^{(t)} | \theta_n^{(t)}) - \sup_{\theta \in \Theta_n \setminus B_r(\Theta)} G_n(\theta | \theta_n^{(t)}) > \eta \}$. By assumption A5, such $\theta_n^{(t)}$ always exists, and $r_n(\eta | \theta_n^{(t)}) \to 0$ as $\eta \to 0$ for any fixed $\theta_n^{(t)}$. By continuity of $G_n$ with respect to $\theta_n^{(t)}$ and compactness of $\Theta_n$, it is easy to see that $r_n(\eta) = \sup_{\theta_n^{(t)} \in \Theta_n} r_n(\eta | \theta_n^{(t)}) \to 0$ as $\eta \to 0$.

(A7) $r(\eta) = \sup_{n \geq 1} r_n(\eta) \to 0$ as $\eta \to 0$.

Condition (A7) puts a restriction on $G_n(\theta | \theta_n)$ that such functions can not become flatter and flatter around their maxima as $n \to \infty$.

(A8) [Bounds on tails of the imputed data] For any $\theta_n^{(t)} \in \Theta_n$ and $x_{\text{obs}} \in X_{\text{obs}}$, the random variable $x_{\text{mis}} \sim h(\cdot | x_{\text{obs}}, \theta_n^{(t)})$ satisfies:

(a) $\log(f(x_{\text{obs}}, x_{\text{mis}} | \theta)) \in [-M, M]$, for some generic constant $M > 0$.

(b) $\text{var}_{\theta_n^{(t)}}(\log(f(x_{\text{obs}}, x_{\text{mis}} | \theta))) \leq \sigma^2$, for some generic constant $\sigma^2 > 0$.

Condition A8 is more or less a technical condition, which will not affect our practice as $M$ can be set to a big number.

**Theorem 1.** Assume conditions A1-A8 hold. For any $T$ such that $\log T = o_p(n)$, consider $\Theta_n^T$ as an arbitrary subsets of $\Theta_n$ with $T$ elements (can be allowed to have replicates). Then,

(i) $\sup_{\theta_n^{(t)} \in \Theta_n^T} \sup_{\theta \in \Theta_n} | \hat{G}_n(\theta | x, \theta_n^{(t)}) - G_n(\theta | \theta_n^{(t)}) | \to p 0$.

(ii) For each element $\theta_n^{(t)} \in \Theta_n^T$, define $\theta_n^{(t+1)} = \arg \max_{\theta \in \Theta_n} \hat{G}_n(\theta | x, \theta_n^{(t)}) - \frac{1}{n} P(\theta)$, and $\theta_n^* = \arg \max_{\theta \in \Theta_n} G_n(\theta | \theta_n^{(t)})$. Then, $\sup_{\theta_n^{(t)} \in \Theta_n^T} | \theta_n^{(t+1)} - \theta_n^* | \to p 0$.

**Proof.** (i) According to results in Lemma 1, it is suffice to show that $\sup_{\theta_n^{(t)} \in \Theta_n^T} \sup_{\theta \in \Theta_n} | \hat{G}_n(\theta | x, \theta_n^{(t)}) - \hat{G}_n(\theta | \theta_n^{(t)}) | \to p 0$. 

19
Fixing $x_1^{\text{obs}}, \ldots, x_n^{\text{obs}}$, the $\tilde{x}_i^{\text{mis}}|\theta_n^{(t)}$ for all $\theta_n^{(t)} \in \Theta_n^T$ are i.i.d., and they are independent for different $\theta_n^{(t)} \in \Theta_n^T$. Again, we need a ULLN. Denote $\Theta_n^T := \{\theta_n^{(1)}, \ldots, \theta_n^{(T)}\}$. Define $Z_t(\theta) = n|\hat{G}_n(\theta|x, \theta_n^{(t)}) - \tilde{G}_n(\theta|\theta_n^{(t)})|, t = 1, 2, \ldots, T$. It is easy to see that $Z_t \to 0$ pointwisely. In addition, by Bernstein’s inequality, $P(|Z_t| > z) \leq 2\exp\{-\frac{1}{2}n\sigma^2 + \frac{z^2}{2}\}$.

Following Lemma 2.2.10 of Van der Vaart (1996), for Orlicz norm $\psi_1$, we have

$$\max_{1 \leq t \leq T} \sup_{\theta \in \Theta_n} Z_t(\theta) \leq \epsilon + K(M/3 \log(1 + TN(\epsilon, \mathcal{G}_{n,M,L_1(\mathbb{P}_n)})) + \sqrt{n} \sigma \sqrt{\log(1 + TN(\epsilon, \mathcal{G}_{n,M,L_1(\mathbb{P}_n)}))}.$$  

Or equivalently,

$$\sup_{\theta \in \Theta_n, t=1,2,\ldots,T} |\hat{G}_n(\theta|x, \theta_n^{(t)}) - \tilde{G}_n(\theta|\theta_n^{(t)})| \leq \epsilon + K(M/3 \log(1 + TN(\epsilon, \mathcal{G}_{n,M,L_1(\mathbb{P}_n)})))/n$$

$$+ \sigma \sqrt{\log(1 + TN(\epsilon, \mathcal{G}_{n,M,L_1(\mathbb{P}_n)}))/n} \to P^* \epsilon,$$

for any $\epsilon > 0, M > 0$. Thus, $\sup_{\theta \in \Theta_n, t=1,2,\ldots,T} |\hat{G}_n(\theta|x, \theta_n^{(t)}) - \tilde{G}_n(\theta|\theta_n^{(t)})| \to 0$. Consequently, statement (i) holds.

(ii) For any $\eta > 0$, we know that, by (i), there exists $N > 0$ large enough such that for all $n > N,

$$\sup_{\theta_n^{(t)} \in \Theta_n^T} \sup_{\theta \in \Theta_n} |\hat{G}_n(\theta|x, \theta_n^{(t)}) - G_n(\theta|\theta_n^{(t)})| < \eta.$$  

Therefore, by assumption (A7), $\|\theta_n^{(t+1)} - \theta_n^{(t)}\| \leq r(\eta)$ for all $\theta_n^{(t)} \in \Theta_n^T$, which concludes the proof.

Since the imputation step draws random data at each iteration $t$, there is no way to show uniform convergence of $\theta_n^{(t+1)}$ to $\theta_n^{(t)}$ over all possible $\theta_n^{(t)} \in \Theta_n$. However, we are able to prove that the consistency results hold for any sequence of $\theta_n^{(1)}, \ldots, \theta_n^{(T)}$ with $T$ being not too large compared to $e^n$. This is enough for Theorems 2-5. To justify this, we may consider the worst case that the dimension of $\theta_n$ grows with $n$ at an exponential rate $p = O(e^{\alpha n})$ with $\alpha = 1/2$. Then it is easy to see that when $n > 4$, the ratio $T/p$ has an order of

$$O(e^n/p) = O(e^{n-n\alpha}) = O(e^{0.5n}) = O(p^{0.5\sqrt{n}}),$$

which implies that essentially there is no constraint on the setting of $T$. Note that for MCMC simulations, the number of iterations is often set to be a low order polynomial function of the dimension of the state space for a given set of observations.

2. Proof of ergodicity of the Markov chain $\{\theta_n^{(t)}\}$  

Lemma 2 and Lemma 3 are proved in a similar way to Nielsen (2000).

Lemma 2. If A8 holds, the Markov chain $\{\theta_n^{(t)}\}$ is $\varphi$-irreducible and aperiodic.

Proof. Let the state space of $\{\theta_n^{(t)}\}$ be $\Theta_n^T$, which may be smaller than $\Theta_n$. For any $\tilde{x}_1^{\text{mis}}, \ldots, \tilde{x}_n^{\text{mis}}$ and $\theta \in \Theta_n^T$, the assumption A8 implies $P(\tilde{X}_1^{\text{mis}} = \tilde{x}_1^{\text{mis}}, \ldots, \tilde{X}_n^{\text{mis}} = \tilde{x}_n^{\text{mis}}|\theta_n^{(t)} = \theta, x_1^{\text{obs}}, \ldots, x_n^{\text{obs}}) > 0$. Therefore, for any $\theta', \theta \in \Theta_n^T, P(\theta_n^{(t+1)} = \theta'|\theta_n^{(t)} = \theta, x_1^{\text{obs}}, \ldots, x_n^{\text{obs}}) > 0$, where $\theta'$ is uniquely determined by $\tilde{x}_1^{\text{mis}}, \ldots, \tilde{x}_n^{\text{mis}}$ for a given consistency estimator. This verifies both $\varphi$-irreducible and aperiodicity of the Markov chain.
Lemma 3. If A1 and A8 hold, the Markov chain \( \{\theta_n^{(t)}\} \) has the weak Feller property, and compact subsets of \( \Theta \) are small.

Proof. Condition A1 implies that for all measurable sets \( B \),
\[
P(\theta_n^{(t+1)} \in B, \theta_n^{(t)} = \theta') = \int_O h(\bar{x}^{mis}|x^{obs}, \theta') d\bar{x}^{mis} \rightarrow \int_O h(\bar{x}^{mis}|x^{obs}, \theta) d\bar{x}^{mis} = P(\theta_n^{(t+1)} \in B, \theta_n^{(t)} = \theta),
\]
as \( \theta' \rightarrow \theta \), where \( x^{obs} = \{x_1^{obs}, \ldots, x_n^{obs}\} \), \( x^{mis} = \{x_1^{mis}, \ldots, x_n^{mis}\} \) and \( O = \{\bar{x}^{mis}| \theta_n^{(t+1)} \in B, \theta_n^{(t)} = \theta'\} \) denotes the domain of \( x^{mis} \) for which the next iteration IC estimator belongs to the set \( B \). Hence, the Markov chain has the weak Feller property.

(ii) If \( K \subseteq \Theta \) is compact, then for any measurable set \( B \), we can find a point \( \theta' \in K \) such that
\[
\min_{\theta \in K} P(\theta_n^{(t+1)} \in B|\theta_n^{(t)} = \theta) = P(\theta_n^{(t+1)} \in B|\theta_n^{(t)} = \theta').
\]
Hence, \( K \) is small. \( \square \)

Lemma 4. (Dominated Convergence Theorem) If for some random variable \( Z \), \( |X_n| \leq |Z| \) for all \( n \) and \( E|Z| < \infty \), then \( X_n \rightarrow X \) implies that \( EX_n \rightarrow EX \).

The proof of this lemma is based on the Skorohod Representation Theorem and can be found anywhere.

Theorem 2. If A1–A8 hold, then \( \{\theta_n^{(t)}\} \) is almost surely ergodic for sufficiently large \( n \).

Proof. Let \( v(\theta) = C - \frac{1}{n} \log f(x_1^{obs}, \ldots, x_n^{obs}|\theta) \), where \( C \) denotes a constant such that \( v(\theta) \geq 0 \) for all \( \theta \in \Theta_n \). Since \( v(\theta) \) is nonnegative, it can be used to build the drift condition. Define
\[
\Delta v(\theta) = E_h[v(\theta_n^{(t+1)}) - v(\theta_n^{(t)})|\theta_n^{(t)} = \theta]
\]
where \( E_h \) refers to the expectation with respect to the predictive distribution \( h(x^{mis}|x^{obs}, \theta_n^{(t)}). \)

First, we consider the negative of part (I). Let \( M(\theta) = \arg \max_{\theta'} E_{\theta'} \log f(\bar{x}|\theta') \). Then,
\[
-(I) = E_h[\frac{1}{n} \log f(\bar{x}|\theta_n^{(t+1)}) - \frac{1}{n} \log f(\bar{x}|\theta_n^{(t)})] = E_h[\frac{1}{n} \log f(\bar{x}|\theta_n^{(t+1)})] - E_h[\frac{1}{n} \log f(\bar{x}|\theta_n^{(t+1)})] + E_h[\frac{1}{n} \log f(\bar{x}|M(\theta_n^{(t)})) - \frac{1}{n} \log f(\bar{x}|\theta_n^{(t)})].
\]

From part (ii) of Theorem 1, we have
\[
\theta_n^{(t+1)} - M(\theta_n^{(t)}) \rightarrow 0, \quad \text{in probability},
\]
for sufficiently large \( n \). In part (i) of Theorem 1, we established a ULLN for \( \frac{1}{n} \log f(\tilde{x}|\theta) \). Then it implies by Theorem 2.1 of Newey (1991) that \( \frac{1}{n} \log f(\tilde{x}|\theta) \) is stochastically equicontinuous as a random function of \( \theta \) given the compactness of \( \Theta_n \). Further, by the property of stochastic equicontinuity (see e.g., Pötscher and Prucha, 1994), we have

\[
\frac{1}{n} \log f(\tilde{x}\theta_n(t+1)) - \frac{1}{n} \log f(\tilde{x}|M(\theta_n(t))) \to 0, \quad \text{in probability.}
\]

By condition A3, A6 and Lemma 4, we have

\[
E \left[ \frac{1}{n} \log f(\tilde{x}|\theta_n(t+1)) - \frac{1}{n} \log f(\tilde{x}|M(\theta_n(t))) \right] \to 0, \tag{11}
\]

where the expectation is with respect to the joint density function of \( \tilde{x} = (x_{\text{obs}}, x_{\text{mis}}) \). Note that for any \( \theta \in \Theta_n \), we have

\[
E_h \left[ \frac{1}{n} \log f(\tilde{x}|\theta) \right] = \frac{1}{n} \sum_{i=1}^{n} E_h \log f(\tilde{x}_i|\theta) \Delta \frac{1}{n} \sum_{i=1}^{n} g(x_{i,\text{obs}}), \tag{12}
\]

where \( g(x_{i,\text{obs}}) \)'s are iid random variables by assuming that the missing is at random. Then, by (11) and the strong law of large numbers (SLLN), we have

\[
E_h \left[ \frac{1}{n} \log f(\tilde{x}|\theta_n(t+1)) - \frac{1}{n} \log f(\tilde{x}|M(\theta_n(t))) \right] \to 0, \quad \text{a.s.,}
\]

as \( n \to \infty \).

Since \( \Theta_n \) is compact, there exists a constant \( c > 0 \) and a large number \( N \) such that

\[-c < E_h \left[ \frac{1}{n} \log f(\tilde{x}|\theta_n(t+1)) - \frac{1}{n} \log f(\tilde{x}|M(\theta_n(t))) \right] < c, \quad \text{a.s.}, \tag{13}\]

for any \( n > N \) and any \( t > 0 \).

With a similar argument to (12), by invoking SLLN, we can show that there exists a constant \( \delta > 0 \) such that

\[
E_h \left[ \frac{1}{n} \log f(\tilde{x}|M(\theta_n(t))) - \frac{1}{n} \log f(\tilde{x}|\theta_n(t)) \right] \to \delta, \quad \text{a.s.}, \tag{14}
\]

for any \( t > 0 \) as \( n \to \infty \). Combining (13) and (14), we have \(-c - \delta < (I) < c\) holds almost surely for sufficiently large \( n \).

Next, by Jensen’s inequality, we have

\[
(II) = E_h \left[ \frac{1}{n} \log h(\tilde{x}_{\text{mis}}, x_{\text{obs}}|\theta_n(t+1)) - \frac{1}{n} \log h(\tilde{x}_{\text{mis}}, x_{\text{obs}}|\theta_n(t)) \right] \leq \frac{1}{n} \log E_h \left( \frac{h(\tilde{x}_{\text{mis}}|x_{\text{obs}}, \theta_n(t+1))}{h(\tilde{x}_{\text{mis}}|x_{\text{obs}}, \theta_n(t))} \right)
\]

\[
= \frac{1}{n} \log \int h(\tilde{x}_{\text{mis}}|x_{\text{obs}}, \theta_n(t+1))d\tilde{x}_{\text{mis}} = d(\theta_n(t+1)) < \infty,
\]

where the last inequality holds by noting that \( d \) is a continuous function of \( \theta \) and \( \Theta_n \) is compact.

Combining the results of (I) and (II), we have that \( \Delta v(\theta) < c + d \) almost surely for all \( \theta \in \Theta_n \). Choose \( b \) as a negative value less than \( c + d \) and \( D \) as a compact set including \( \{ \theta \in \Theta_n : \Delta v(\theta) \in [b, c + d) \} \). In summary, we have

\[
\Delta v(\theta) \leq \begin{cases} 
  c + d, & \theta \in D, \\
  b, & \theta \in D^c,
\end{cases}
\]

22
almost surely, where \( D^c = \Theta_n \setminus D \). Hence, the strict drift condition \( V_2 \) (Meyn and Tweedie, 2009, p263) is almost surely satisfied.

Since \( (\theta_n^{(t)})_{t \in \mathbb{N}_0} \) also has weak Feller property (see Lemma 3), we can further conclude that an invariant probability measure \( \pi \) almost surely exists for this Markov chain (Meyn and Tweedie, 2009, Theorem 12.3.4). Since \( (\theta_n^{(t)})_{t \in \mathbb{N}_0} \) is \( \varphi \)-irreducible (shown in Lemma 2), \( D \) is a compact set and thus a small set (shown in Lemma 3), and the drift condition \( V_2 \) is stronger than the drift condition \( V_1 \) (Meyn and Tweedie, 2009, p189), we can show that \( (\theta_n^{(t)})_{t \in \mathbb{N}_0} \) is Harris recurrent (Meyn and Tweedie, 2009, Theorem 9.1.8). Since \( (\theta_n^{(t)})_{t \in \mathbb{N}_0} \) is \( \varphi \)-irreducible and has an invariant probability measure \( \pi \), it is also a positive chain (Meyn and Tweedie, 2009, p 230). Therefore, it is a positive Harris recurrent chain (Meyn and Tweedie, 2009, p 231).

Finally, since \( (\theta_n^{(t)})_{t \in \mathbb{N}_0} \) is aperiodic (shown in Lemma 2) and positive Harris recurrent, we can conclude that it is almost surely ergodic (Meyn and Tweedie, 2009, Theorem 13.3.3).

\[ \] 3. Proof of consistency of the IC estimator

To prove the consistency of the IC estimator, we define the mapping

\[
M(\theta) = \arg \max_{\theta'} E_{\theta} \log f_{\theta'}(\bar{x}). \tag{15}
\]

For the C-step, we have \( \theta_n^{(t)} = M(\theta_n^{(t)}) \). Also, \( \theta^* \), the true value of \( \theta_n \), is a fixed point of the mapping. Further, to show that the mean of the stationary distribution of the Markov chain forms a consistent estimate of \( \theta^* \), we make the following assumption.

(A9) The mapping \( M(\theta) \), defined in (15), is differentiable. Let \( \lambda_n(\theta) \) be the largest singular value of \( \partial M(\theta)/\partial \theta \). There exists a number \( \lambda^* < 1 \) such that \( \lambda_n(\theta) \leq \lambda^* \) for all \( \theta \in \Theta_n \) for sufficiently large \( n \) and almost every \( x^{\text{obs}} \)-sequence.

Remark As aforementioned, \( \theta^* \) forms a fixed-point of the mapping \( M(\cdot) \). Let \( \| \cdot \| \) denote the Euclidean-norm. The condition A9 directly implies that

\[
\| M(\theta_n^{(t)}) - \theta^* \| = \| M(\theta_n^{(t)}) - M(\theta^*) \| \leq \lambda^* \| \theta_n^{(t)} - \theta^* \|,
\]

that is, the mapping is a contraction. We note that a continuous application of the mapping, i.e., setting \( \theta_n^{(t+1)} = M(\theta_n^{(t)}) \) for all \( t \), leads to a monotone increase of the expectation \( E_{\theta_n^{(t)}} \log f_{\theta_n^{(t)}}(\bar{x}) \). Since \( E_{\theta_n^{(t)}} \log f_{\theta_n^{(t)}}(\bar{x}) \) attains its maximum at \( E_{\theta^*} \log f_{\theta^*}(\bar{x}) \), it is reasonable to assume that \( M(\theta_n^{(t)}) \) is closer to \( \theta^* \) than \( \theta_n^{(t)} \). This condition should hold for sufficiently large \( n \), at which \( \theta_n^{(t)} \)'s and \( \theta^* \) are all unique as assumed in condition (A5). We note that a similar condition has been used in analysis of the SEM algorithm (Proposition 3, Nielsen, 2000). Some other conditions can potentially be specified based on the fixed-point theory (see e.g., Khamisi and Kirk, 2001).

Theorem 3. If A1–A9 hold, then for sufficiently large \( n \), sufficiently large \( t \), and almost every \( x^{\text{obs}} \)-sequence, we have \( \| \theta_n^{(t)} - \theta^* \| = o_p(1) \). Furthermore, the sample average of the Markov chain also forms a consistent estimate of \( \theta^* \), i.e., \( \| \frac{1}{T} \sum_{t=1}^{T} \theta_n^{(t)} - \theta^* \| = o_p(1) \), as \( n \to \infty \) and \( T \to \infty \).
Proof. From part (ii) of Theorem 1, we have

$$\theta_n^{(t+1)} - M(\theta_n^{(t)}) \to 0, \quad \text{in probability,}$$  \hspace{1cm} (16)

for sufficiently large value $n$. Recall the mapping $M(\theta) = \arg \max_{\theta'} E_\theta \log f(\tilde{x}|\theta')$. Further, we note that $\theta^*$, the true value of $\theta$, forms as a fixed-point of the mapping, i.e., $M(\theta^*) = \theta^*$. By condition $(A_6)$, we have

$$\|M(\theta) - \theta^*\| = \|M(\theta) - M(\theta^*)\| \leq \lambda^* \|\theta - \theta^*\|,$$

for any $\theta \in \Theta_n$, sufficiently large $n$, and almost every $x^{\text{obs}}$-sequence, where $\| \cdot \|$ denotes the Euclidean norm.

By Theorem 2, $\{\theta_n^{(t)}\}$ converges to a stationary distribution. For simplicity, we suppress the supscript $t$, let $\theta_n$ denote the current sample, and let $\theta_n'$ denote the next iteration sample. Therefore,

$$\|\theta_n' - \theta^*\| \leq \|\theta_n' - M(\theta_n)\| + \|M(\theta_n) - \theta^*\| \leq \|\theta_n' - M(\theta_n)\| + \lambda^* \|\theta_n - \theta^*\|.$$

Taking expectation on both sides, we have

$$E\|\theta_n' - \theta^*\| \leq E\|\theta_n' - M(\theta_n)\| + \lambda^* E\|\theta_n - \theta^*\| \leq \frac{1}{1 - \lambda^*} E\|\theta_n' - M(\theta_n)\| = \frac{1}{1 - \lambda^*} o(1) = o(1),$$  \hspace{1cm} (17)

where the second inequality follows from the stationarity of the Markov chain, and the third equality follows from the following reasoning: Since $\Theta_n$ is compact and $M(\theta)$ is a mapping from $\Theta_n$ to $\Theta_n$, $E\|\theta_n' - M(\theta_n)\|^{2+\epsilon} < \infty$ holds for any $\epsilon > 0$; furthermore, it follows from (16), we have

$$E\|\theta_n' - M(\theta_n)\| \to 0,$$  \hspace{1cm} (18)

according to Theorem 25.12 and its corollary (Billingsley, 1986, p.348).

Finally, by Markov’s inequality, we conclude the consistency of $\theta_n^{(t)}$ as an estimator of $\theta^*$. By (17), we have

$$\|E(\theta_n) - \theta^*\| \leq E\|\theta_n - \theta^*\| = o(1),$$

which implies that the mean of the stationary distribution of $\{\theta_n^{(t)}\}$ converges to $\theta^*$ for sufficiently large $n$. Further, by the ergodicity of the Markov chain $\{\theta_n^{(t)}\}$, we conclude the proof. \hfill \Box

4. Proof of ergodicity of the Markov chain for the ICC algorithm

**Theorem 4.** If $A1$–$A8$ hold, the Markov chain $\{(\theta_n^{(1,t)}, \ldots, \theta_n^{(t,k)})\}$ is almost surely ergodic for sufficiently large $n$.

This theorem can be proved in a similar way to Theorem 2 with the detail given in Section 6.1 of the Supplementary Material.
5. Proof of consistency of the ICC estimator

(A9′) Let $M_i$ denote the mapping of the $i$th part of the CC-step, i.e.,
\[ \theta_n^{(i)} = M_i(\theta_n^{(i+1,1)}, \ldots, \theta_n^{(i+1,i-1)}), \]
\[ \theta_n^{(i)} = \theta_n^{(i+1,k)}, \]
Let $M = M_k \circ M_{k-1} \circ \cdots \circ M_1$ denote the joint mapping of $M_1, \ldots, M_k$. Let $\lambda_n(\theta)$ denote the largest singular value of $\partial M(\theta)/\partial \theta$. There exists a number $\lambda^* < 1$ such that $\lambda_n(\theta) \leq \lambda^*$ for all $\theta \in \Theta_n$, all sufficiently large $n$, and almost every $x^{obs}$-sequence.

This condition is reasonable: It is easy to see that a continuous application of the mapping $M$, i.e., applying $M_i$’s in a circular manner, leads to a monotone increase of the function $E_{\theta_n^{(i)}} \log f_{\theta}(\tilde{x})$. Similar to Theorem 3, we can prove the following theorem with the detail given in Section 6.2 of the Supplementary Material.

**Theorem 5.** If A1-A8 and A9′ hold, then for sufficiently large $n$, sufficiently large $t$, and almost every $x^{obs}$-sequence, $\|\theta_n^{(i)} - \theta^*\| = o_p(1)$. Furthermore, the sample average of the Markov chain also forms a consistent estimate of $\theta^*$, i.e., $\|\frac{1}{T} \sum_{t=1}^{T} \theta_n^{(i)} - \theta^*\| = o_p(1)$, as $n \to \infty$ and $T \to \infty$.

**References**


Zhao, Y. and Long, Q. (2013). Multiple imputation in the presence of high-dimensional data. *Statistical Methods in Medical Research, 1*-15.