Transportation Research Part C 98 (2019) 73–84

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation

Xinyu Chen[a], Zhaocheng He[a], Lijun Sun[b,*]

[a] Guangdong Provincial Key Laboratory of Intelligent Transportation Systems, Research Center of Intelligent Transportation System, Sun Yat-Sen University, Guangzhou, Guangdong 510006, China

[b] Department of Civil Engineering and Applied Mechanics, McGill University, Montreal, Quebec H3A 0C3, Canada

ARTICLE INFO

Keywords:
Spatiotemporal traffic data
Tensor decomposition
Bayesian inference
Markov chain Monte Carlo
Missing data imputation
Data representation

ABSTRACT

The missing data problem is inevitable when collecting traffic data from intelligent transportation systems. Previous studies have shown the advantages of tensor completion-based approaches in solving multi-dimensional data imputation problems. In this paper, we extend the Bayesian probabilistic matrix factorization model by Salakhutdinov and Mnih (2008) to higher-order tensors and apply it for spatiotemporal traffic data imputation tasks. In doing so, we care about not only the model configuration but also the representation of data (i.e., matrix, third-order tensor and fourth-order tensor). Using a nine-week spatiotemporal traffic speed data set (road segment × day × time of day) collected in Guangzhou, China, we evaluate the performance of this fully Bayesian model and explore how different data representations affect imputation performance through extensive experiments. The results show the proposed model can produce accurate imputations even under temporally correlated data corruption. Our experiments also show that data representation is a crucial factor for model performance, and a third-order tensor structure outperforms the matrix and fourth-order tensor representations in preserving information in our data set. We hope this work could give insights to practitioners when performing spatiotemporal data imputation tasks.

1. Introduction

With the development and application of intelligent transportation systems, large quantities of urban traffic data are collected on a continuous basis from various sources, such as loop detectors, cameras, and floating vehicles. These data sets capture the underlying states and dynamics of transportation networks and the whole system and become beneficial to many traffic operation and management applications, including routing, signal control, travel time prediction, and so on. In general, traffic data register full spatial and temporal features (where and when the data is collected), together with some other site-specific attributes. Usually, we can organize the spatiotemporal traffic data into a multi-dimensional structure. For example, a link's flow/speed information over a week can be summarized into seven daily time series. Combined with information from other links in a city, the overall spatiotemporal data can be structured as a multi-dimensional array (road segment × day × time of day), which is often referred to as a tensor.

A common drawback that undermines the use of such spatiotemporal data is the "missingness" problem, which may result from various factors such as hardware/software failure, network communication problems, and zero/limited reports from floating/crowdsourcing systems. For example, when a sensor at a particular location/site is not functional, we will lose observations from that

* Corresponding author.
E-mail addresses: chenxy346@mail2.sysu.edu.cn (X. Chen), hezhch@mail.sysu.edu.cn (Z. He), lijun.sun@mcgill.ca (L. Sun).

https://doi.org/10.1016/j.trc.2018.11.003
Received 4 June 2018; Received in revised form 22 October 2018; Accepted 10 November 2018
Available online 28 November 2018

sensor continuously until it is replaced or repaired. The missing data problem becomes even more common in floating sensing and crowdsourcing systems. For example, when using floating vehicles to collect link speed data, we cannot perform estimation if no floating vehicles appear on that link in a given time window; however, this information is an essential input for route planning and travel time prediction tasks. In order to take full use of the incomplete spatiotemporal data, a critical research question is to provide robust estimates of those missing (or not observed) entries in a spatiotemporal traffic data set. There exists a large body of literature about missing data imputation in transportation systems. Li et al. (2014) and Ni et al. (2005) provide a comprehensive review of imputation methods for traffic data. The authors summarize imputation algorithms into three categories: prediction, interpolation, and statistical learning-based methods.

In modeling spatiotemporal traffic data, it is critical to take the strong spatiotemporal correlation patterns into consideration. Tensor decomposition is a standard technique to capture the multi-dimensional structural dependencies. The overall idea is to model the original multi-dimensional data using a compact structure, such as the CANDECOMP/PARAFAC (CP) model and the Tucker model (Kolda and Bader, 2009). For imputation tasks, we are interested in making robust predictions of those unobserved entries rather than in estimating model parameters. This inspires researchers to take a probabilistic approach to tackle the missing data imputation problem. To this end, for instance, Salakhutdinov and Mnih (2008) developed a Bayesian matrix factorization algorithm which provides reliable estimation. Another problem for traffic data tensor is to identify the most appropriate data representation for missing data imputation tasks. For example, traffic data collected in several weeks from multiple links can be organized into different representations, such as a matrix (road segment × time series), a third-order tensor (road segment × day × time of day), and even a fourth-order tensor (road segment × week × day of week × time of day). Ran et al. (2016) studied the effect of spatial information by comparing the performance between a fourth-order day × week × time × space tensor with separately defined third-order day × week × time × tensors for each location. However, to our knowledge, so far no studies have examined the effect of different representations of the whole data set on imputation performance. These two problems are important when applying tensor decomposition techniques in real-world traffic data imputation problems.

In this paper, we address the above two challenges for missing data imputation in a spatiotemporal multi-dimensional setting. In doing so, we extend the Bayesian matrix factorization model by Salakhutdinov and Mnih (2008) to a higher-order case to learn the underlying statistical patterns in spatiotemporal traffic data—Bayesian Gaussian CANDECOMP/PARAFAC (BGCP) tensor decomposition model. This fully Bayesian model characterizes the data generation process and thus allows us to impute missing entries efficiently. We place conjugate priors on hyper-parameters to analytically derive the posterior distributions, and this allows us to develop an efficient Markov chain Monte Carlo (MCMC) algorithm to estimate the model. This Bayesian approach can achieve stable performance with varying missing rates and even under non-random correlated missing conditions, which is a difficult problem for traditional tensor-based imputation methods. Most importantly, the MCMC algorithm provides a natural way for multiple imputation (Ni et al., 2005), which allows us to compute not only the unbiased point estimates but also robust uncertainty measures of those missing values. To answer the data representation problem, we organize the spatiotemporal traffic data into three different representations: (A) matrix (road segment × time series), (B) third-order tensor (road segment × day × time interval), and (C) fourth-order tensor (road segment × week × day × time interval). We evaluate the performance of different tensor-based imputation models on these three representations and find that the BGCP model performs consistently well with a third-order tensor structure.

The main contributions of this paper are threefold: (1) we propose a Bayesian probabilistic imputation framework for robust missing data imputation in a spatiotemporal (or other multi-dimensional) setting; (2) we demonstrate that BGCP model produces accurate imputation even under temporally correlated data corruptions; (3) we show that data representation is an important factor determining the overall imputation performance, and find the third-order (road segment × day × time interval) structure most appropriate for our traffic speed data set.

The remainder of this paper is structured as follows. Section 2 reviews tensor computation methods and their applications in general missing data imputation problems and also some specific transportation data modeling problems. In Section 3, we introduce the BGCP model which characterizes the multi-dimensional data generation process and present an efficient Gibbs sampling algorithm to obtain posterior distributions of different variables. Using a large-scale traffic speed data set collected in Guangzhou (Chen et al., 2018), we demonstrate the application of the BGCP model in Section 4, and also examine how data representation and different missing scenarios affect model performance in the missing data imputation tasks. Finally, Section 5 concludes this study and suggests some future research directions.

2. Literature review

Tensor computation is a powerful technique in various statistical learning problems. Kolda and Bader (2009) gives a very comprehensive review of tensor decomposition and its applications. In computer science, tensor computation techniques are widely used for estimation tasks such image recovery and data imputation (Liu et al., 2013; Zhao et al., 2015a,b) and also for recommendation systems (Xiong et al., 2010). In terms of missing data imputation in tensors, the central methodology of previous work can be summarized into two types. The first type is to use low-rank tensor completion without a decomposition structure (Liu et al., 2013). This approach has a good property to avoid the non-convex optimization problem. The other approach is tensor decomposition, which can be considered a higher-order extension of the singular value decomposition (SVD) algorithm. In this case, a low-rank approximation model is estimated using partially observed data. Then one can compute a missing entry from the estimated low-rank model (Anandkumar et al., 2014). There exists a large body of literature about missing traffic data imputation problem. We refer interested readers to Ran et al. (2016) and the references therein for a brief review about the imputation problems and solutions and some state-of-the-art models. In this section, we only review some representative tensor-based methods and applications.

Tensor has also been a handy tool in transportation research, in particular on modeling complex traffic/transportation data set, which often has a spatiotemporal multi-dimensional property. For example, in transportation planning, if collecting origin-destination (OD) demand matrices over time, we can create a third-order (origin × destination × time) tensor that allows us to take the temporal variation in demand patterns into consideration. Spatiotemporal traffic speed/flow data—as mentioned before—is also a good example to apply the tensor structure, since observations can be simply specified by a third-order road segment × day × time of day tensor (Chen et al., 2018). Recently, factorization-based approaches—in particular tensor decomposition—have attracted more and more attention and have been tested in many empirical works (e.g., Qu et al., 2009; Asif et al., 2013b; Li et al., 2013; Tan et al., 2013; Asif et al., 2016; Goulart et al., 2017). For example, Ran et al. (2016) employed HaLRTC—a low-rank tensor completion algorithm proposed by Liu et al. (2013)—to estimate the missing traffic volume data. Asif et al. (2016) and Chen et al. (2018) used tensor decomposition models to recover missing traffic speed data collected from urban road networks. Goulart et al. (2017) developed a Tucker decomposition methods to promote parsimony in the core tensor. Apart from missing data imputation, a good body of previous studies have demonstrated the application of tensor computation in solving other transportation problems, such as traffic prediction (Tan et al., 2016), data compression (Asif et al., 2013a), and mobility pattern discovery (Sun and Axhausen, 2016).

Most previous studies on tensor completion rely on trace norm minimization to find a low-rank approximation to the original incomplete tensor (Liu et al., 2013; Ran et al., 2016). However, such optimization is often prone to overfitting because it only computes a single point estimate, in particular when the missing rate is large. As a result, in dealing with a sparse tensor, it becomes difficult for an optimization-based completion algorithm to capture the global information and make a good estimate (Salakhutdinov and Mnih, 2008; Anandkumar et al., 2014). To address this issue, Bayesian inference methods such as MCMC and variational inference have been designed and used for tensor decomposition (Xiong et al., 2010; Rai et al., 2014; Zhao et al., 2015a,b). The advantage of a Bayesian framework is twofold: (1) a Bayesian model requires fewer observations to perform inference, thus overcoming the sparsity problem or extreme missing conditions (Salakhutdinov and Mnih, 2008; Rai et al., 2014); (2) the MCMC algorithm provides a natural way for multiple imputation, providing not only point estimates but also robust uncertainty measures for those missing entries. In the next section, we extend the neat work of Salakhutdinov and Mnih (2008)—a Bayesian probabilistic matrix factorization model—to a multi-dimensional tensor setting to model a large-scale spatiotemporal traffic data set.

Apart from the tensor-based approach, a wide variety of data-driven spatiotemporal imputation methods have been covered in many real-world applications. Here we review some signature works on different machine learning models. Tang et al. (2015) developed a hybrid approach which applies the Fuzzy C-Means-based method for imputation and uses Genetic Algorithm (GA) to find optimal membership functions and centroids of the Fuzzy C-Means models. Duan et al. (2016) proposed to use stacked denoising stacked autoencoders (SDAE) to extract features from high dimension traffic data and impute missing values. Li et al. (2018) proposed a multi-view (spatial and temporal) learning algorithm that integrates long short-term memory (LSTM) units and support vector regression (SVR). Two cokriging methods were proposed in Bae et al. (2018) to model spatiotemporal dependencies and perform imputation in traffic data. In particular, the authors evaluated their models against different missing pattern scenarios (e.g., random missing and block missing). Based on spatiotemporal correlations, Laña et al. (2018) proposed two machine learning models for missing data imputation: a spatial context sensing model (using information from surrounding sensors) and an automated clustering algorithm (obtaining optimal number of patterns). Notably, a recent work from Rodrigues et al. (2018) used multi-output Gaussian processes—as a flexible Bayesian nonparametric model—to capture the complex spatial and temporal patterns in crowdsourced traffic data. Gaussian processes offer a very flexible way to capture spatiotemporal dependencies at different scales.

3. Bayesian Gaussian CP decomposition

In this section, we present the framework to fill missing/unobserved entries in a multi-dimensional traffic data tensor. We extend the Bayesian matrix factorization model of Salakhutdinov and Mnih (2008) to higher-order tensors and introduce a Bayesian Gaussian CANDECOMP/PARAFAC (BGCP) tensor decomposition model which characterizes the data generation process.

3.1. Model description

Let $\mathcal{Y} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$ denote a dth-order tensor, where n_k is the dimension along the kth way ($k \in \{1, 2, ..., d\}$). In this tensor $\mathcal{Y}$, we use $\boldsymbol{i} = (i_1, i_2, ..., i_d)$ ($1 \leqslant i_k \leqslant n_k, \ \forall\, k \in \{1, 2, ..., d\}$) to denote the index of an entry and we denote by $y_{\boldsymbol{i}}$ the value of this entry. The idea of CP decomposition is to approximate $\mathcal{Y}$ using a low-rank structure as follows

$$\widehat{\mathcal{Y}} = \sum_{j=1}^{r} \boldsymbol{u}_j^{(1)} \circ \boldsymbol{u}_j^{(2)} \circ \cdots \circ \boldsymbol{u}_j^{(d)}, \tag{1}$$

where $\boldsymbol{u}_j^{(k)} \in \mathbb{R}^{n_k}$ is the jth column vector of the kth decomposed factor matrix $U^{(k)} \in \mathbb{R}^{n_k \times r}$. The symbol $\circ$ means vector outer product, and thus $\boldsymbol{u}_j^{(1)} \circ \boldsymbol{u}_j^{(2)} \circ \cdots \circ \boldsymbol{u}_j^{(d)}$ is a rank-one tensor. With this formulation, the CP decomposition of a tensor can be considered a sum of r rank-one component tensors (we call r the CP rank of tensor $\widehat{\mathcal{Y}}$). Element-wise, Eq. (1) is equivalent to

$$\widehat{y}_{\boldsymbol{i}} = \sum_{j=1}^{r} u_{i_1 j}^{(1)} u_{i_2 j}^{(2)} \cdots u_{i_d j}^{(d)}, \tag{2}$$

where $u_{i_k j}^{(k)}$ is the value at (i_k, j) (row i_k, column j) in the kth factor matrix $U^{(k)}$.

The formulation above provides the general idea of CP decomposition. It should be noted that the tensor $\mathscr{Y}$ is incomplete with missing entries. We denote by Ω the index set of those observed entries.

We next introduce a fully Bayesian model for the data generation process. First, we assume that the noise term for each observed entry ($\boldsymbol{i} \in \Omega$) in the approximation follows independent Gaussian distribution

$$x_{\boldsymbol{i}} \sim \mathcal{N}(\hat{x}_{\boldsymbol{i}}, \tau_\epsilon^{-1}), \tag{3}$$

where $\mathcal{N}(\cdot)$ denotes a Gaussian distribution and τ_ϵ is the precision, which is a universal parameter for all elements.

To model the tensor data properly, we place flexible prior distributions over both the group of factor matrices $U^{(k)}$ and the precision τ_ϵ. Specifically, the prior distributions over the row vectors in all factor matrices are assumed to be multivariate Gaussians:

$$\boldsymbol{u}_{ik}^{(k)} \sim \mathcal{N}(\boldsymbol{\mu}^{(k)}, (\Lambda^{(k)})^{-1}). \tag{4}$$

In a Bayesian setting, we place conjugate Gaussian-Wishart priors for hyper-parameters $\boldsymbol{\mu}^{(k)} \in \mathbb{R}^r$ and $\Lambda^{(k)} \in \mathbb{R}^{r\times r}$. This will enhance the robustness of the model and also speed up the convergence when performing model inference using sampling algorithms. The hyper-priors on $\boldsymbol{\mu}^{(k)}$ and $\Lambda^{(k)}$ ($k = 1, ..., d$) are defined as below:

$$\begin{aligned}&(\boldsymbol{\mu}^{(k)}, \Lambda^{(k)}) \sim \text{Gaussian-Wishart}(\boldsymbol{\mu}_0, \beta_0, W_0, \nu_0),\\ &p(\boldsymbol{\mu}^{(k)}, \Lambda^{(k)}|-) = \mathcal{N}(\boldsymbol{\mu}^{(k)}|\boldsymbol{\mu}_0, (\beta_0\Lambda^{(k)})^{-1}) \times \text{Wishart}(\Lambda^{(k)}|W_0, \nu_0).\end{aligned} \tag{5}$$

A Wishart distribution with ν_0 degrees of freedom and a $r \times r$ scale matrix W_0 is given by:

$$\text{Wishart}(\Lambda^{(k)}|W_0, \nu_0) = \frac{1}{C}|\Lambda^{(k)}|^{\frac{\nu_0-r-1}{2}}\exp\left\{-\frac{1}{2}\text{tr}(W_0^{-1}\Lambda^{(k)})\right\}, \tag{6}$$

where $\text{tr}(\cdot)$ (the trace function) of a square matrix is the sum of all elements on its main diagonal.

Under the Gaussian assumption in Eq. (3), the precision parameter τ_ϵ captures the degree of noise in the data. Note that in the traffic speed data the precision τ_ϵ is also unknown to us and it cannot be fully captured by the inverse of variance of all observations. Therefore, different from Salakhutdinov and Mnih (2008) which uses a fixed precision value, here we place a flexible conjugate Gamma prior over τ_ϵ to improve robustness of the model:

$$\tau_\epsilon \sim \text{Gamma}(a_0, b_0), \tag{7}$$

where a_0 and b_0 are the shape parameter and the rate parameter, respectively. If a random variable $x \sim \text{Gamma}(a, b)$, we have $p(x) = \frac{b^a}{\Gamma(a)}x^{a-1}\exp(-bx)$.

Fig. 1 shows the overall graphical structure characterizing the data generation process of the Bayesian probabilistic CP decomposition introduced above. Two graphical models are shown here as instances for third-order tensors ($d = 3$, the left panel) and fourth-order tensors ($d = 4$, the right panel), respectively. In this graphical model, the shaded node $x_{\boldsymbol{i}}$ ($\boldsymbol{i} \in \Omega$) represents the observed data; $U^{(k)}$ and τ_ϵ are parameters; $\boldsymbol{\mu}^{(k)}$, $\Lambda^{(k)}$, a_0, and b_0 are hyper-parameters. We place hyper-priors Gaussian-Wishart($\boldsymbol{\mu}_0, \beta_0, W_0, \nu_0$) on ($\boldsymbol{\mu}^{(k)}, \Lambda^{(k)}$). These hyper-parameters should be set when initializing the sampling algorithm. We can derive Gibbs sampling algorithm for model inference based on this graphical model.

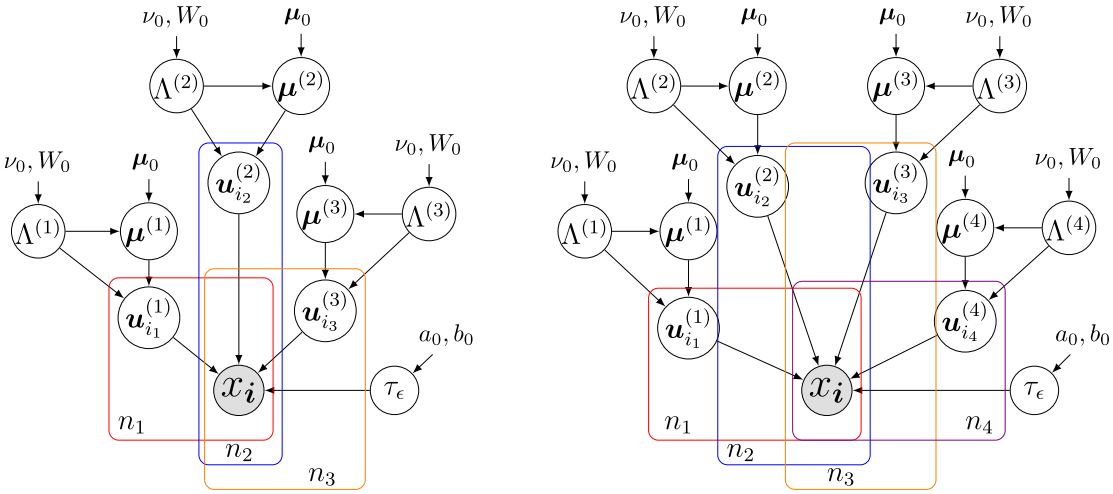


Fig. 1. The probabilistic graphical models of CP decomposition for tensors.

3.2. Gibbs sampling

Based on the work of Salakhutdinov and Mnih (2008), we next introduce a Gibbs sampling algorithm to perform inference on the presented graphical model. Gibbs sampling is a widely used MCMC algorithm for Bayesian inference. The idea of Gibbs sampling is to update all variables sequentially in each iteration. Each variable is sampled from its distribution conditional on the current values of all other variables. The key of a Gibbs sampling algorithm is to define such distributions for all variables. These conditional distributions are often referred to as full conditionals.

Since we have used conjugate priors for parameters and hyper-parameters, the posterior distributions in the graphical model presented in Fig. 1 can be derived analytically. Using a third-order ($d = 3$) tensor as an example, we next derive the posterior distributions for each parameter in closed-form based on its Markov blanket. The same analyses can be performed on a higher-order tensor structure.

3.2.1. Sampling factor matrices $U^{(k)}$ for $k = 1, 2, 3$

Essentially, the goal of sampling factor matrices is to capture the dependencies between observations $x_{\boldsymbol{i}}$ ($\boldsymbol{i} \in \Omega$) and hyper-parameters $\boldsymbol{\mu}^{(k)}$ and $\Lambda^{(k)}$ ($k = 1, 2, 3$). Given a partially observed tensor $\mathscr{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we first define an indicator tensor $\mathscr{B}$ which is of the same size as $\mathscr{X}$, with each element $b_{\boldsymbol{i}}$ being 1 if $\boldsymbol{i} \in \Omega$ and 0 otherwise.

The factor matrix $U^{(1)}$ is updated by sampling all $\boldsymbol{u}_{i_1}^{(1)} \in \mathbb{R}^r$ ($1 \leqslant i_1 \leqslant n_1$) one by one. Considering the Gaussian assumption in Eq. (3), the likelihood can be written as

$$\begin{aligned}\mathscr{L}(\mathscr{X}|\boldsymbol{u}_{i_1}^{(1)}, U^{(2)}, U^{(3)}, \tau_\epsilon) &\propto \prod_{i_2=1}^{n_2}\prod_{i_3=1}^{n_3} \exp\left\{-\tfrac{\tau_\epsilon b_{\boldsymbol{i}}}{2}(x_{\boldsymbol{i}} - \hat{x}_{\boldsymbol{i}})^2\right\} \\ &= \exp\left\{-\tfrac{\tau_\epsilon}{2}\sum_{i_2=1}^{n_2}\sum_{i_3=1}^{n_3} b_{\boldsymbol{i}}(x_{\boldsymbol{i}} - (\boldsymbol{u}_{i_1}^{(1)})^T(\boldsymbol{u}_{i_2}^{(2)} \circledast \boldsymbol{u}_{i_3}^{(3)}))^2\right\},\end{aligned} \tag{8}$$

where the symbol $\circledast$ represents Hadamard product. Note that only the slice $\mathscr{X}(i_1, :, :) \in \mathbb{R}^{n_2 \times n_3}$ is used to compute the likelihood term in Eq. (8).

Combining Eqs. (8) and (4) gives us the posterior distribution, which can be also written in the form of a multivariate Gaussian

$$\begin{aligned}p(\boldsymbol{u}_{i_1}^{(1)}|\mathscr{X}, U^{(2)}, U^{(3)}, \tau_\epsilon, \boldsymbol{\mu}^{(1)}, \Lambda^{(1)}) &\propto \mathscr{L}(\mathscr{X}|\boldsymbol{u}_{i_1}^{(1)}, U^{(2)}, U^{(3)}, \tau_\epsilon) \times \mathscr{N}(\boldsymbol{u}_{i_1}^{(1)}|\boldsymbol{\mu}^{(1)}, (\Lambda^{(1)})^{-1}) \\ &\propto \mathscr{N}(\boldsymbol{u}_{i_1}^{(1)} \Big| \hat{\boldsymbol{\mu}}_{i_1}^{(1)}, \left(\hat{\Lambda}_{i_1}^{(1)}\right)^{-1})\end{aligned} \tag{9}$$

where

$$\begin{aligned}\hat{\Lambda}_{i_1}^{(1)} &= \tau_\epsilon \textstyle\sum_{i_2=1}^{n_2}\sum_{i_3=1}^{n_3} b_{\boldsymbol{i}}(\boldsymbol{u}_{i_2}^{(2)} \circledast \boldsymbol{u}_{i_3}^{(3)})(\boldsymbol{u}_{i_2}^{(2)} \circledast \boldsymbol{u}_{i_3}^{(3)})^T + \Lambda^{(1)}, \\ \hat{\boldsymbol{\mu}}_{i_1}^{(1)} &= (\hat{\Lambda}_{i_1}^{(1)})^{-1}\Big(\tau_\epsilon \textstyle\sum_{i_2=1}^{n_2}\sum_{i_3=1}^{n_3} x_{\boldsymbol{i}}(\boldsymbol{u}_{i_2}^{(2)} \circledast \boldsymbol{u}_{i_3}^{(3)}) + \Lambda^{(1)}\boldsymbol{\mu}^{(1)}\Big).\end{aligned} \tag{10}$$

It should be noted that in Eq. (10) only the two terms $x_{\boldsymbol{i}}$ and $b_{\boldsymbol{i}}$ contain index i_1. This allows us to sample all vectors $\boldsymbol{u}_{i_1}^{(1)}$ ($1 \leqslant i_1 \leqslant n_1$) in $U^{(k)}$ in parallel. Following the same procedure deriving $p(\boldsymbol{u}_{i_1}^{(1)}|-)$, we can also write the posteriors of $\boldsymbol{u}_{i_2}^{(2)}$ and $\boldsymbol{u}_{i_3}^{(3)}$ with a similar derivation.

3.2.2. Sampling $\boldsymbol{\mu}^{(k)}$ and $\Lambda^{(k)}$ for $k = 1, 2, 3$

For simplicity, here we only show the derivation for $k = 1$ as an example and other cases are equivalent. When sampling $\boldsymbol{\mu}^{(1)}$ and $\Lambda^{(1)}$ (see Fig. 1(a)), the likelihood of the factor matrix $U^{(1)} \in \mathbb{R}^{n_1 \times r}$ can be factorized into the product of conditional distributions of n_1 individual vectors:

$$\mathscr{L}(U^{(1)}|\boldsymbol{\mu}^{(1)}, \Lambda^{(1)}) \propto |\Lambda^{(1)}|^{\frac{n_1}{2}} \prod_{i_1=1}^{n_1} \exp\left\{-\frac{1}{2}(\boldsymbol{u}_{i_1}^{(1)} - \boldsymbol{\mu}^{(1)})^T\Lambda^{(1)}(\boldsymbol{u}_{i_1}^{(1)} - \boldsymbol{\mu}^{(1)})\right\}. \tag{11}$$

Given the likelihood term in Eq. (11) and the Gaussian-Wishart hyper-prior in Eq. (5), we can write down and factorize the joint posterior distribution for hyper-parameters $\boldsymbol{\mu}^{(1)}$ and $\Lambda^{(1)}$ as follows:

$$\begin{aligned}p(\boldsymbol{\mu}^{(1)}, \Lambda^{(1)}|U^{(1)}, \boldsymbol{\mu}_0, W_0, \nu_0, \beta_0) &\propto \mathscr{L}(U^{(1)}|\boldsymbol{\mu}^{(1)}, \Lambda^{(1)}) \times \mathscr{N}(\boldsymbol{\mu}^{(1)}|\boldsymbol{\mu}_0, (\beta_0\Lambda^{(1)})^{-1}) \times \text{Wishart}(\Lambda^{(1)}|W_0, \nu_0) \\ &\propto \mathscr{N}(\boldsymbol{\mu}^{(1)} \Big| \hat{\boldsymbol{\mu}}^{(1)}, (\hat{\Lambda}^{(1)})^{-1}) \times \text{Wishart}(\Lambda^{(1)}|\widehat{W}_0^{(1)}, \hat{\nu}_0^{(1)}),\end{aligned} \tag{12}$$

where the parameters in these two distributions can be computed by:

$$\begin{aligned}\widehat{W}_0^{(1)} &= \left(n_1 S^{(1)} + \frac{n_1\beta_0}{n_1+\beta_0}\left(\bar{\boldsymbol{u}}^{(1)} - \boldsymbol{\mu}_0\right)\left(\bar{\boldsymbol{u}}^{(1)} - \boldsymbol{\mu}_0\right)^T + W_0^{-1}\right)^{-1}\\ \widehat{\nu}_0^{(1)} &= n_1 + \nu_0,\\ \widehat{\boldsymbol{\mu}}^{(1)} &= \frac{1}{n_1+\beta_0} n_1\left(\bar{\boldsymbol{u}}^{(1)} + \beta_0\boldsymbol{\mu}_0\right),\\ \widehat{\Lambda}^{(1)} &= (n_1+\beta_0)\Lambda^{(1)},\end{aligned} \tag{13}$$

where $\bar{\boldsymbol{u}}^{(1)}$ and $S^{(1)}$ are two statistics defined as below:

$$\bar{\boldsymbol{u}}^{(1)} = \sum_{i_1=1}^{n_1} \boldsymbol{u}_{i_1}^{(1)},\ S^{(1)} = \frac{1}{n_1}\sum_{i_1=1}^{n_1}\left(\boldsymbol{u}_{i_1}^{(1)} - \bar{\boldsymbol{u}}^{(1)}\right)\left(\boldsymbol{u}_{i_1}^{(1)} - \bar{\boldsymbol{u}}^{(1)}\right)^T.$$

Above we have shown the derivation of the full conditionals for $\boldsymbol{\mu}^{(1)}$ and $\Lambda^{(1)}$ used in Gibbs sampling. The same procedure can be applied for $k = 2$ and $k = 3$.

3.2.3. Sampling precision τ_ϵ

The likelihood of all observations is given by

$$\mathscr{L}(\mathscr{X}|U^{(1)}, U^{(2)}, U^{(3)}, \tau_\epsilon) \propto \prod_{\boldsymbol{i}\in\Omega} (\tau_\epsilon)^{1/2}\exp\left\{-\frac{\tau_\epsilon}{2}(x_{\boldsymbol{i}} - \hat{x}_{\boldsymbol{i}})^2\right\}. \tag{14}$$

Combining the likelihood term in Eq. (14) and the prior term in Eq. (7) will give the posterior of τ_ϵ, which is also a Gamma distribution parameterized by $\hat{a}_0$ and $\hat{b}_0$:

$$\begin{aligned}p(\tau_\epsilon|\mathscr{X}, U^{(1)}, U^{(2)}, U^{(3)}, a_0, b_0) &\propto \mathscr{L}(\mathscr{X}|U^{(1)}, U^{(2)}, U^{(3)}, \tau_\epsilon) \times \text{Gamma}(\tau_\epsilon|a_0, b_0)\\ &\propto \text{Gamma}(\tau_\epsilon|\hat{a}_0, \hat{b}_0),\end{aligned} \tag{15}$$

where $\hat{a}_0 = \frac{1}{2}\sum_{\boldsymbol{i}\in\Omega} b_{\boldsymbol{i}} + a_0$ and $\hat{b}_0 = \frac{1}{2}\sum_{\boldsymbol{i}\in\Omega}(x_{\boldsymbol{i}} - \hat{x}_{\boldsymbol{i}})^2 + b_0$.

3.3. Implementation

Above we have derived the full conditionals for all variables in the Gibbs sampling for a third-order tensor as illustrated in Fig. 1(a). Following the same steps, one can also derive the Gibbs updating rules for a fourth-order tensor (Fig. 1(b)). All missing values can be estimated from Monte Carlo approximation after the Gibbs sampling algorithm reaches stationary. We summarize the Gibbs sampling algorithm of the BGCP model as below:

Input: a d-th order tensor $\mathscr{X}$ and the corresponding indicator tensor $\mathscr{B}$.
Initialization: For hyper-priors, we set $\boldsymbol{\mu}_0 = \mathbf{0}$, $\beta_0 = 1$, $W_0 = I$ (identity matrix), $\nu_0 = r$ (the low rank). For hyper-parameters, we initialize $U^{(k)}$ with random values ($\forall\ k = 1, ..., d$), $a_0 = 1$, and $b_0 = 1$.
Repeat
  For $k = 1, ..., d$
    Compute $\widehat{W}_0^{(k)}$ and $\widehat{\nu}_0^{(k)}$ (Eq. (13)).
    Sample $\Lambda^{(k)} \sim \text{Wishart}\left(\widehat{W}_0^{(k)}, \widehat{\nu}_0^{(k)}\right)$.
    Compute $\widehat{\boldsymbol{\mu}}^{(k)}$ and $\widehat{\Lambda}^{(k)}$ (Eq. (13)).
    Sample $\boldsymbol{\mu}^{(k)} \sim \mathcal{N}\left(\widehat{\boldsymbol{\mu}}^{(k)}, \left(\widehat{\Lambda}^{(k)}\right)^{-1}\right)$.
    For $i_k = 1, ..., n_k$ (can be done in parallel)
      Compute $\widehat{\boldsymbol{\mu}}_{i_k}^{(k)}$ and $\widehat{\Lambda}_{i_k}^{(k)}$ (Eq. (10))
      Sample $\boldsymbol{u}_{i_k}^{(k)} \sim \mathcal{N}\left(\widehat{\boldsymbol{\mu}}_{i_k}^{(k)}, \left(\widehat{\Lambda}_{i_k}^{(k)}\right)^{-1}\right)$.
    End for
  End for
  Compute $\hat{a}_0$ and $\hat{b}_0$ (see Eq. (15)).
  Sample $\tau_\epsilon \sim \text{Gamma}(\hat{a}_0, \hat{b}_0)$.
Until maximum number of iterations.

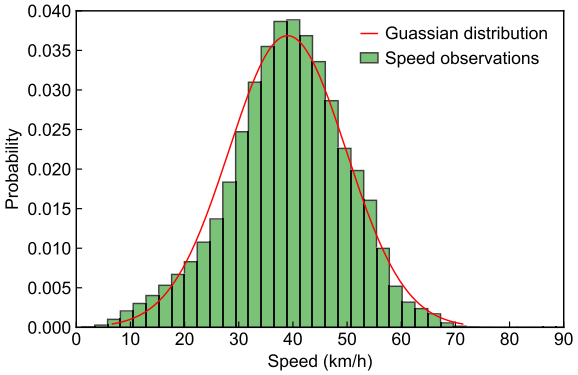


Fig. 2. Histogram of traffic speed observations associated with a Gaussian distribution $x_{\boldsymbol{i}} \sim \mathcal{N}(\mu, \sigma^2)$, $\boldsymbol{i} \in \Omega$ where $\mu = 39.01$ and $\sigma = 10.82$.

4. Numerical experiment

4.1. Traffic speed data

To demonstrate the performance of this model, in this section we conduct numerical experiments based on a large-scale traffic speed data set collected in Guangzhou, China. The data set is generated by a widely-used navigation app on smart phones. We make the data set publicly available at https://doi.org/10.5281/zenodo.1205229. The data set contains travel speed observations from 214 road segments in two months (61 days from August 1, 2016 to September 30, 2016) at 10-min interval (144 time intervals in a day). The speed data can be organized as a third-order tensor (road segment × day × time interval, with a size of $214 \times 61 \times 144$). Among the 1.88 million elements, about 1.29% are not observed or provided in the raw data. Fig. 2 shows the histogram of the observed speed data, together with a Gaussian distribution fitted with maximum likelihood. As we can see, the speed data is well captured by a Gaussian distribution without any extreme values, supporting the Gaussian assumption of the error term (a critical assumption). Apart from our data set, the Gaussian distribution assumption for observations appears in many studies (e.g., Salakhutdinov and Mnih (2008), Zhao et al. (2015a,b)).

Before estimating the missing entries, we first investigate the time-varying pattern of the traffic speed data set and show it in Fig. 3. Fig. 3(a) shows the time window × date speed profile by averaging speed over 214 road segments. Fig. 3(b) shows the traffic speed trends of the weekday (Monday through Friday) and weekend (Saturday and Sunday) by averaging observations over links and days. Both panels show the differences between peak hours and off-peak hours clearly, and also between weekdays and weekends. Traffic speed in the evening peak hours is much slower than in the early morning or in the night. Moreover, daily traffic speed trends on the weekday (or weekend) are quite similar (see Fig. 3(a)), except that the traffic speed on the weekends is larger than that on the weekdays, in particular during peak hours. Fig. 3(c) shows the detailed time-varying speed data on two road segments (#1 and #2). As we can see, the traffic speed data shows both recurrent and non-recurrent patterns (e.g., Sept 15–17, Mid-Autumn Festival holiday).

Since only 1.29% entries are not observed in the raw data, we create synthetic data by randomly removing a certain amount of entries, and thus divide the raw data into two groups: the observed (Ω) and the missing (removed). For these "missing" entries, we also have the corresponding ground truth, which allows us to assess the imputation performance directly. The model performance is evaluated by imputing those removed entries.

4.2. Missing data imputation

The imputation in the BGCP model is done by averaging multiple Gibbs runs after stationary:

$$p(x_{\boldsymbol{i}}|\mathcal{Y}) \approx \frac{1}{T}\sum_{t=1}^{T} \mathcal{N}\left(\sum_{j=1}^{r} u_{i_1 jt}^{(1)} u_{i_2 jt}^{(2)} \cdots u_{i_d jt}^{(d)}, \tau_t^{-1}\right), \tag{16}$$

where $\mathcal{Y}$ is the observed data, $x_{\boldsymbol{i}}$ is a missing entry, $u_{i_d jt}^{(d)}$ is (i_d, j)-th entry in the factor matrix of the t-th Gibbs iteration (same for τ_t), and T is number of samples used.

In the numerical experiment, we compare the BGCP model with two other tensor decomposition-based imputation methods:

- high accuracy low-rank tensor completion (HaLRTC) (Liu et al., 2013), which is used in Ran et al. (2016), and
- SVD-combined tensor decomposition (STD) (Chen et al., 2018).

Note that we need to set tensor rank for the BGCP model, while HaLRTC and STD do not require additional parameters. The mean absolute percentage error (MAPE) and root mean square error (RMSE) are used to evaluate model performance:

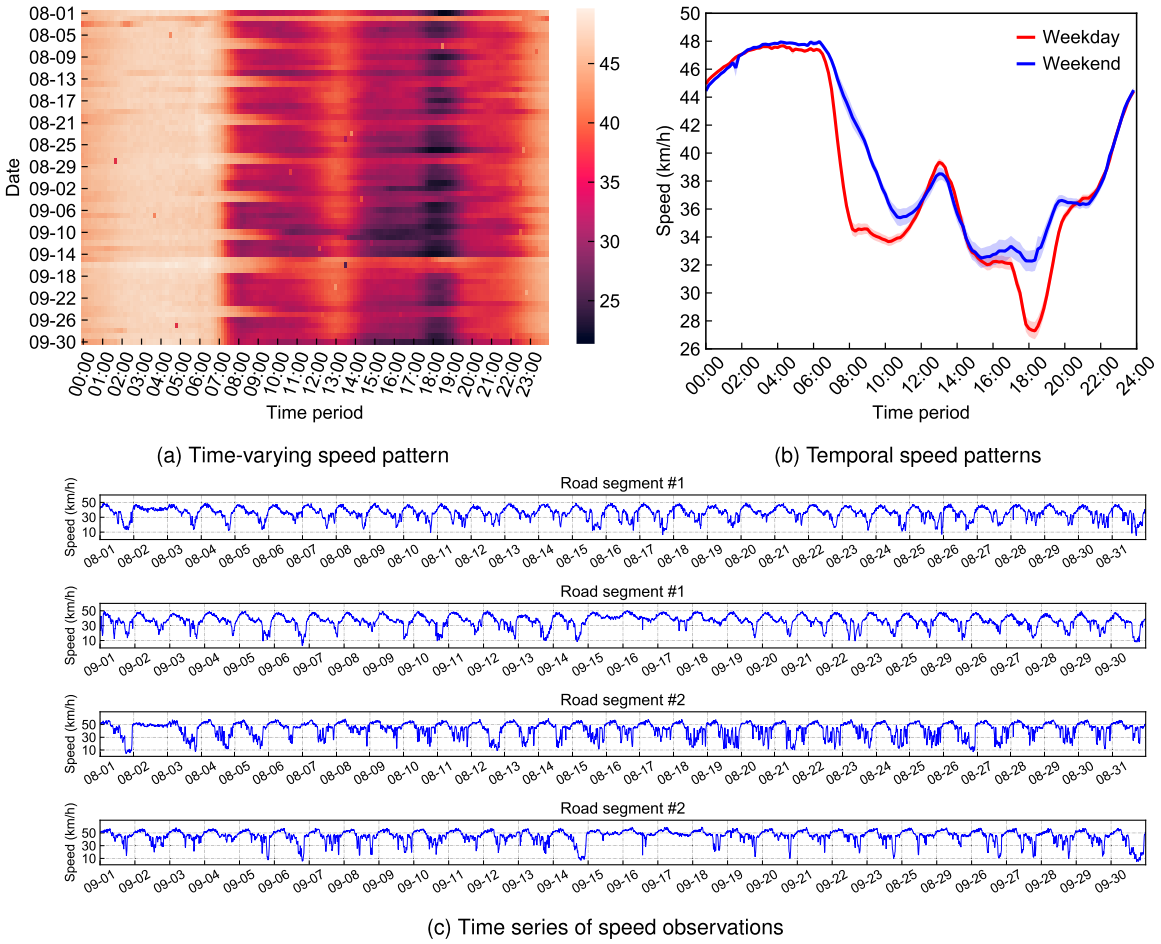


Fig. 3. Multi-dimensional traffic speed sequences. (a) The heat-map shows averaged speed values over 214 road segments for each specific day and time window. (b) The curves show averaged speed values over 214 road segments. (b) Examples of time series of traffic speed observations.

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|x_i - \hat{x}_i|}{x_i},$$
$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2}. \tag{17}$$

where n is the total number of missing entries, and x_i and $\hat{x}_i$ are the actual value of a missing entry and its imputation, respectively.

In the experiment, we would like to assess the performance of different models and different data representations under different missing rate and also different missing scenarios (random missing vs non-random missing). In other words, we consider data representation a decision variable, and in practice one need to choose the most appropriate representation before choosing a model. In doing so, we create a synthetic data set for each combination of missing rate and missing scenario.

4.3. Data representations

The structural representation of the tensor is also a crucial factor for us to consider. Based on the traffic speed data set, we propose three different types of data representations:

(**A**) Matrix (second-order) representation of road segment × time interval, i.e., $\mathscr{X} \in \mathbb{R}^{214\times 8784}$. Here we organize the day × time interval as a single time series.
(**B**) Third-order tensor representation of road segment × day × time interval, i.e., $\mathscr{X} \in \mathbb{R}^{214\times 61\times 144}$.
(**C**) Fourth-order tensor representation of road segment × week × day of week × time interval, i.e., $\mathscr{X} \in \mathbb{R}^{214\times 9\times 7\times 144}$. Here the 61 days are organized as nine weeks (with two days missing).

4.4. Random missing v.s. non-random missing

For the random missing scenario, we just randomly remove some entries in representation (A) and reshape the matrix to representations (B) and (C). However, as mentioned, in field applications the missing data problem mainly results from various factors such as hardware/software failure, network communication problems. Therefore, in practice, missing data often show correlated corruptions over a period. To take this effect into account, we design a non-random missing scenario—fiber missing—following Chen et al. (2018).

We first re-organize the traffic speed data into a third-order tensor (road segment × day × time interval). To create synthetic data for the fiber missing scenario, we randomly select some combinations of road segment × day and remove the corresponding time interval vectors from the tensor. This gives us a fiber missing scenario following representation (B), and we can easily reshape this tensor to representations (A) and (C). Essentially, in the fiber missing scenario, we lose more information than the random missing case (even with the same number of missing entries) and it gets more difficult for a tensor-based algorithm to perform imputation tasks.

4.5. Results

The Matlab code for numerical experiments is available at https://github.com/lijunsun/bgcp_imputation. Our first experiment examines the performance of different models and different representations in the random missing scenario. Note that here we also consider choosing the right data representation as a decision to make in field applications and we are interested in not only identifying the best model but also the best data representation. Since STD mainly address the non-convexity problem, it does not offer too much improvement in the matrix representation (A), and for the fourth-order tensor (C). Therefore, we only apply the STD model to the third-order tensor representation (B) in the comparison.

We create five synthetic data sets with missing rates ranging from 10% to 50%. For the BGCP model, we run the MCMC sampling algorithm for 1500 iterations. We find that the values of those parameters and hyper-parameters typically stabilize after a few hundred iterations. Although the BGCP model is a latent class model, we have not observed evident label switching phenomenon in our case. We take the first 1000 as burn-in and estimate those missing entries based on the last 500 iterations ($T = 500$). Note that we do not conduct thinning since the correlation does not matter for making predictions when choosing a large T.

Table 1 shows the imputation performance of BGCP, HaLRTC, STD and other models under different missing rates and different data presentations. DA (daily average) fills the missing value with an average of observed data (over different days) for the same road segment and the same time window (Li et al., 2013). kNN is another baseline method where the neighbors refers to road segments. We compute pairwise Euclidean distance (a full vector has 61×144 elements and missing entries are not considered) and quantify the performance of kNN by varying the size of neighbors ($k = 2, ..., 20$). The minimum RMSE is achieved when $k = 10$. From Table 1, daily average and kNN are combated by the state-of-the-art tensor completion models. Since the traffic speed still encodes sufficient redundant information in the random missing scenario, we run the BGCP model with a large rank ($r = 50$, $r = 80$, and $r = 110$) for better performance. Due to the high computational cost for the fourth-order tensor, we only apply $r = 50$ for representation (C).

Overall, we do find that data representation profoundly affects overall imputation performance. The third-order tensor representation (B) is demonstrated to be the best algebraic structure to model the traffic speed data. The matrix representation (A) is less effective than both third-order (B) and fourth-order (C) tensors. This might result from the fact that the model requires a large number of factors and parameters ($r \times 8784$) to capture the variation over the whole period. The HaLRTC model outperforms the BGCP model at a low missing rate (10–20%); however, HaLRTC is highly sensitive to different missing rates. The BGCP model, on the

Table 1
The imputation performance of BGCP, HaLRTC, STD, DA (daily average) and kNN for three data representations in the random missing scenario (best models are highlighted in bold).

		10%		20%		30%		40%		50%	
		MAPE	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE	RMSE
(A)	BGCP(50)	0.0937	3.9981	0.0952	4.0467	0.0962	4.0903	0.0976	4.1457	0.0997	4.2242
	BGCP(80)	0.0925	3.9483	0.0941	3.9958	0.0951	4.0449	0.0968	4.1091	0.0990	4.1919
	BGCP(110)	0.0920	3.9303	0.0937	3.9790	0.0948	4.0292	0.0965	4.0937	0.0986	4.1731
	HaLRTC	0.0957	3.9666	0.0976	4.0232	0.0991	4.0820	0.1009	4.1467	0.1035	4.2337
	DA	0.1213	5.1778	0.1218	5.1905	0.1217	5.1977	0.1217	5.1993	0.1221	5.2113
	kNN(10)	0.1303	5.1101	0.1314	5.1486	0.1322	5.1966	0.1333	5.2565	0.1356	5.3573
(B)	BGCP(50)	0.0862	3.7097	0.0867	3.7199	0.0867	3.7298	0.0867	3.7317	0.0871	3.7466
	BGCP(80)	0.0823	3.5614	0.0827	3.5660	0.0827	3.5775	0.0829	3.5851	0.0833	3.6009
	BGCP(110)	0.0795	3.4521	**0.0798**	3.4531	**0.0799**	**3.4655**	**0.0801**	**3.4756**	**0.0807**	**3.5042**
	HaLRTC	0.0777	3.1917	0.0815	3.3324	0.0850	3.4748	0.0887	3.6143	0.0931	3.7730
	STD	0.0888	3.7708	0.0911	3.8308	0.0936	3.9286	0.0963	4.0265	0.0993	4.1253
(C)	BGCP(50)	0.0896	3.8517	0.0900	3.8540	0.0898	3.8548	0.0900	3.8608	0.0903	3.8740
	HaLRTC	**0.0776**	**3.1716**	0.0817	**3.3231**	0.0857	3.4802	0.0900	3.6397	0.0951	3.8259

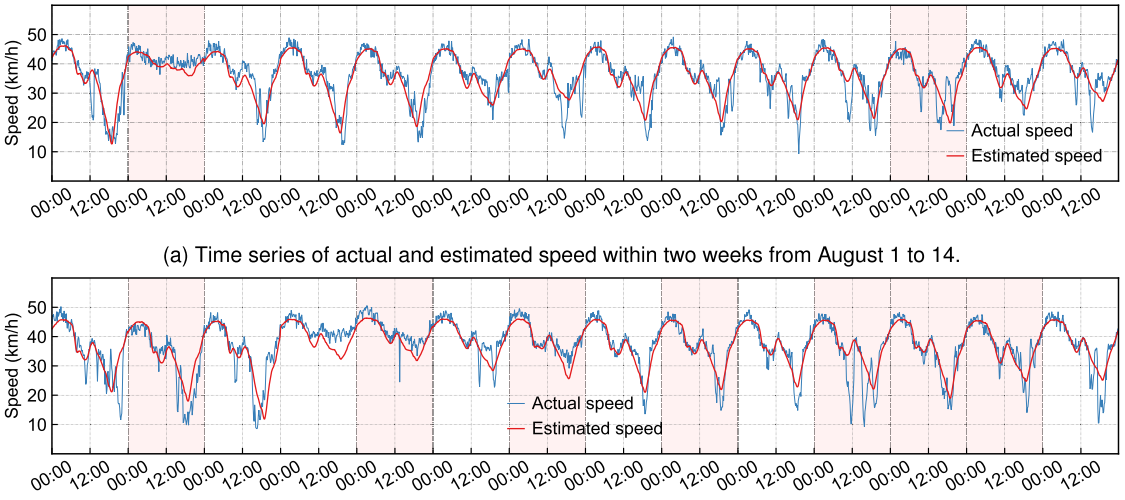
Table 2
The imputation performance of BGCP, HaLRTC, STD, DA and kNN for three data representations in the fiber missing scenario (best models are highlighted in bold).

		10%		20%		30%		40%		50%	
		MAPE	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE	RMSE
(A)	BGCP(15)	0.1011	4.2458	0.1013	4.2674	0.1020	4.3162	0.1031	4.3915	0.1055	4.4952
	BGCP(20)	0.1005	4.2307	0.1010	4.2755	0.1017	4.3229	0.1031	4.4124	0.1055	4.5070
	HaLRTC	0.1015	**4.1322**	0.1022	4.1716	0.1035	4.2372	0.1057	4.3232	0.1090	4.4472
	DA	0.1208	5.1128	0.1207	5.1353	0.1200	5.1408	0.1196	5.1434	0.1211	5.2018
	kNN(13)	0.1342	5.1714	0.1340	5.1983	0.1346	5.2591	0.1388	5.4405	0.1445	5.6516
(B)	BGCP(15)	0.0992	4.1760	0.0995	4.1949	0.0999	4.2425	**0.1001**	**4.2881**	**0.1030**	**4.4199**
	BGCP(20)	**0.0980**	4.1413	**0.0984**	**4.1477**	**0.0980**	**4.1857**	0.1006	4.4556	0.1036	4.6496
	HaLRTC	0.1033	4.1576	0.1046	4.2086	0.1062	4.2792	0.1088	4.3813	0.1131	4.5271
	STD	0.1019	4.1881	0.1054	4.3300	0.1068	4.4029	0.1115	4.5573	0.1133	4.6291
(C)	BGCP(15)	0.1039	4.3397	0.1030	4.3199	0.1026	4.3426	0.1040	4.4158	0.1051	4.4586
	BGCP(20)	0.1019	4.2691	0.1018	4.2861	0.1014	4.3158	0.1025	4.4291	0.1062	4.6320
	HaLRTC	0.1089	4.3367	0.1109	4.4173	0.1136	4.5303	0.1178	4.7054	0.1243	4.9463

other hand, shows stable performance over varying missing rates for all these three representations.

In the second experiment, we present a more realistic temporally correlated missing scenario. Here we adopt the fiber missing experiments presented in Chen et al. (2018). The DA and kNN algorithms are implemented in the same way as in our first experiment. For kNN, the minimum RMSE is achieved when $k = 13$. Due to the temporally correlated corruption in the fiber missing scenario, it becomes difficult to borrow information from other dimensions. Hence, we choose a smaller CP rank ($r = 15$ and $r = 20$) to characterize the main factors better and avoid overfitting. As the results in Table 2 show, the BGCP model and the third-order tensor representation (B) clearly outperforms other models and structures in this fiber missing scenario. In terms of the CP rank, we find that $r = 20$ works better than $r = 15$ when missing rate is less than 30%. As mentioned, the fiber missing case is more complicated to deal with since information is lost in a correlated manner. Therefore, we suspect that a larger rank may result in overfitting when α is large. Fig. 4 takes road segment #1 as an example and shows the imputation performance of BGCP with $r = 15$ and $\alpha = 30\%$. The missing fibers are shown as red windows in this figure. The BGCP model works well in reproducing the recurrent and non-recurrent traffic speed data by borrowing information from other days and other road segments.

What is the most appropriate data representation for the missing data imputation task? In Ran et al. (2016), the authors suggest that defining an additional "week" dimension (representation (C)) may help achieve better performance than applying decomposition on structures aggregating days of several weeks as one dimension, while our experiment prefers the use of aggregated third-order tensor along the temporal dimension (B). As can be seen from Table 1, the third-order representation (B) generally shows the best performance, although fourth-order (C) performs slightly better under the HaLRTC model with missing rate $\alpha = 10\%$. Moreover,



(a) Time series of actual and estimated speed within two weeks from August 1 to 14.

(b) Time series of actual and estimated speed within two weeks from September 12 to 25.

Fig. 4. The imputation performance of BGCP (CP rank $r = 15$ and missing rate $\alpha = 30\%$) under the fiber missing scenario with third-order tensor representation (B), where the estimated result of road segment #1 is selected as an example. In the both two panels, red rectangles represent fiber missing (i.e., speed observations are lost in a whole day).

structure (B) provides consistently good imputation performance with the increase of missing rate. The fiber missing scenario further proves the superiority of third-order tensor (see Table 2). Using the HaLRTC model as an example, structure (C) clearly goes in the wrong direction when the missing rate reaches $\alpha = 40\%$ and $\alpha = 50\%$.

The major difference between representations (B) and (C) is that (C) has taken the weekly effect into account explicitly, while the third-order tensor (B) only captures the hidden patterns in days and does not model the homogeneity and heterogeneity among different weeks. Our results suggest that the day and the time interval dimensions are sufficient to characterize the time series of traffic speed data. The "week" dimension does not exhibit enough variation for the decomposition model to define r latent factors on it, and as a result, it is not necessary to define this dimension. A well-known drawback of Bayesian methods is the computational cost in Markov chain Monte Carlo sampling. In our case, the BGCP model is also comparable with HaLRTC and STD. Each Gibbs iteration in BGCP takes less than one second if rank $r \leqslant 80$. In this case, training the model for 1,500 Gibbs iterations takes less than half an hour, while HaLRTC and STD in general take more than half an hour to converge.

5. Conclusions and discussion

In this paper, we present the Bayesian Gaussian CANDECOMP/PARAFAC (BGCP) decomposition model—as an extension of Bayesian probabilistic matrix factorization—to impute multi-dimensional incomplete traffic data. Since the same data set can be summarized into different structures (matrix, third-order tensor, and fourth-order tensor), we also consider data representation as a decision variable in field applications. Using a spatiotemporal traffic speed data set collected in Guangzhou, China, we compare the BGCP model with two other tensor completion methods: HaLRTC (Liu et al., 2013) and STD (Chen et al., 2018).

We design two missing scenarios: the random missing scenario and a more realistic fiber missing scenario to capture the temporally correlated corruptions in the traffic data. From the empirical analyses, we find that the BGCP model shows consistent/robust performance in both the random missing scenario and the fiber missing scenario. This fully Bayesian model has several advantages over other methods: (1) it allows us to have a robust imputation of missing values with prediction confidence taken into account; (2) by placing flexible priors/hyper-priors on model parameters/hyper-parameters, the Bayesian model can efficiently and automatically characterize the variation in the data and avoid overfitting.

In terms of tensor representation, we find that a third-order (road segment $\times$ day $\times$ time interval) gives the best results in both random missing and fiber missing scenarios. Ideally, the use of an additional dimension seems to be helpful, but this should be conditional on the fact that the additional dimension captures enough heterogeneity (latent factors). However, in our case, the week-to-week variation in the traffic speed data seems to be minor, and thus the week dimension does not require as many latent patterns as other dimensions. This finding is in contrast to the experiments of Ran et al. (2016), which suggests the use of a "week" dimension may achieve better performance.

The foundation of all tensor decomposition-based methods is the low-rank assumption. The traffic speed data set clearly shows this property and enables those tensor models to have good performance in our experiment. However, given the strong spatiotemporal correlation in urban traffic data, we can develop new models with increased flexibility and generalization power. We propose the following directions for future research:

- Determine the rank r in the BGCP model. In the case of random missing, we prefer choosing a larger r since the observed data still encodes a lot cross-dimensional information given the low-rank assumption. However, since information is lost in a temporally correlated manner in the fiber missing scenario, it becomes difficult to borrow information from other dimensions. Therefore, we prefer choosing a smaller r to characterize the main factors and avoid overfitting in the fiber missing case. One potential question is how to choose an appropriate rank r in different situations. As a future research topic, we would like to explore the use of non-parametric Bayesian methods (e.g., Dirichlet process) to determine r automatically given the size and complexity of the data.
- Develop a Bayesian Tucker Decomposition model. The CP decomposition may require a large number of redundant factors (r) when the data set gets larger, in particular when the numbers of latent factors of different dimensions are imbalanced. The Tucker decomposition can share factors across different dimension and thus is able to capture more information with less number of factors. For this topic, a critical challenge is to develop efficient MCMC/variational inference algorithms.
- Integrate spatiotemporal information explicitly in the decomposition model. One limitation of the factorization/decomposition-based models is that they only capture spatiotemporal patterns in a data-driven way, instead of physically coding the patterns. As a result, the model itself is invariant to random permutation on each dimension (or even on all the dimensions one by one). This fact inspires us to explicitly encode the external information such as spatial adjacency (e.g., using spatial auto-regressive (AR) models/graph embedding) and temporal correlation (e.g., using time series models) into the modeling framework.
- Apply this model for prediction tasks. While this paper only investigates the imputation problem for historical data, a more interesting problem is how to apply this framework for prediction tasks. Besides the dynamic tensor completion approach (Tan et al., 2016), a potential research direction is to integrate a physical prediction model (e.g., auto-regressive (AR), auto-regressive moving average (ARMA), and auto-regressive integrated moving average (ARIMA)) into the temporal dimension of the tensor to forecast future observations.

Acknowledgement

The authors would like to thank four anonymous referees for their valuable comments. This research is mainly supported by the Science and Technology Planning Project of Guangzhou, China (No. 201804020012), and partially supported by Open Funding of

Tongji University Road and Transport Engineering Key Laboratory (No. TJDDZHCX001). The data set for this project is available at https://doi.org/10.5281/zenodo.1205229 and the Matlab code is available at https://github.com/lijunsun/bgcp_imputation.

References

Anandkumar, A., Ge, R., Hsu, D., Kakade, S.M., Telgarsky, M., 2014. Tensor decompositions for learning latent variable models. J. Mach. Learn. Res. 15, 2773–2832.
Asif, M.T., Kannan, S., Dauwels, J., Jaillet, P., 2013a. Data compression techniques for urban traffic data. In: IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS), pp. 44–49.
Asif, M.T., Mitrovic, N., Dauwels, J., Jaillet, P., 2016. Matrix and tensor based methods for missing data estimation in large traffic networks. IEEE Trans. Intell. Transport. Syst. 17 (7), 1816–1825.
Asif, M.T., Mitrovic, N., Garg, L., Dauwels, J., Jaillet, P., 2013b. Low-dimensional models for missing data imputation in road networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 3527–3531.
Bae, B., Kim, H., Lim, H., Liu, Y., Han, L.D., Freeze, P.B., 2018. Missing data imputation for traffic flow speed using spatio-temporal cokriging. Transport. Res. Part C: Emerg. Technol. 88, 124–139.
Chen, X., He, Z., Wang, J., 2018. Spatial-temporal traffic speed patterns discovery and incomplete data recovery via SVD-combined tensor decomposition. Transport. Res. Part C: Emerg. Technol. 86, 59–77.
Duan, Y., Lv, Y., Liu, Y.-L., Wang, F.-Y., 2016. An efficient realization of deep learning for traffic data imputation. Transport. Res. Part C: Emerg. Technol. 72, 168–181.
Goulart, J.d.M., Kibangou, A., Favier, G., 2017. Traffic data imputation via tensor completion based on soft thresholding of tucker core. Transport. Res. Part C: Emerg. Technol. 85, 348–362.
Kolda, T.G., Bader, B.W., 2009. Tensor decompositions and applications. SIAM Rev. 51 (3), 455–500.
Laña, I., Olabarrieta, I.I., Vélez, M., Del Ser, J., 2018. On the imputation of missing data for road traffic forecasting: new insights and novel techniques. Transport. Res. Part C: Emerg. Technol. 90, 18–33.
Li, L., Li, Y., Li, Z., 2013. Efficient missing data imputing for traffic flow by considering temporal and spatial dependence. Transport. Res. Part C: Emerg. Technol. 34, 108–120.
Li, L., Zhang, J., Wang, Y., Ran, B., 2018. Missing value imputation for traffic-related time series data based on a multi-view learning method. IEEE Trans. Intell. Transport. Syst.
Li, Y., Li, Z., Li, L., 2014. Missing traffic data: comparison of imputation methods. IET Intell. Transport Syst. 8 (1), 51–57.
Liu, J., Musialski, P., Wonka, P., Ye, J., 2013. Tensor completion for estimating missing values in visual data. IEEE Trans. Pattern Anal. Mach. Intell. 35 (1), 208–220.
Ni, D., Leonard, J.D., Guin, A., Feng, C., 2005. Multiple imputation scheme for overcoming the missing values and variability issues in ITS data. J. Transport. Eng. 131 (12), 931–938.
Qu, L., Li, L., Zhang, Y., Hu, J., 2009. PPCA-based missing data imputation for traffic flow volume: a systematical approach. IEEE Trans. Intell. Transport. Syst. 10 (3), 512–522.
Rai, P., Wang, Y., Guo, S., Chen, G., Dunson, D., Carin, L., 2014. Scalable Bayesian low-rank decomposition of incomplete multiway tensors. In: Proceedings of the 31st International Conference on Machine Learning (ICML), vol. 32. pp. 1800–1808.
Ran, B., Tan, H., Wu, Y., Jin, P.J., 2016. Tensor based missing traffic data completion with spatial-temporal correlation. Phys. A: Stat. Mech. Appl. 446, 54–63.
Rodrigues, F., Henrickson, K., Pereira, F.C., 2018. Multi-output Gaussian processes for crowdsourced traffic data imputation. IEEE Trans. Intell. Transport. Syst.
Salakhutdinov, R., Mnih, A., 2008. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In: Proceedings of the 25th International Conference on Machine Learning (ICML). ACM, pp. 880–887.
Sun, L., Axhausen, K.W., 2016. Understanding urban mobility patterns with a probabilistic tensor factorization framework. Transport. Res. Part B: Methodol. 91, 511–524.
Tan, H., Feng, G., Feng, J., Wang, W., Zhang, Y.-J., Li, F., 2013. A tensor-based method for missing traffic data completion. Transport. Res. Part C: Emerg. Technol. 28, 15–27.
Tan, H., Wu, Y., Shen, B., Jin, P.J., Ran, B., 2016. Short-term traffic prediction based on dynamic tensor completion. IEEE Trans. Intell. Transport. Syst. 17 (8), 1524–9050.
Tang, J., Zhang, G., Wang, Y., Wang, H., Liu, F., 2015. A hybrid approach to integrate fuzzy c-means based imputation method with genetic algorithm for missing traffic volume data estimation. Transport. Res. Part C: Emerg. Technol. 51, 29–40.
Xiong, L., Chen, X., Huang, T.-K., Schneider, J., Carbonell, J.G., 2010. Temporal collaborative filtering with Bayesian probabilistic tensor factorization. In: SIAM International Conference on Data Mining, pp. 211–222.
Zhao, Q., Zhang, L., Cichocki, A., 2015a. Bayesian CP factorization of incomplete tensors with automatic rank determination. IEEE Trans. Pattern Anal. Mach. Intell. 37 (9), 1751–1763.
Zhao, Q., Zhang, L., Cichocki, A., 2015b. Bayesian Sparse Tucker Models for Dimension Reduction and Tensor Completion (Available from: arXiv: 1505.02343).