

Article

Bayesian Matrix Factorization for Electricity Load Imputation

Peili Mao ¹ , Xinqi Li ²  and Ke Li ^{3,*} ¹ School of Electronic Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; 202111081348@std.uestc.edu.cn² Glamorous AI Ltd., Altrincham, Cheshire WA14 2DT, UK³ The Department of Computer Science, University of Exeter, North Park Road, Exeter EX4 4QE, UK

* Correspondence: k.li@exeter.ac.uk

Abstract

The missing values in electricity load are a critical issue in grid applications. The electricity demand time series data are usually large-scaled with complicated patterns and cause difficulties for imputation. This paper presents a Bayesian matrix factorization (BMF)-based imputation method for large-scale electricity load missing value imputation. Through factorizing the original electricity load matrix into two latent matrices, the intrinsic information of the electricity load matrix can be discovered. Two Bayesian inference algorithms, Gibbs sampling and iterated conditional models are applied to solve the BMF model. The affect of the matrix rank to the electricity load imputation task is empirically studied. Experimental results on three real-world electricity load datasets are presented to show the superiority of the proposed method against five benchmark algorithms.

Keywords: electricity load imputation; Bayesian matrix factorization; Bayesian inference

1. Introduction

With the prosperous development of smart grid technologies, a large volume of electricity load data are collected from heterogeneous grid equipment, such as phasor measurement units, infrastructures, and baseload plants. These electricity load data are usually represented as multivariate time series for more than one input source. They have been widely used in a variety of applications including, but not limited to, grid simulations [1], fault-detection [2], load forecasting [3–7], and load management [8]. The performance of electricity load applications is determined by the data quality of input time series, which largely depend on various factors such as the accuracy of the data acquisition and the reliability of the data transmission. For example, noise and outliers encountered at the data acquisition process are usually recorded as missing values during the data pre-processing [9]. Moreover, unanticipated accidents such as power outages at the data transmission process can lead to the loss of data [10].

In statistics, imputation is the process of replacing missing data with substituted values [11]. As a classic imputation technique, interpolation aims to fill the missing values based on the range of observed data points, such as using the previous or next observation in a given range of statistical indicators (i.e., mean and median) [12]. It is often used to impute time series with frequent high patterns. As one of the most interpolation methods, *K*-nearest neighbor (KNN) has been successfully used to impute missing data in power time series [11] by mean predictions. Although KNN is simple and fast, it cannot provide reliable predictions when facing a large percentage of missing data and there exist complex patterns in the time series [13]. Expectation maximization (EM) is another imputation approach that



Academic Editor: Roberto Zivieri

Received: 17 May 2026

Revised: 14 July 2026

Accepted: 15 July 2026

Published: 22 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

takes both the empirical mean and variance into account by maximizing the likelihood in an iterative procedure [14]. However, some empirical studies demonstrated that both the EM and the KNN imputation methods can be biased when the percentage of the missing data is more than 10% [15]. To mitigate this issue, copy-paste imputation (CPI) is a recently proposed method for univariate energy time series with a fixed interval of missing values [16]. Note that these aforementioned approaches are restricted to the univariate time series while the correlation between different time series, in view of the strong dependency among the electricity load multivariate time series in a range of geographical and time-scale ranges, is largely ignored. Multi-task Gaussian process (MTGP) [17] is an ideal choice for multivariate time series, and it has been applied to the financial signal processing in [17]. However, since the conventional MTGP is notorious for its cubic time complexity for the model training and inference, it is hardly useful for large-scale applications.

Nonnegative matrix factorization (NMF) seeks to reveal the hidden structure of non-negative data by decomposing a nonnegative matrix $V \in \mathbb{R}^{m \times n}$ into a product of two low-rank nonnegative latent matrices $W \in \mathbb{R}^{m \times r}$ and $H \in \mathbb{R}^{r \times n}$ [18], where $r \ll m$ and $r \ll n$. As a probabilistic variant of the NMF approach, Bayesian matrix factorization (BMF) has shown to be more effective to capture the correlation between data [19]. It has been applied to a wide range of applications, such movie recommendation system [19,20], gene expression [21], and geographical modeling [22].

Bearing the above mentioned concerns in mind, this paper develops a BMF-based framework to address the multivariate correlation issue and computational bottleneck of the imputation for electricity load data. By factorizing the electricity load data into a time latent and a location latent matrix, respectively, in a probabilistic manner, we can capture the corresponding spatiotemporal relationship. Furthermore, by tweaking the rank of the matrix factorization, the computational complexity and the accuracy of imputation can therefore be controlled in a principled manner. By comparing the performance with five selected peer imputation approaches in the literature on three benchmark datasets, we confirm the outstanding performance of our proposed BMF-based framework for the imputation of electricity load data.

The rest of this paper is organized as follows. Section 2 starts with a problem statement and followed by a pragmatic overview of existing works on imputation for electricity load data. Our proposed methods are delineated in Section 3. The experimental settings are given in Section 4 and the results are discussed in Section 5. Finally, Section 6 concludes this paper and sheds some lights on future directions.

2. Preliminary

This section starts with a formal problem statement from a matrix factorization perspective. Then, we give a pragmatic overview of the existing works on imputation for electricity load data.

2.1. Problem Statement

Let $X \in \mathbb{R}^{N \times T}$ be a multivariate time series of the electricity load data collected from N different locations for T time steps. Each column vector $\mathbf{x}_i = (x_{i,1}, \dots, x_{i,t-1}, x_{i,t}, \dots, x_{i,T})^\top$ where $1 \leq t \leq T$ denote t -th time step and i is the i -th location. Figure 1 shows a graphical representation of the electricity load multivariate time series. The missing values are marked in red.

Let X factorize into a product two latent matrices U and V . The factorization rank k is usually smaller than both row rank and column rank. The goal of matrix factorization is finding a product of two low-rank matrices that minimizes the difference between the original matrix and the factorization results. Different from the traditional matrix factor-

ization applications, the rows (number of time stamps) of the electricity load prediction tasks are much higher than the column numbers (number of locations/infrastructures). The selection of the rank is crucial to the missing electricity load data imputation task as the rank affects both performance and computation cost of the matrix factorization.

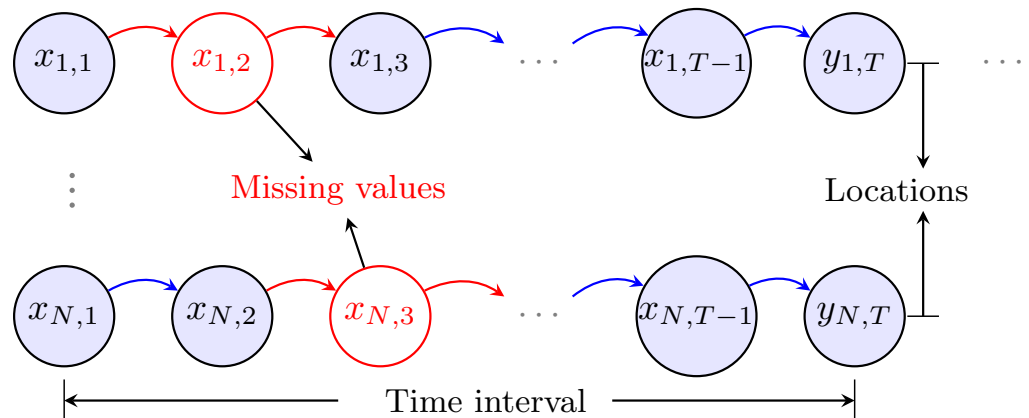


Figure 1. Example of electricity load multivariate time series.

2.2. Related Works

In this subsection, four representative imputation methods with three variants of MTGP are selected to cover the techniques broadly applied in the literature and representative of various imputation strategies.

(i) K-nearest neighbors:

K-nearest neighbors (KNN) is a widely applied linear interpolation method because of its low complexity [11,23]. It computes the interpolating weights w_i and provides a smoother approximation $G(x)$ based on the k nearest neighbors of x , such that:

$$G(\mathbf{x}) = \sum_{i=1}^k w_i(\mathbf{x}) f(\mathbf{x}_i), \quad (1)$$

where $f(x_i)$ is the observed value of x_i . From Equation (1), it can be seen that KNN interpolation gives inaccurate approximation if there is a wide range of neighbor points missed.

(ii) Expected Maximization (EM):

EM is an iterative algorithm to find a maximum likelihood estimation [24,25]. In each iteration, EM imputes the missing values with the most possible values given by the observed data's empirical mean and variance-covariance matrix [14]. The algorithm terminates when the imputed values do not change in new iterations.

(iii) Copy-paste imputation (CPI):

Copy-paste imputation is a recently proposed imputation method for univariate energy time series with an interval of missing values [16]. CPI imputes the missing data with using a Prophet-based method to find weekly patterns, therefore it cannot be applied to the electricity load data having missing values within days. In addition, CPI is designed for univariate time series, the correlations between multivariate electricity load time series can hardly be captured.

(iv) Multi-task Gaussian process (MTGP):

Gaussian process is a non-parametric approach for nonlinear interpolation with placing a prior distribution over latent functions defined for any number of data points. MTGP is an extension of Gaussian process to multiple outputs [26].

Let $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$ denotes a set of N input data, $\mathcal{Y} = \{y_{1,1}, \dots, y_{1N}, \dots, y_{T1}, \dots, y_{TN}\}$ denotes a set of outputs of each input on T tasks. Let $\mathcal{Y}'$ be a subset of $\mathcal{Y}$ denotes the observed data points, the unseen values $\mathcal{Y}^c$ can be predicted by the latent function $f(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_T(\mathbf{x}))^\top$. For an arbitrary $\mathbf{x}$, a GP prior on the latent function is defined as $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$, where mean function $m(\mathbf{x})$ is assumed to be 0, kernel function $K(\mathbf{x}, \mathbf{x}')$ is a positive semi-definite matrix.

With more expressive kernels are developed, MTGP can describe spatial-temporal relations among infinitely many random variables while maintaining statistical dependence between tasks. The linear model of coregionalization (LMC) is the most general positive definite form of a multi-output covariance kernel. The covariance matrices obtained under the LMC assumption can be represented in the form of a sum of separable kernels [27]. A spectral mixture kernel is a positive-definite stationary kernel first introduced in [26] by modeling the spectral density as a weighted mixture of Q square-exponential functions with weights w_q , centers μ_q , and diagonal covariance matrix Σ_q :

$$k(\tau) = \sum_{q=1}^Q w_q \exp\left(-\frac{1}{2} \tau^\top \Sigma_q \tau\right) \cos(\mu_q^\top \tau), \quad (2)$$

where $\mu_q \in \mathbb{R}^n$, $\Sigma = \text{diag}(\sigma_1^q, \dots, \sigma_n^q)$, and $w_q, \sigma_q \in \mathbb{R}_+$.

The MTGP with spectral mixture-linear coregionalization (SM-LMC) model is the first introduces the spectral mixture kernel in the LMC framework [26] to represent complex spectral relationships between tasks. To further capture the cross-phase spectrum between tasks, MTGP with cross-spectral mixture (CSM) kernel is introduced in [28]. An extension work on CSM built on Cramér's Theorem named multi-output spectral mixture (MOSM) kernel is proposed in [29].

In this paper, the multi-task GP with these three spectral-based kernels are used in comparison with the proposed Bayesian matrix factorization approach. Due to the cubic time complexity of inference, the GP models cannot perform on large-scale dataset. Therefore, a month-long interval is extracted from the full dataset for MTGP comparison.

Remark 1. Most, if not all, existing energy load imputation methods in the smart grid literature lack of the ability to capture spatialtemporal relationship for large-scale multivariate time series. However, the real-world applications have the need of handling missing values for large dataset in daily uses.

3. Proposed Method

This section starts with the theoretical foundation of BMF, followed by the architecture of our proposed BMF-based framework for imputation of the electricity load data. At the end, we come up with two probabilistic inference methods of the BMF model.

3.1. The BMF Model

Let $U \in \mathbb{R}^{N \times r}$ be a latent time feature matrix and $V \in \mathbb{R}^{r \times T}$ be a latent location feature matrix, where r denotes the rank. The BMF approach decomposes the input matrix $X \in \mathbb{R}^{N \times T}$ into a product of U and V together with a residual matrix $E \in \mathbb{R}^{N \times T}$ as:

$$X = UV + E, \quad (3)$$

where x_{ij} denotes element on the i -th row and j -th column of matrix X .

In the probabilistic approach, the latent matrices U and V are treated as random variables.

The observed data x_{ij} in matrix X is expressed as a likelihood function comes from the product of U and V with additional Gaussian noise σ . The conditional distribution over the observed electricity load X is:

$$p(X|U, V, \sigma) = \prod_{i=1}^N \prod_{j=1}^T \mathcal{N}(x_{ij}|u_i \times v_j, \sigma^{-1}), \quad (4)$$

where u_i is the i -th row vector of U , v_j is the j -th column vector of V , $\mathcal{N}(x|\mu, \sigma)$ is the density of Gaussian distribution with mean μ and precision σ .

The prior distributions over the U and V can either be assumed to be Gaussian [19]:

$$\begin{aligned} p(U|\mu_U, \lambda_U) &= \prod_{i=1}^N \mathcal{N}(u_i|\mu_U, \lambda_U^{-1}) \\ p(V|\mu_V, \lambda_V) &= \prod_{j=1}^T \mathcal{N}(v_j|\mu_V, \lambda_V^{-1}), \end{aligned} \quad (5)$$

or can be assumed to be independently exponentially distributed with scales $\Gamma_U > 0$ and $\Gamma_V > 0$ [20]:

$$p(U|\Gamma_U) = \prod_{i=1}^N \mathcal{E}(u_i|\Gamma_U), \quad p(V|\Gamma_V) = \prod_{j=1}^T \mathcal{E}(v_j|\Gamma_V), \quad (6)$$

and an inverse Gamma density prior for the noise variance $p(\sigma) = \mathcal{G}^{-1}(\sigma|S_G, \Gamma_G)$ with shape S_G and scale Γ_G .

For the Gaussian prior distributions in (5), in order to reduce Bayesian updating to modifying the hyperparameter, multivariate Gaussian distribution's conjugate prior, Gaussian-Wishart prior, is placed on the location hyperparameters $\{\mu_U, \lambda_U\}$ and time hyperparameters $\{\mu_V, \lambda_V\}$ [19]:

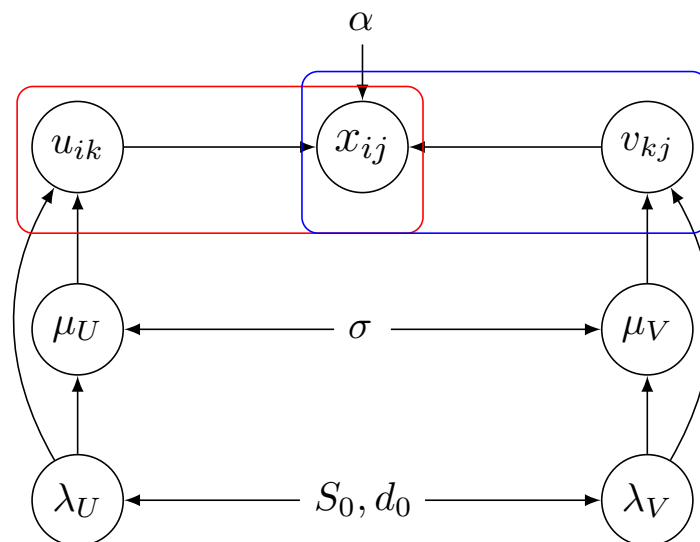
$$\begin{aligned} &p(\{\mu_U, \lambda_U\}|\{\mu_0, \lambda_0\}) \\ &= \mathcal{N}(\mu_U|\mu_0, (\beta_0 \lambda_U)^{-1}) \mathcal{W}(\lambda_U|S_0, d_0) \\ &p(\{\mu_V, \lambda_V\}|\{\mu_0, \lambda_0\}) \\ &= \mathcal{N}(\mu_V|\mu_0, (\beta_0 \lambda_V)^{-1}) \mathcal{W}(\lambda_V|S_0, d_0), \end{aligned} \quad (7)$$

where $\mathcal{W}$ is the Wishart distribution with d_0 degrees of freedom and a scale matrix S_0 with dimension $r \times r$ and a normalizing constant c . Follow the non-informative setting in [19], the degree of freedom $d_0 = r$, S_0 is set as the identity matrix for both location and time hyperparameters, $\mu_0 = 0$. The graphical model representation of BMF is shown in Figure 2.

The predictive distribution of the missing electricity load data x_{ij}^* for the location i at time j can be obtained by a marginal probability over its model parameters and hyperparameters.

Let $\mathcal{H}$ denote the set of the model parameters and hyperparameters, the predictive distribution is [19]:

$$\begin{aligned} p(x_{ij}^*|X, \mathcal{H}_0) &= \iint p(x_{ij}^*|u_i, v_j) p(U, V|X, \mathcal{H}_U, \mathcal{H}_V) \\ &\quad p(\mathcal{H}_U, \mathcal{H}_V|\mathcal{H}_0) d\{U, V\} d\{\mathcal{H}_U, \mathcal{H}_V\}. \end{aligned} \quad (8)$$



$$\forall i = 1, \dots, N; j = 1, \dots, T; k = 1, \dots, r.$$

Figure 2. The graphical model representation of the BMF approach for electricity load imputation.

3.2. Probabilistic Inference

As Equation (8) is analytically intractable, Bayesian inference is required to approximate the predictive distribution. Gibbs sampling is a Markov chain Monte Carlo-based algorithm that was first used in BMF and has been applied to collaborative filtering [30] and user/movie ratings [19]. However, since the electricity load data is dense and may have a high percentage of missing values up to 50% of total amounts of data, Gibbs sampling is computational expensive in this case. As an alternative, a more efficient block coordinate ascent type algorithm named iterated conditional models (ICM) [20] is used.

3.2.1. Gibbs Sampling

Gibbs sampling is a Markov chain Monte Carlo algorithm by alternatively drawing new samples from the conditional posterior densities of the model parameters to estimate the joint posterior [31].

In (5) and (7), since the Gaussian-Wishart prior is used as the prior, the conditional distribution can be expressed analytically. The conditional distributions of U and V have same form, so do their associated hyperparameters $\{\mu_U, \lambda_U\}$ and $\{\mu_V, \lambda_V\}$. In this case, only the conditional distributions of U and its hyperparameters are addressed herein.

For the hyperparameters $\{\mu_U, \lambda_U\}$, the conditional distribution over U and its hyperparameters is:

$$\begin{aligned} p(\mu_U, \lambda_U | U, \mathcal{H}_0) \\ = \mathcal{N}(\mu_U | \mu_U^*, (\beta_U^* \lambda_U)^{-1}) \mathcal{W}(\lambda_U | S_0 U^*, d_U^*), \end{aligned} \quad (9)$$

where

$$\begin{aligned} \mu_U^* &= \frac{\beta_0 \mu_0}{\beta_0 + N}, \quad \beta_U^* = \beta_0 + N, \quad d_U^* = d_0 + N, \\ (S_U^*)^{-1} &= S_0^{-1} + N S_U + \frac{\beta_0 N}{\beta_0 + N} (\bar{U} - \mu_0)(\bar{U} - \mu_0)^\top \\ \bar{U} &= \sum_{i=1}^N u_i, \quad \bar{S} = \frac{1}{N} \sum_{i=1}^N u_i u_i^\top. \end{aligned} \quad (10)$$

The conditional distribution of the location latent matrix U is a factorization of the products of the conditional distributions of its column vectors u_i , for $i \in \{1, \dots, N\}$, i.e.,

$$p(U|X, V, \mathcal{H}_U) = \prod_{i=1}^N p(u_i, X, V, \mathcal{H}_U). \quad (11)$$

Therefore, the parallelization is enabled for each u_i . For each u_i , the conditional distribution $p(u_i|X, V, \mathcal{H}_U, \sigma)$ is a Gaussian distribution $\mathcal{N}(u_i|\mu_i^*, (\lambda_i^*)^{-1})$, where

$$\begin{aligned} \mu_i^* &= (\lambda_i^*)^{-1}(\mu_U \lambda_U + \sigma \sum_{j=1}^M (\mathbf{v}_j x_{ij})) \\ \lambda_i^* &= \lambda_U + \sigma \sum_{j=1}^T v_j v_j^\top. \end{aligned} \quad (12)$$

The full algorithm procedure is detailed in Algorithm 1.

Algorithm 1: Gibbs Sampling

Input: $U_0 \in \mathbb{R}^{N \times r}$, $V_0 \in \mathbb{R}^{r \times T}$,
maximum iteration $I_{\max}$.

Output: X .

```

1 for iterations  $i = 1$  to  $I_{\max}$  do
2   Draw  $U$ 's hyperparameters by eqn. (9):  $\mathcal{H}_U^i \sim p(\mu_U, \lambda_U|U^i, \mathcal{H}_0)$ ;
3   for  $j = 1$  to  $N$  do
4     Draw  $u_j^{i+1} \sim p(u_j|X, V^i, \mathcal{H}_U^i)$ 
5   Draw  $V$ 's hyperparameters:  $\mathcal{H}_V^i \sim p(\mu_V, \lambda_V|V^i, \mathcal{H}_0)$ ;
6   for  $j = 1$  to  $T$  do
7     Draw  $v_j^{i+1} \sim p(v_j|X, U^{i+1}, \mathcal{H}_V^i)$ 
8 return  $X = U_{I_{\max}+1} \times V_{I_{\max}+1}$ .
```

3.2.2. Iterated Conditional Models (ICM)

ICM is an iterative inference algorithm first introduced in [20]. It works similar to Gibbs sampling but it sets the latent matrices U and V and their hyperparameters as the modes of their corresponding distributions, instead of sampling from the conditional posterior distributions. The modes can be obtained through their closed-form expressions and reduces the computational cost.

Consider the exponential distributed case in (6), the conditional distribution of U is propotional to the product of a normal distribution and an exponential distribution, such that:

$$\begin{aligned} p(u_i|X, U_{\setminus(i)}, V, \sigma) &\propto \mathcal{N}(u_i|\mu_{u_i}, \sigma) \times \mathcal{E}(u_i|\Gamma_U) \\ p(v_j|X, V_{\setminus(j)}, U, \sigma) &\propto \mathcal{N}(v_j|\mu_{v_j}, \sigma) \times \mathcal{E}(v_j|\Gamma_V), \end{aligned} \quad (13)$$

where $U_{\setminus(i)}$ and $V_{\setminus(j)}$ denote all elements in U and V , except the i -th row in U and the j -th column in V . μ_{u_i} and μ_{v_j} are the mean vectors for U and V , respectively. σ is the precision.

When the algorithm converges, ICM gives a maximum a posterior estimation. Due to computer's precision issue, ICM may converge to solutions with multiple zero columns. To avoid poor approximation, all the zero entries are added by a small positive value during the iterations as in [32].

4. Experimental Setup

This section introduces the setup of our empirical study including the dataset, peer algorithms and their parameter settings, evaluation metrics, and the statistical tests.

4.1. Datasets

Three real-world annual electricity load datasets are used in the experiments.. Each dataset contains hourly average power values in kilowatt-hour, resulting in 8760 values ($24 \text{ h} \times 365 \text{ days}$) of six to eight locations a year. The reference and industrial dataset are provided by [33]. It consists of a set of 6 and a set of 8 electricity load data collected from reference and industrial end-use sectors respectively in 2020. Island dataset contains the electricity load over year collected from 7 islands.

The information of each dataset is detailed in Table 1. The first month's data of there benchmark dataset are plotted in Figure 3. Two separate experiments are conducted on a short-term (one month) and a long-term (one year) for comparison. The short-term dataset randomly selects an interval of 720 timestamps from the full dataset. For each dataset, a binary mask matrix with the same dimensions as the original data matrix is first generated, where all entries are initialized to one. Then, 10% to 50% of the entries are randomly selected and assigned to zero. The entries corresponding to the zeros in the mask matrix are treated as missing, resulting in the incomplete datasets used for evaluation.

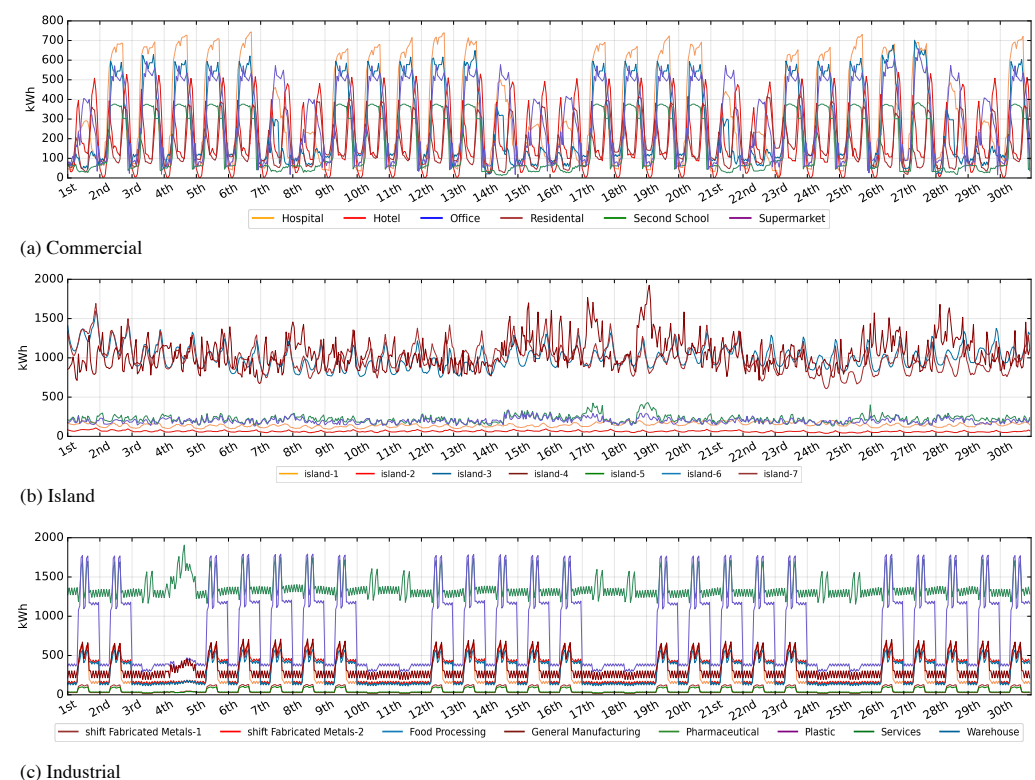


Figure 3. The first month's plots of three benchmark datasets with 720 timestamps.

Table 1. Dataset description.

Statistic	Reference	Island	Industrial
No. of locations	6	7	8
No. of timestamps	8760	8760	8760
Min	0	0	17.4
Max	1000	1975	2387.7

Table 1. *Cont.*

Statistic	Reference	Island	Industrial
Median	233.18	188.50	300.00
Mean	291.7232	381.5900	490.9752
Variance	5.62×10^4	1.32×10^5	2.76×10^5

4.2. Peer Algorithms and Parameter Settings

The performance of the two proposed BMF-based imputation methods, Gibbs sampling and ICM, are compared with five commonly used imputation methods: KNN, EM, and three MTGPs with different kernels introduced in Section 2. The parameter settings of each algorithm are listed as follows.

- Gibbs sampling and ICM: The maximum iteration number is set as 200, which is sufficient to achieve stable convergence according to our empirical study.
- KNN: The nearest neighbours number k is set as the number of locations in each experiment to incorporate information from all available spatial locations.
- EM: The maximum iteration number is set as 1000 for sufficient convergence and the error tolerance is set as 10^{-6} to ensure that the parameter estimation process terminates only when the improvement between consecutive iterations becomes negligible.
- MTGPs: The rank Q of three MTGP methods are set as the number of locations, which follows the original MTGP formulation, where the latent rank is related to the number of output tasks. The optimizers use the most common stochastic gradient descent named Adam. The learning rates are set as 0.1. The maximum iteration numbers are set as 200 to ensure a fair comparison with Gibbs sampling and ICM.

4.3. Performance Metric and Statistical Tests

The performance of the algorithms are evaluated by the root mean squared error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2} \quad (14)$$

where x_i is the observed value, $\hat{x}_i$ is the predicted value.

Besides, the CPU runtime are recorded to compare the computational cost of the algorithms. All the experiments are conducted on MacBook with M1 chip with 8-core CPU and 16 GB ram.

Wilcoxon signed-rank test [34] is applied for statistical interpretation of the significance of the comparison results. It is a non-parametric statistical test without making assumptions about the data's underlying distribution. The significance level is set to 0.05 in the experiments.

To ensure that the observed performance differences are not caused by trivial variations, we employ the A_{12} statistic [35] as an effect size measure to quantify the probability that one algorithm outperforms another. Specifically, for a given pair of peer algorithms, $A_{12} = 0.5$ indicates that the two algorithms have equivalent performance. When $A_{12} > 0.5$, the first algorithm is considered to perform better than the second one in more than 50% of cases. Following the commonly adopted interpretation of effect sizes, $0.56 \leq A_{12} < 0.64$ represents a small effect, $0.64 \leq A_{12} < 0.71$ represents a medium effect, and $A_{12} \geq 0.71$ indicates a large effect.

5. Results and Discussion

The empirical study is driven by addressing the following four research questions (RQs).

- **RQ1:** What is the impact of matrix factorization rank to BMF-based imputation?
- **RQ2:** Does the BMF framework work for different patterns of electricity load dataset?
- **RQ3:** Does the percentage of missing values affect the BMF-based imputation, and how is the performance comparing to the benchmark algorithms' performances?
- **RQ4:** Does the BMF-based method scale to long-term electricity load dataset?

5.1. Impact of Factorization Rank

Figure 4 shows the performance of Gibbs sampling and ICM with ranks ranging from the minimum number of locations 6 to 20 on three datasets. Subfigures (a) and (b) have similar RMSE performance when increase the percentage of missing values from 10% to 50%.

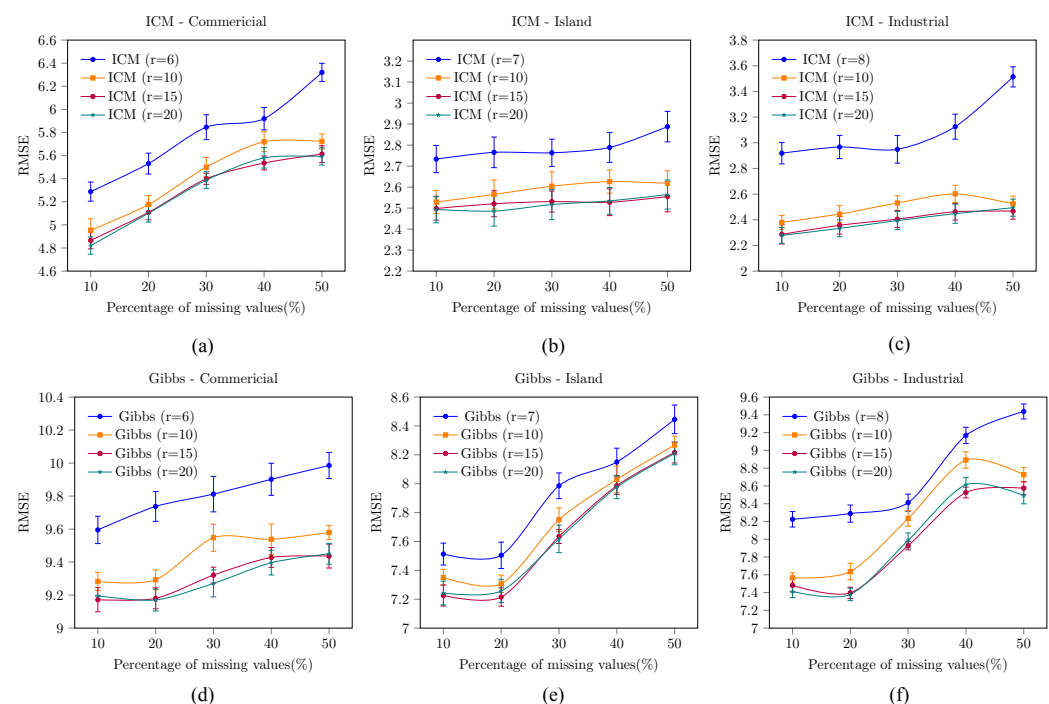


Figure 4. Performance of Gibbs sampling and ICM with different ranks on three datasets.

Table 2 records the runtime of Gibbs sampling and ICM on three benchmark datasets with 50% missing values with different rank values. As the BMF computes the factorization of the original matrix, the number of missing values does not affect the runtime. The table shows that the runtime increases while the rank increases. Since more elements of U and V are required to be inferred during the algorithm process, more computational resources are required.

Table 2. Runtime (seconds) of Gibbs Sampling (first row in each sub-block) and ICM (second row in each sub-block) on three benchmark datasets with 50% missing values.

Dataset	Algorithm	Rank = (6, 7, 8)	10	15	20
Commercial	Gibbs	72.28 (6.26)	126.14 (5.37)	195.73 (7.37)	261.69 (6.53)
	ICM	1.6 (0.01)	2.49 (0.01)	3.65(0.01)	4.91 (0.01)
Island	Gibbs	69.23 (5.76)	100.064 (6.11)	155.07 (6.44)	210.19 (8.02)
	ICM	1.99 (0.01)	2.73 (0.02)	4.01 (0.01)	5.34 (0.01)
Industrial	Gibbs	76.98 (6.20)	99.92 (5.85)	153.672 (7.61)	226.20 (7.92)
	ICM	2.33 (0.01)	2.90 (0.01)	4.14 (0.01)	5.35 (0.01)

Since the RMSE is not improved obviously from rank 15 to rank 20 and more runtime is required, the final ranks for Gibbs sampling and ICM are set as 15 on these three benchmark datasets in the later experiments.

Response to RQ1: *From the experiment in this subsection, we find that the increment of the rank (from $r = 6$ to $r = 15$) leads to decrement RMSE result till $r = 15$ with 200 fixed iterations. In addition, a high-rank value leads to high computational time.*

5.2. Performance Evaluation of the Effectiveness of the BMF Imputation

Table 3 records the RMSE performance of seven imputation algorithms on three short-term (one month) electricity load datasets with missing values ranging from ten percent to fifty percent. It can be observed that ICM performs the best among all. It competes the second-best results by improving the RMSE up to 45.75% on ‘commercial’, 71.53% percent on ‘island’, and 68.74% percent on ‘industrial’. The MTGP with CSM and MOSM achieves better performance than Gibbs sampling when the percentage of missing values is lower than 30% on ‘commercial’ and ‘industrial’ datasets, but are all competed by the two BMF inference methods on ‘island’. When the number of missing values increases, Gibbs Sampling and ICM are less affected while the RMSEs of the other comparison algorithms increase dramatically, especially for KNN and EM. ICM obtains A_{12} values larger than 0.71 in all comparisons with other peer algorithms, indicating a large performance advantage. These results confirm that the superiority of ICM is not due to trivial effects, but reflects a substantial and statistically meaningful improvement over the compared methods.

Table 3. RMSE on three short-term ($T = 720$) datasets with missing values from 10% to 50%. Best results are boldfaced, second best are underlined, and † indicates statistical significance (Wilcoxon’s rank-sum test, 5% level), L indicates large effect size.

Datasets	Algorithms	10%	20%	30%	40%	50%
Commercial	KNN	67.18 _(1.19) ^{†,L}	76.21 _(1.25) ^{†,L}	90.78 _(2.96) ^{†,L}	89.65 _(3.42) ^{†,L}	92.78 _(3.08) ^{†,L}
	EM	79.23 _(1.50) ^{†,L}	87.81 _(2.88) ^{†,L}	98.67 _(4.01) ^{†,L}	103.84 _(3.52) ^{†,L}	110.84 _(3.25) ^{†,L}
	MTGP+SM-LMC	8.14 _(0.32) ^{†,L}	8.58 _(0.57) ^{†,L}	9.43 _(0.48) ^{†,L}	10.86 _(0.65) ^{†,L}	12.22 _(0.34) ^{†,L}
	MTGP+CSM	<u>7.22_(0.40)</u> ^{†,L}	7.60 _(0.42) ^{†,L}	<u>8.25_(0.70)</u> ^{†,L}	9.50 _(0.29) ^{†,L}	10.87 _(0.62) ^{†,L}
	MTGP+MOSM	7.31 _(0.45) ^{†,L}	<u>7.74_(0.47)</u> ^{†,L}	8.37 _(0.52) ^{†,L}	9.63 _(0.51) ^{†,L}	11.28 _(0.52) ^{†,L}
	Gibbs	8.81 _(0.37) ^{†,L}	9.12 _(0.37) ^{†,L}	9.25 _(0.33) ^{†,L}	<u>9.26_(0.30)</u> ^{†,L}	<u>9.37_(0.33)</u> ^{†,L}
	ICM	3.68_(0.36)	3.99_(0.36)	4.44_(0.35)	4.36_(0.38)	4.42_(0.28)
Island	KNN	86.92 _(2.89) ^{†,L}	94.51 _(2.46) ^{†,L}	105.92 _(4.51) ^{†,L}	107.41 _(3.77) ^{†,L}	108.90 _(3.11) ^{†,L}
	EM	50.49 _(3.35) ^{†,L}	52.45 _(2.67) ^{†,L}	66.28 _(3.50) ^{†,L}	63.37 _(5.02) ^{†,L}	69.45 _(4.59) ^{†,L}
	MTGP+SM-LMC	7.51 _(0.22) ^{†,L}	8.27 _(0.48) ^{†,L}	8.68 _(0.57) ^{†,L}	10.25 _(0.63) ^{†,L}	11.15 _(0.55) ^{†,L}
	MTGP+CSM	6.93 _(0.33) ^{†,L}	7.37 _(0.45) ^{†,L}	8.04 _(0.42) ^{†,L}	9.12 _(0.23) ^{†,L}	10.80 _(0.44) ^{†,L}
	MTGP+MOSM	7.09 _(0.30) ^{†,L}	7.66 _(0.29) ^{†,L}	7.82 _(0.47) ^{†,L}	8.80 _(0.39) ^{†,L}	10.43 _(0.37) ^{†,L}
	Gibbs	<u>7.23_(0.38)</u> ^{†,L}	<u>7.12_(0.30)</u> ^{†,L}	<u>7.46_(0.31)</u> ^{†,L}	<u>7.97_(0.33)</u> ^{†,L}	<u>8.11_(0.24)</u> ^{†,L}
	ICM	2.33_(0.30)	2.33_(0.29)	2.42_(0.23)	2.49_(0.30)	2.46_(0.22)

Table 3. *Cont.*

Datasets	Algorithms	10%	20%	30%	40%	50%
Industrial	KNN	33.00 _(0.82) ^{†,L}	47.53 _(1.07) ^{†,L}	77.49 _(2.82) ^{†,L}	101.68 _(4.06) ^{†,L}	106.34 _(4.96) ^{†,L}
	EM	38.63 _(1.01) ^{†,L}	45.28 _(1.52) ^{†,L}	60.85 _(2.09) ^{†,L}	75.23 _(2.93) ^{†,L}	100.34 _(3.56) ^{†,L}
	MTGP+SM-LMC	7.38 _(0.18) ^{†,L}	7.70 _(0.21) ^{†,L}	7.76 _(0.30) ^{†,L}	9.91 _(0.28) ^{†,L}	0.81 _(0.38) ^{†,L}
	MTGP+CSM	7.25 _(0.15) ^{†,L}	7.27 _(0.19) ^{†,L}	7.27 _(0.26) ^{†,L}	8.25 _(0.35) ^{†,L}	9.02 _(0.49) ^{†,L}
	MTGP+MOSM	7.17 _(0.22) ^{†,L}	7.04 _(0.25) ^{†,L}	7.64 _(0.17) ^{†,L}	8.22 _(0.28) ^{†,L}	8.30 _(0.41) ^{†,L}
	Gibbs	7.37 _(0.31) ^{†,L}	7.23 _(0.30) ^{†,L}	8.02 _(0.32) ^{†,L}	8.44 _(0.27) ^{†,L}	8.53 _(0.30) ^{†,L}
	ICM	2.13 _(0.27)	2.21 _(0.29)	2.35 _(0.29)	2.38 _(0.29)	2.35 _(0.31)

Response to RQ2: The experimental results show that BMF-based imputation methods work consistently well on three electricity load datasets with different patterns compared to the benchmark algorithms.

Response to RQ3: From the observations in this experiment, we see that the increment of missing values still leads to a decrement of the BMF's and Gibbs' RMSE results. The decrement ratio of the proposed methods is similar to MTGP methods but far lower than KNN and EM.

5.3. Performance Evaluation of the BMF Imputation on Long-Term Electricity Load Dataset

Table 4 records the RMSE performance on a long-term (one year) electricity load dataset with missing values ranging from ten percent to fifty percent. Three GP-based imputation methods are not recorded in this table since they cannot be computed on a large scale. Although the RMSE values are slightly increased for ICM and Gibbs sampling on these long-term datasets than on the short-term datasets, the results are similar to Table 3. ICM defeats all the benchmark algorithms and Gibbs sampling. Gibbs sampling achieves the second best on all the datasets with the absence of MTGP models. To further evaluate whether the performance improvements on the long-term dataset are meaningful, we also analyze the A_{12} effect size between ICM and the competing algorithms. The results show that ICM achieves A_{12} values have large effect sizes in all comparisons with the available peer algorithms, indicating that its improvements are not caused by trivial variations.

Table 4. RMSE on three long-term ($T = 8760$) datasets with missing values ranging from 10% to 50%.

Datasets	Algorithms	10%	20%	30%	40%	50%
Commercial	KNN	69.24 _(1.24) ^{†,L}	119.97 _(3.05) ^{†,L}	137.57 _(3.30) ^{†,L}	144.39 _(3.67) ^{†,L}	147.06 _(3.93) ^{†,L}
	EM	68.73 _(1.35) ^{†,L}	92.56 _(2.79) ^{†,L}	111.19 _(3.51) ^{†,L}	127.94 _(3.11) ^{†,L}	136.04 _(3.68) ^{†,L}
	MTGP+SM-LMC	-	-	-	-	-
	MTGP+CSM	-	-	-	-	-
	MTGP+MOSM	-	-	-	-	-
	Gibbs	9.17 _(0.44) ^{†,L}	9.18 _(0.38) ^{†,L}	9.39 _(0.35) ^{†,L}	9.49 _(0.29) ^{†,L}	9.44 _(0.42) ^{†,L}
	ICM	4.87 _(0.35)	5.11 _(0.24)	5.40 _(0.13)	5.54 _(0.34)	5.61 _(0.20)

Table 4. Cont.

Datasets	Algorithms	10%	20%	30%	40%	50%
Island	KNN	81.60 _(1.77) ^{†,L}	106.85 _(2.10) ^{†,L}	110.61 _(3.52) ^{†,L}	120.30 _(2.84) ^{†,L}	127.61 _(2.55) ^{†,L}
	EM	82.49 _(1.41) ^{†,L}	92.44 _(3.54) ^{†,L}	102.07 _(2.55) ^{†,L}	108.63 _(2.90) ^{†,L}	114.56 _(3.69) ^{†,L}
	MTGP+SM-LMC	-	-	-	-	-
	MTGP+CSM	-	-	-	-	-
	MTGP+MOSM	-	-	-	-	-
	Gibbs	7.23 _(0.25) ^{†,L}	7.22 _(0.28) ^{†,L}	7.63 _(0.53) ^{†,L}	7.99 _(0.36) ^{†,L}	8.22 _(0.29) ^{†,L}
	ICM	2.50 _(0.30)	2.52 _(0.22)	2.52 _(0.49)	2.53 _(0.43)	2.55 _(0.39)
Industrial	KNN	37.83 _(0.98) ^{†,L}	71.13 _(2.32) ^{†,L}	101.85 _(3.66) ^{†,L}	117.61 _(3.00) ^{†,L}	120.49 _(3.18) ^{†,L}
	EM	31.06 _(1.20) ^{†,L}	55.07 _(1.89) ^{†,L}	82.93 _(2.56) ^{†,L}	104.48 _(2.54) ^{†,L}	114.18 _(3.60) ^{†,L}
	MTGP+SM-LMC	-	-	-	-	-
	MTGP+CSM	-	-	-	-	-
	MTGP+MOSM	-	-	-	-	-
	Gibbs	7.48 _(0.19) ^{†,L}	7.40 _(0.41) ^{†,L}	7.93 _(0.32) ^{†,L}	8.53 _(0.23) ^{†,L}	8.57 _(0.25) ^{†,L}
	ICM	2.29 _(0.29)	2.36 _(0.18)	2.41 _(0.20)	2.47 _(0.23)	2.47 _(0.35)

Table 5 records the runtime of each run in Table 4. ICM takes the least time to compute when the percentage of missing values is greater than 30%, and KNN takes the least time to compute when it is less or equal to 30%. Both Gibbs sampling and ICM have almost the same runtime for all percentages of missing values, whereas the runtime of KNN and EM increases when the percentage of missing values increase. As the MTGPs have a cubic time complexity, they averagely take 2 h to compute on the short-term dataset and extremely long time to compute on the long-term dataset. The experiments were not conducted for the MTGPs on the long-term dataset, and thus the runtime are not recorded for them.

Table 5. Runtime (seconds) on three long-term datasets with missing values from 10% to 50%.

Datasets	Algorithms	10%	20%	30%	40%	50%
Commercial	KNN	1.41 _(0.00)	2.45 _(0.00)	3.52 _(0.00)	4.37 _(0.00)	4.95 _(0.00)
	EM	4.35 _(0.01)	12.52 _(0.08)	24.01 _(1.33)	45.29 _(3.53)	83.83 _(5.08)
	MTGPs	-	-	-	-	-
	Gibbs	190.97 _(10.24)	188.02 _(7.28)	185.55 _(10.19)	179.33 _(11.43)	195.73 _(13.54)
	ICM	3.64 _(0.00)	3.65 _(0.00)	3.65 _(0.00)	3.64 _(0.00)	3.65 _(0.00)
Island	KNN	1.64 _(0.00)	2.57 _(0.00)	4.00 _(0.00)	4.85 _(0.00)	5.18 _(0.00)
	EM	4.83 _(0.01)	11.50 _(0.05)	23.24 _(1.72)	46.97 _(1.88)	91.19 _(3.35)
	MTGPs	-	-	-	-	-
	Gibbs	157.24 _(8.68)	152.13 _(9.15)	160.90 _(11.21)	152.87 _(10.54)	155.07 _(11.95)
	ICM	4.01 _(0.00)	4.00 _(0.00)	4.01 _(0.00)	4.01 _(0.00)	4.01 _(0.00)
Industrial	KNN	1.72 _(0.00)	2.84 _(0.00)	4.13 _(0.00)	5.25 _(0.00)	5.77 _(0.00)
	EM	4.45 _(0.01)	11.48 _(0.06)	23.66 _(0.32)	45.77 _(1.79)	89.52 _(4.70)
	MTGPs	-	-	-	-	-
	Gibbs	169.73 _(9.66)	155.02 _(10.03)	165.88 _(8.69)	158.52 _(11.71)	153.67 _(10.53)
	ICM	4.14 _(0.00)	4.13 _(0.00)	4.13 _(0.00)	4.14 _(0.00)	4.14 _(0.00)

Response to RQ4: From the experimental results recorded herein in comparison to the previous subsections, we conclude that ICM is able to scale to long-term electricity load dataset with a small increment of RMSE and runtime.

6. Conclusions

This paper presents a BMF model for imputing electricity load missing data for solving general smart grid data missing problems. The spatiotemporal relationship can be well captured by factorizing the electricity multivariate time series into two latent matrices. Two Bayesian inference methods, Gibbs sampling and ICM, are applied to solve the BMF model. Due to the large-scale of the electricity load dataset, Gibbs sampling cannot perform enough samplings to infer a good distribution during the algorithm procedure, and the performance is limited. Compared to Gibbs sampling, ICM is fast as it obtains the latent matrices and their hyperparameters by taking the local maximum of the joint probability with their closed form.

The experimental results show that BMF using the ICM inference method outperforms all the benchmark algorithms and BMF using the Gibbs sampling method. The percentage of missing values does not affect the runtime of Gibbs sampling of ICM as the algorithms find the factorization of the full data matrix in all runs no matter with the percentage of missing values. In addition, when the multivariate time series patterns are different (e.g., ‘island’ dataset), BMF is able to capture the intrinsic information by the latent matrices, unlike MTGP is better in filling time series with similar trends. Future investigations may aim at developing online Bayesian approaches based on other models or inference methods for online smart grid applications.

Author Contributions: Conceptualization, P.M., X.L. and K.L.; Methodology, P.M. and X.L.; Software, P.M.; Validation, P.M. and X.L.; Formal analysis, P.M.; investigation, X.L.; Writing—original draft preparation, P.M. and X.L.; writing—review and editing, K.L.; supervision, K.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: <https://github.com/maopl/BNMF/tree/main/3-BNMF-imputation> (accessed on 7 April 2026).

Conflicts of Interest: Author Xinqi Li was employed by the company Glamorous AI Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Rivera, J.; Nasirifard, P.; Leimhofer, J.; Jacobsen, H. Automatic Generation of Real Power Transmission Grid Models From Crowdsourced Data. *IEEE Trans. Smart Grid* **2019**, *10*, 5436–5448. [\[CrossRef\]](#)
2. Gilanifar, M.; Cordova, J.; Wang, H.; Stifter, M.; Ozguven, E.; Strasser, T.; Arghandeh, R. Multi-Task Logistic Low-Ranked Dirty Model for Fault Detection in Power Distribution System. *IEEE Trans. Smart Grid* **2020**, *11*, 786–796. [\[CrossRef\]](#)
3. Fiot, J.; Dinuzzo, F. Electricity Demand Forecasting by Multi-Task Learning. *IEEE Trans. Smart Grid* **2016**, *9*, 544–551. [\[CrossRef\]](#)
4. Kong, W.; Z, D.; Hill, D.; Luo, F.; Xu, Y. Short-Term Residential Load Forecasting Based on Resident Behaviour Learning. *IEEE Trans. Power Syst.* **2018**, *33*, 1087–1088. [\[CrossRef\]](#)
5. Chen, K.; Chen, K.; Wang, Q.; He, Z.; Hu, J.; He, J. Short-Term Load Forecasting With Deep Residual Networks. *IEEE Trans. Smart Grid* **2019**, *10*, 3943–3952. [\[CrossRef\]](#)
6. Luo, J.; Hong, T.; Fang, S.C. Robust Regression Models for Load Forecasting. *IEEE Trans. Smart Grid* **2019**, *10*, 5397–5404. [\[CrossRef\]](#)

7. Aprillia, H.; Yang, H.; Huang, C. Statistical Load Forecasting Using Optimal Quantile Regression Random Forest and Risk Assessment Index. *IEEE Trans. Smart Grid* **2021**, *12*, 1467–1480. [[CrossRef](#)]
8. Baniasadi, A.; Habibi, D.; Bass, O.; Masoum, M.A. Optimal Real-Time Residential Thermal Energy Management for Peak-Load Shifting with Experimental Verification. *IEEE Trans. Smart Grid* **2019**, *10*, 5587–5599. [[CrossRef](#)]
9. Akouemo, H.N.; Povinelli, R.J. Data Improving in Time Series Using ARX and ANN Models. *IEEE Trans. Power Syst.* **2017**, *32*, 3352–3359. [[CrossRef](#)]
10. King, C.; Strapp, J. Chapter 11—Software infrastructure and the smart grid. In *Smart Grid: Integrating Renewable, Distributed and Efficient Energy*; Academic Press: Cambridge, MA, USA, 2012; pp. 259–288.
11. Kim, M.; Park, S.; Lee, J.; Joo, Y.; Choi, J. Learning-Based Adaptive Imputation Method with kNN Algorithm for Missing Power Data. *Energies* **2017**, *10*, 1668. [[CrossRef](#)]
12. Steffensen, J.F. *Interpolation*; Dover: Mineola, NY, USA, 2013.
13. Beretta, L.; Santaniello, A. Nearest neighbor imputation algorithms: A critical evaluation. *BMC Med. Inform. Decis. Mak.* **2016**, *16*, 74. [[CrossRef](#)] [[PubMed](#)]
14. Do, C.; Batzoglou, S. What is the expectation maximization algorithm? *Nat. Biotechnol.* **2008**, *26*, 897–899. [[CrossRef](#)] [[PubMed](#)]
15. Bennett, D. How can I deal with missing data in my study? *Aust. N. Z. J. Public Health* **2001**, *25*, 464–469. [[CrossRef](#)]
16. Weber, M.; Turowski, M.; Çakmak, H.K.; Mikut, R.; Kühnapfel, U.G.; Hagenmeyer, V. Data-Driven Copy-Paste Imputation for Energy Time Series. *IEEE Trans. Smart Grid* **2021**, *12*, 5409–5419. [[CrossRef](#)]
17. Wolff, T.; Cuevas, A.; Tobar, F. Gaussian Process Imputation of Multiple Financial Series. In Proceedings of the 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 8444–8448. [[CrossRef](#)]
18. Lee, H.; Seung, D. Learning the parts of objects by nonnegative matrix factorization. *Nature* **1999**, *401*, 788–791. [[CrossRef](#)] [[PubMed](#)]
19. Salakhutdinov, R.; Mnih, A. Bayesian Probabilistic Matrix Factorization Using Markov Chain Monte Carlo. In Proceedings of the 25th International Conference on Machine Learning, ICML '08, Helsinki, Finland, 5–9 July 2008; pp. 880–887. [[CrossRef](#)]
20. Schmidt, M.; Winther, O.; Hansen, L. Bayesian Non-Negative Matrix Factorization. In *Independent Component Analysis and Signal Separation Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 540–547.
21. T.D., S.; Gao, T.; Fertig, E. CoGAPS 3: Bayesian non-negative matrix factorization for single-cell analysis with asynchronous updates and sparse data structures. *BMC Bioinform.* **2020**, *21*, 453.
22. Liu, B.; Xiong, H.; Papadimitriou, S.; Fu, Y.; Yao, Z. A General Geographical Probabilistic Factor Model for Point of Interest Recommendation. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 1167–1179. [[CrossRef](#)]
23. Sibson, R. A brief description of natural neighbor interpolation (Chapter 2). In *Interpreting Multivariate Data*; Wiley: Hoboken, NJ, USA, 1981; pp. 21–36.
24. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Maximum Likelihood from Incomplete Data Via the EM Algorithm. *J. R. Stat. Soc. Ser. B (Methodol.)* **1977**, *39*, 1–22. [[CrossRef](#)]
25. Wang, M.; Tsai, C.; Lin, W. Towards missing electric power data imputation for energy management systems. *Expert Syst. Appl.* **2021**, *174*, 114743. [[CrossRef](#)]
26. Wilson, A.; Adams, R. Gaussian Process Kernels for Pattern Discovery and Extrapolation. In *Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 17–19 June 2013*; PMLR (Proceedings of Machine Learning Research): New York, NY, USA, 2013; Volume 28:3, pp. 1067–1075.
27. Álvarez, M.A.; Rosasco, L.; Lawrence, N.D. Kernels for Vector-Valued Functions: A Review. *Found. Trends Mach. Learn.* **2012**, *4*, 195–266. [[CrossRef](#)]
28. Ulrich, K.R.; Carlson, D.E.; Dzirasa, K.; Carin, K. GP Kernels for Cross-Spectrum Analysis. In *Proceedings of the Advances in Neural Information Processing Systems*; Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2015; Volume 28.
29. Parra, G.; Tobar, F. Spectral Mixture Kernels for Multi-Output Gaussian Processes. In *Proceedings of the Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
30. Salakhutdinov, R.; Mnih, A. Probabilistic Matrix Factorization. In *Proceedings of the Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 3–6 December 2007*; Curran Associates, Inc.: Red Hook, NY, USA, 2007; pp. 1257–1264.
31. Neal, R.M. *Probabilistic Inference Using Markov Chain Monte Carlo Methods*; Technical Report CRG-TR-93-1; University of Toronto: Toronto, ON, Canada, 1993.
32. Thomas, B.; Jes, F.; Pietro, L. Comparative Study of Inference Methods for Bayesian Nonnegative Matrix Factorisation. In Proceedings of the European Conference on Principles of Data Mining and Knowledge Discovery, Skopje, Macedonia, 18–22 September 2017.

33. Angizeh, F.; Jafari, A.G.M. *Dataset on Hourly Load Profiles for a Set of 24 Facilities from Industrial, Commercial, and Residential End-Use Sectors, V1*; Mendeley Data: London, UK, 2020.
34. Wilcoxon, F. *Individual Comparisons by Ranking Methods*; Springer: New York, NY, USA, 1992; pp. 192–202.
35. Vargha, A.; Delaney, H.D. A Critique and Improvement of the ‘CL’ Common Language Effect Size Statistics of McGraw and Wong. *J. Educ. Behav. Stat.* **2000**, *25*, 101–132. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.