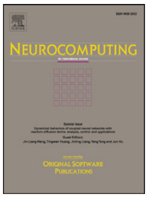
Neurocomputing 303 (2018) 47–59

Contents lists available at ScienceDirect

Neurocomputing

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



Graph regularized local self-representation for missing value imputation with applications to on-road traffic sensor data

Xiaobo Chen[a,b,*], Yingfeng Cai[a], Qiaolin Ye[c], Lei Chen[d], Zuoyong Li[e]

[a] *Automotive Engineering Research Institute, Jiangsu University, Zhenjiang 212013, China*
[b] *School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China*
[c] *College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China*
[d] *Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing University of Posts and Telecommunications, Nanjing 210003, China*
[e] *Fujian Provincial Key Laboratory of Information Processing and Intelligent Control, Minjiang University, Fuzhou 350108, China*

ARTICLE INFO

Article history:
Received 28 October 2017
Revised 26 March 2018
Accepted 24 April 2018
Available online 4 May 2018

Communicated by Liang Lin

Keywords:
Graph regularization
Local constraint
Matrix completion
Missing values
Traffic sensor data

ABSTRACT

Recovering missing values (MVs) from incomplete data is an important problem for many real-world applications. Previous research efforts toward solving MVs problem primarily exploit the global and/or local structure of data. In this work, we propose a novel MVs imputation method by combing sample self-representation strategy and underlying local linear structure of data in a uniformed framework. Specifically, the proposed method consists of the following steps. First, an existing method is applied to obtain the first-round estimation of MVs. Then, a graph, characterizing local proximity structure of data, is constructed based on imputed data. Next, a novel model coined as graph regularized local self-representation (GRLSR) is proposed by integrating two crucial elements: local self-representation and graph regularization. The former assumes each sample can be well represented (reconstructed) by linearly combining the neighboring samples while the latter further requires the neighboring samples should not deviate too much from each other after reconstruction. By doing so, MVs can be more accurately restored due to the joint imputation as well as local linear reconstruction. We also develop an effective alternating optimization algorithm to solve GRLSR model, thereby achieving final imputation. The convergence and computational complexity analysis of our method are also presented. To evaluate our method, extensive experiments are conducted on both traffic flow dataset and UCI benchmark datasets. The results demonstrate the effectiveness of our proposed method compared with a set of widely-used competing methods.

1. Introduction

Intelligent transportation system (ITS), as a promising application domain of numerous machine learning techniques, has attracted increasing attention from researchers in many scientific and industrial communities. Through comprehensively integrating various techniques such as information, computer, communication, pattern recognition and so on, ITS offers great promise for alleviating traffic congestion, improving road safety, enhancing driving experience, and improving traffic management in the context of smart cities. Yet in current ITS, the data available often suffer from missing value problem. For example, it was reported that [1] for a dense road network in the city of Melbourne, about 8% of sensors can reach up to 56% missing data and there are only approximately 1% of sensors without missing data [2]. Besides ITS, missing data frequently occurs in many other industrial, academic, business domains. Missing data may be caused by diverse reasons, such as hardware malfunction, failure of transmission network, data corruption, and other unexpected exogenous factors. As a result, the data obtained in reality is usually incomplete, posing some serious problems for subsequent data analysis tasks. For example, it may result in biased conclusions being drawn due to the difference between missing and complete data. It also impedes the application of existing mature data analysis algorithms (e.g., support vector machine [3,4]) because the input to these algorithms needs to be complete without missing values (MVs). Conventional approach that addresses this problem is to discard the samples with MVs in their attributes. This approach, however, suffers from some serious issues, such as (i) it loses the useful information contained in the partially observed attributes of those samples discarded, (ii) it may distort the distribution of samples and thus cause unintentional bias, especially when encountering with limited data.

Another popular way to overcome the challenges arising from missing data is imputation, a class of techniques capable of

[*] Corresponding author at: School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China.
E-mail address: xbchen82@gmail.com (X. Chen).

https://doi.org/10.1016/j.neucom.2018.04.029
0925-2312/© 2018 Elsevier B.V. All rights reserved.

replacing MVs with some plausible values [5,6]. It should be pointed out that recovering MVs accurately from arbitrary matrix is ill-posed, thus difficult or even impossible, without making any assumption about the data structure. The feasibility of MVs imputation is mainly attributed to certain underlying structure that data exhibits. Taking traffic sensor data as an example. Because of road network connectivity, spatial and temporal correlations with varying degrees exist between traffic samples captured by traffic sensors distributed in the road network. The profiles of traffic flow at neighboring road segments are highly correlated to each other. The obvious advantage of imputation approaches is that it does not need to discard samples with MVs, thus avoiding the issues aforementioned. Nevertheless, the recovery performance of MVs becomes a crucial problem since it exerts direct influence on subsequent data analysis tasks [7].

Many studies have been carried out in seek of accurate MVs imputation solutions during the last few decades. Some typical methods include mean imputation (MI), K-nearest neighbors (KNN) imputation, singularity value decomposition (SVD), local least squares regression (LLS) [8,9], probabilistic principal component analysis (PPCA) [10,11], matrix completion [12–15], etc. Among these methods, Low-rank matrix completion (LRMC), as one of the most successful MVs imputation methods, has attracted much attention because of its capability to recover a matrix with partially observed entries. LRMC enforces the given data matrix to be of low-rank structure to ensure that MVs can be reliably recovered with the help of rank minimization. However, this is not always the case for many real-world applications. To alleviate this drawback, MVs imputation based on sample self-representation (SR) was proposed recently [14], under the assumption that each sample can be well reconstructed by other samples belonging to a common linear subspace. This method relaxes the prerequisite of LRMC, thus being applicable to the setting of high-rank or even full-rank matrices. Among various variants of SR-based imputation [14], a sparse version termed as SRSp usually results in superior performance due to the incorporation of the l_1-norm based sparsity penalty for selecting few samples in the reconstruction of target sample. It should be emphasized that sparse and low-rank learning have drawn much attention during the last decades as they are capable of identifying inherent subspace structure of data. For example, in some computer vision tasks [16], e.g. object tracking [17–19], the features extracted from different patches within an image are believed to distribute on or close to multiple subspaces, thus allowing sparse or low-rank representation. However, most studies premise the full availability of raw data, which is violated in some applications.

Many existing MVs imputation approaches, including SVD, PPCA, LRMC, etc., treat the data as a whole and tend to exploit the global linear structure of data, without sufficiently accounting for the local geometric structure of real-world data. In reality, samples are probably sampled from much more complicated distributions. In such a case, the recovery performance of these methods will deteriorate due to the violation of global linear structure. A few other MVs imputation techniques, such as KNN, LLS, etc., are able to consider the local structure information of data, unfortunately, they tend to recover MVs in an individual way, thereby lacking of consistent constraint across the entire data space.

In this paper, aiming to improve the recovery performance of MVs, we propose a novel imputation approach termed as graph regularized local self-representation (GRLSR) which provides a consistent framework for recovering MVs, while incorporating local structure information of data. Our work is interesting from the following aspects:

1) We concentrate on each sample and its surrounding samples with high probability of residing on or close to a linear patch of data distribution. By doing this, each sample can be more reasonably represented by its neighbors through linear combination [20,21].
2) Local proximity structure, characterized by a graph [22,23], is further incorporated into the linear representation of each sample, ensuring that the reconstructed sample does not deviate too much from its neighbors.
3) To solve the resulting GRLSR model, we develop an efficient alternating optimization algorithm. The convergence and time complexity of our algorithm are also analyzed.
4) The proposed GRLSR is applied to real-world road network traffic sensor data and also extensively evaluated on multiple benchmark datasets. Experimental results clearly demonstrate that our method can achieve better imputation results than other competing approaches.

The paper is organized as follows. We begin in Section 2 by describing some related works for MVs imputation. Section 3 describes in detail the whole imputation flowchart and GRLSR model, including its mathematical model, optimization, and detailed analysis. Section 4 provides extensive experimental comparison of our approach with other state-of-the-art imputation approaches on traffic sensor data as well as a set of standard machine learning datasets. Finally, Section 5 contains a closing discussion of the paper and the future works needed.

2. Related work

Conventional methods for recovering MVs in data include mean imputation (MI), K-nearest neighbor (KNN), singular value decomposition (SVD), etc. For MI, the missing value on a certain variable is replaced by the average taken over the observed values of that variable for all samples. This method implicitly supposes the variables are independent of each other and obey normal distribution, without accounting for the intrinsic correlation structure between attributes. KNN imputation first selects K samples located in the vicinity of the sample with MVs, followed which the MVs can be estimated by weighted average of only those selected samples. It makes an assumption that the samples with small distance should have similar attributes. SVD is another popular baseline imputation method with a hypothesis that data is distributed in a low-dimensional linear subspace. It alternates between MVs recovery and SVD until convergence or the maximum iteration count is reached. SVD may work well on data with strong global correlation structure [24]. The aforementioned approaches are easy to implement, however, suffering from limited imputation performance. Over the past decade, a number of more advanced methods have been proposed in the literate and demonstrated significantly improved imputation performance by exploiting the correlation between data. In the following sections, we briefly discuss several typical methods classified into three major categories: probabilistic model based imputation [11,25], regression based imputation [9,26], and matrix completion based imputation [6,27].

2.1. Probabilistic model based imputation

This type of methods supposes the data we observe follows a specific probability distribution, e.g., multivariate Gaussian mixture model, which describes the statistical features of data. In probabilistic principal component analysis (PPCA) [28], maximum likelihood estimation is applied for the joint optimization of model parameters and MVs. The Expectation Maximization (EM) provides a unified framework for fitting the model and imputing MVs. Further, Bayesian PCA [25] integrates Bayesian learning and PPCA such that effective dimensionality of the latent space (the number of retained principal components) can be determined automatically through Bayesian inference. These methods belong to generative latent variable model and are suitable when a global structure is dominant in data. In addition, although this type of methods and SVD both depends on global structure of data, the former generally works better owing to complete theoretical and algorithmic foundation. One major disadvantage of these methods is that their performance is heavily dependent on the prior assumption about data distribution which is unknown in practice. Actually, due to the diversity of data generation mechanism, it may be improper to postulate a uniformed distribution for different types of data.

2.2. Regression based imputation

This type of methods attempts to model the inherent relationship between variable with MVs and other observed variables by means of various regression techniques, such as least squares [8,9,26,29], support vector machine [4,30,31], neural networks [32], etc. Despite some difference in terms of specific regression models, these methods share common motivation. Local least squares (LLS) [8], a representative regression based imputation method, was originally developed for DNA microarray data [26], but also applicable for other type of data [9]. Sharing similar flavor with KNN imputation, LLS also needs to selects K most similar genes for the target gene with MVs. However, instead of weighted average as used in KNN, LLS aims to characterize the relationship between target gene and its surrounding neighbors based on least squares regression. In such a way, LLS exploits the local similarity structure in the data and has been successfully applied to traffic sensor data with encouraging results [9]. Overall, benefiting from having a local connotation, KNN and LLS methods success in utilizing the local relationship of data. Unfortunately, these methods recover the MVs in data individually, thus failing to consider the consistency across the whole sample space.

2.3. Matrix completion based imputation

In this type of methods, data is organized in matrix form and the MVs are recovered based on specific assumptions about data matrix. Low-rank matrix completion (LRMC) [12] assumes that data matrix is intrinsically of low-rank, indicating the dimension of vector space spanned by its rows (columns) is small with respect to the size of matrix. Then, the MVs can be recovered through rank minimization (or its surrogate nuclear norm) on the whole matrix. This method has been successfully applied in many different domains, such as traffic sensor data [33,34], web service tag refinement [35], etc. However, assuming the whole data matrix has a global low-rank structure is too restrictive for problems with rich structure, thus hindering the applications of LRMC. More recently, matrix self-representation based imputation was proposed [14], with the assumption that each sample can be well represented by a linear combination of other samples [36]. By imposing l_1-norm based sparsity regularization on combinatorial coefficients, a variant termed as SRSp has been proved to be effective because it is able to select few samples in the representation of target sample. Despite encouraging results, they usually disregard the local structure of data [8,37]. As indicated by recent studies [38,39], the local information of data is more essential than sparsity for many tasks. For instance, enforcing the local constraint can naturally incur sparsity [39] since the faraway samples will take no effect in representation, however, it is not always the case conversely. In addition, SRSp has high computational burden, especially when the sample size is large, because of the time-consuming sparse optimization.

3. The proposed methodology

In this section, we first present the entire framework proposed for MVs imputation. Then, graph regularized local self-representation (GRLSR) model is formulated. After that, an efficient iterative algorithm with detailed analysis on the convergence and computational complexity is presented.

Formally, let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in R^{M\times N}$ be a sample matrix, where $\mathbf{x}_i = [x_i(1), x_i(2), \ldots, x_i(M)]^T$ denotes a sample with M features, N is the total number of samples. Not all of the entries in a certain sample $\mathbf{x_i}$ are known. Let r_i be the index set of observed entries in $\mathbf{x}_i$, Correspondingly, we use $\overline{r_i}$ to denote the index set of missing or unobserved entries in $\mathbf{x}_i$.

3.1. Framework of the method

The proposed MVs imputation framework is intuitively shown in Fig. 1. As we can see, it consists of the following steps:

Step 1. An existing MVs imputation algorithm, such as LLS, LRMC etc., is applied on the whole sample set $\mathbf{X}$ so as to obtain the first-round estimation, denoted by $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_N]$, where $\tilde{x}_i(k) = x_i(k)$, if $k \in r_i$ (i.e., $x_i(k)$ is known).

Step 2. Based on the imputed samples, a distance matrix $\mathbf{D} = [d_{ij}] \in R^{N\times N}$, with d_{ij} measuring the pairwise distance between two distinct samples $\mathbf{x}_i$ and $\mathbf{x}_j$, is calculated by weighted Euclidean distance as follows

$$d_{ij} = \sqrt{\sum_{k=1}^{M} \tilde{\theta}_k \big(\tilde{x}_i(k) - \tilde{x}_j(k)\big)^2} \qquad (1)$$

where $\tilde{\theta}_k = \frac{\theta_k}{\sum_{l=1}^{M}\theta_l}$, $\theta_k = 1$, if $k \in r_i \cap r_j$ (i.e., both $x_i(k)$ and $x_j(k)$ are observed), otherwise $\theta_k = \alpha$ where α is a small positive number fixed to be 0.1 in this paper.

Step 3. KNN [23,40], as one of the most popular methods, is used to construct a nearest neighbor graph characterizing the proximity relationship between samples. Specifically, let $Ne(i)$ denote the index set of the K nearest neighbors of x_i based on distance matrix $\mathbf{D}$. Then, similar to [36], a proximity matrix $\mathbf{S} = [s_{ij}] \in R^{N\times N}$ can be defined as

$$s_{ij} = \begin{cases} 1, & \text{if } j \in Ne(i) \text{ or } i \in Ne(j) \\ \varepsilon, & \text{otherwise} \end{cases} \qquad (2)$$

where ε is a very small positive number.

Step 4. GRLSR model, explained in the next section, is constructed based on original sample matrix $\mathbf{X}$ and the constructed weight matrix $\mathbf{S}$. After solving GRLSR, we can obtain the final estimation $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N]$.

It should be pointed out that the selection of existing MV imputation in the above ***Step 1*** is dependent on the specific task/data. In this work, LRMC and LLS are chosen as initialization algorithm for traffic sensor data and UCI benchmark data, respectively. We have evaluated the impact of different initialization on the final MVs estimation in the experimental section. In addition, we call the whole imputation framework (from ***Step 1*** to ***Step 4***) as GRLSR, not only GRLSR model constructed in ***Step 4***.

3.2. GRLSR model

The first motivation of GRLSR is that with properly estimated MVs, each sample can be represented as a linear combination of other samples located in the vicinity of that sample [20]. To be specific, we define the following local self-representation

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{Y}} \|\mathbf{Y} - \mathbf{YW}\|^2 + \lambda_1 \|\mathbf{P} \odot \mathbf{W}\|^2 \\ &\text{s.t. } \operatorname{diag}(\mathbf{W}) = 0,\ \mathbf{Y}_i(k) = \mathbf{X}_i(k),\ k \in r_i, i = 1, 2, \cdots, N \end{aligned} \qquad (3)$$

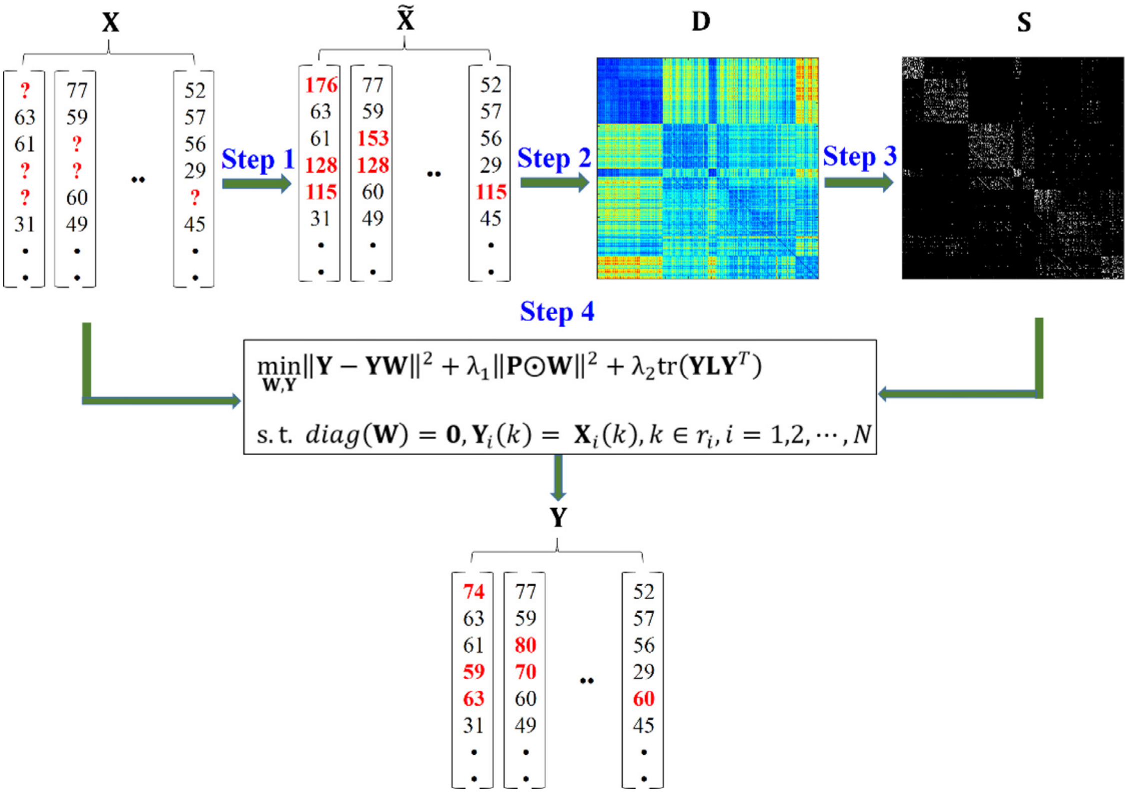

X
? 63 61 ? ? 31
77 59 ? ? 60 49
52 57 56 29 ? 45
Step 1
X̃
176 63 61 128 115 31
77 59 153 128 60 49
52 57 56 29 115 45
Step 2
D
Step 3
S
Step 4
$\min_{\mathbf{W},\mathbf{Y}} \|\mathbf{Y}-\mathbf{YW}\|^2+\lambda_1\|\mathbf{P}\odot\mathbf{W}\|^2+\lambda_2\mathrm{tr}(\mathbf{YLY}^T)$
s.t. $diag(\mathbf{W})=\mathbf{0}, \mathbf{Y}_i(k)=\mathbf{X}_i(k), k\in r_i, i=1,2,\cdots,N$
Y
74 63 61 59 63 31
77 59 80 70 60 49
52 57 56 29 60 45

Fig. 1. Illustration of the proposed framework for MVs imputation.

where $\mathbf{Y}=[\mathbf{y}_1,\mathbf{y}_2,\ldots,\mathbf{y}_N]$ denotes the data matrix to be estimated, $\mathbf{W}=[\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_N]\in R^{N\times N}$ is the coefficient matrix with the i-th column $\mathbf{w}_i$ containing the contributions of other samples when representing $\mathbf{x}_i$, $\mathbf{P}=[\mathbf{p}_1,\mathbf{p}_2,\ldots,\mathbf{p}_N]\in R^{N\times N}$ is a weight matrix for $\mathbf{W}$, the operator $\odot$ means the Hadamard (element-wise) multiplication, λ_1 is a positive trade-off parameter balancing the importance of l_2-norm regularization for reducing overfitting, diag($\mathbf{W}$) is the vector containing the diagonal elements of $\mathbf{W}$, 0 is the vector containing zeros as its elements. In this work, we have $p_i(j)=s_{ij}^{-1}$ implying that if some $\mathbf{y}_j$ falls outside of $\mathbf{y}_i$'s neighborhood, the corresponding weight $p_i(j)$ will become so large that $w_i(j)\to 0$ in the optimization of (3). As a result, $\mathbf{y}_i$ will be represented exclusively by those samples with small distance from $\mathbf{y}_i$ [21]. The constraint diag($\mathbf{W}$) = 0 is applied so as to avoid trivial solution $w_i(i)=1,\ 1\le i\le N$.

Meanwhile, it is desirable that neighboring samples tend to have similar characteristics and the estimated MVs should be consistent with such proximity structure. Namely, the distances between all pairs of neighboring samples should be restricted in a proper range after imputation, thus preventing the imputed sample from deviating too much from its neighboring samples. Therefore, we propose the following criterion to explore such local proximity structure

$$\frac{1}{2}\sum\nolimits_{i,j} s_{ij}\|\mathbf{y}_i-\mathbf{y}_j\|^2=\mathrm{tr}\left(\mathbf{YLY}^T\right) \tag{4}$$

where $\mathbf{L}=\mathbf{E}-\mathbf{S}$, $\mathbf{E}$ is a diagonal matrix with nonzero element $\mathbf{E}_{ii}=\sum_{j=1}^{N} s_{ij}$, $\mathbf{L}$ is called graph Laplacian matrix [23,41], characterizing local geometric structure of data, tr(·) means the matrix trace operator.

Finally, by combining (3) and (4), we obtain the objective function for GRLSR as follows

$$\min_{\mathbf{W},\mathbf{Y}} \|\mathbf{Y}-\mathbf{YW}\|^2+\lambda_1\|\mathbf{P}\odot\mathbf{W}\|^2+\lambda_2\mathrm{tr}\left(\mathbf{YLY}^T\right)$$
$$\text{s.t. diag}(\mathbf{W})=0,\ \ \mathbf{Y}_i(k)=\ \mathbf{X}_i(k),\ \ k\in r_i, i=1,2,\cdots,N \tag{5}$$

where λ_2 is another positive parameters controlling the importance of graph regularization.

3.3. Optimization algorithm

The problem (5) does not allow a closed form optimal solution due to the coupling between decision variables $\mathbf{W}$ and $\mathbf{Y}$. Nevertheless, we notice that the objective is convex in $\mathbf{W}$ (while holding $\mathbf{Y}$ fix) and $\mathbf{Y}$ (while holding $\mathbf{W}$ fix), respectively [42]. To this end, we choose to iteratively optimize the objective (5) by alternatingly optimizing over $\mathbf{W}$ and $\mathbf{Y}$ while holding the other variable fixed.

First, we consider how to solve $\mathbf{W}$ when $\mathbf{Y}$ is fixed. When $\mathbf{Y}$ is fixed, problem (5) can be decomposed into a series of equations as follows

$$\min_{\mathbf{w}_i} \|\mathbf{y}_i-\mathbf{Y}\mathbf{w}_i\|^2+\lambda_1\|\mathbf{p}_i\odot\mathbf{w}_i\|^2$$
$$\text{s.t. w}_i(i)=0 \text{ for } i=1,2,\cdots,N \tag{6}$$

By substituting the constraint into the objective of (6), we obtain the following unconstrained problem

$$\min_{\tilde{\mathbf{w}}_i} \left\|\mathbf{y}_i-\tilde{\mathbf{Y}}_i\tilde{\mathbf{w}}_i\right\|^2+\lambda_1\|\tilde{\mathbf{p}}_i\odot\tilde{\mathbf{w}}_i\|^2 \tag{7}$$

where $\tilde{\mathbf{Y}}_i\in R^{M\times(N-1)}$ denotes the matrix $\mathbf{Y}$, but excluding the i-th column, and similarly for $\tilde{\mathbf{p}}_i$ and $\tilde{\mathbf{w}}_i$. Let $\mathbf{P}_i$ be a diagonal matrix with the elements of vector $\tilde{\mathbf{p}}_i$ on the main diagonal. Then, problem (7) allows a closed-form solution given by

$$\tilde{\mathbf{w}}_i=\left(\tilde{\mathbf{Y}}_i^T\tilde{\mathbf{Y}}_i+\lambda_1\mathbf{P}_i\right)^{-1}\tilde{\mathbf{Y}}_i^T\mathbf{x}_i \tag{8}$$

Then, when $\mathbf{W}$ is fixed, we need to solve $\mathbf{Y}$. In such a case, we define

$$\min_{\mathbf{Y}} f(\mathbf{Y})=\|\mathbf{Y}-\mathbf{YW}\|^2+\lambda_2\mathrm{tr}\left(\mathbf{YLY}^T\right)$$
$$\text{s.t. }\mathbf{Y}_i(k)=\ \mathbf{X}_i(k),\ \ k\in r_i, i=1,2,\cdots,N \tag{9}$$

Algorithm 1. Graph regularized local self-representation for MVs imputation.

```
Input:
Incomplete sample matrix $\mathbf{X}$, parameters $K$, $\lambda_1$, and $\lambda_2$
Procedure:
S1: Initialize the MVs in $\mathbf{X}$ using an existing imputation algorithm and get a complete data matrix $\tilde{\mathbf{X}}$;
S2: Construct pairwise distance matrix $\mathbf{D}$ using (1) and $\tilde{\mathbf{X}}$, proximity matrix $\mathbf{S}$ using $K$ nearest neighboring search and (2);
S3: Construct weight matrix $\mathbf{P}$, and graph Laplacian matrix $\mathbf{L}$;
S4: Let $\mathbf{Y} = \tilde{\mathbf{X}}$, the iteration step $t = 0$;
S5: repeat
S6:   Compute the representation coefficient matrix $\mathbf{W}$ column by column using (8);
S7:   repeat
S8:     Compute gradient $\nabla f(\mathbf{Y})$ by following (10);
S9:   Let $\nabla f(\mathbf{y}_i)(k) = 0$, if $k \in r_i, \forall i = 1, 2, \cdots, N$
S10:   Find step-size $l$ with Armijo rule, i.e., choose $l = \max\{1, \frac{1}{2}, \frac{1}{4}, \cdots\}$ such that
       $f(\mathbf{Y}) - f(\mathbf{Y} + l\nabla f(\mathbf{Y})) \geq \frac{l}{4}||\nabla f(\mathbf{Y})||^2$;
S11:     Update MVs estimation $\mathbf{Y} \leftarrow \mathbf{Y} + l\nabla f(\mathbf{Y})$;
S12:   until convergence criterion satisfied
S13:   $t = t + 1$;
S14: until convergence criterion satisfied
Output:
Imputed sample matrix $\mathbf{Y}$
```

In order to optimize $f(\mathbf{Y})$ with respect to $\mathbf{Y}$ and take into account the constraint, we apply the projected gradient descent method with Armijo rule [43]. The derivative of $f(\mathbf{Y})$ with respect to $\mathbf{Y}$ can be easily computed as

$$\nabla f(\mathbf{Y}) = 2\mathbf{Y}\Big((\mathbf{I} - \mathbf{W})(\mathbf{I} - \mathbf{W})^T + \lambda_2 \mathbf{L}\Big) \tag{10}$$

where $\nabla f(\mathbf{Y}) = [\nabla f(\mathbf{y}_1), \nabla f(\mathbf{y}_2), \ldots, \nabla f(\mathbf{y}_N)]$ and $\mathbf{I}$ is an identity matrix with proper size.

For clarity, the whole algorithm for MVs imputation using the proposed GRLSR is summarized in Algorithm 1.

3.4. Convergence analysis

The optimization procedure of GRLSR model (5) can be divided into two subproblems formulated in (6) and (9). The first subproblem boils down to a series of least squares regression, each of which has an optimal solution given by (8). That is, the value of objective function (5) will decrease after solving $\mathbf{W}$ in this subproblem. For the second subproblem, since objective function (9) is also convex in $\mathbf{Y}$, the gradient descent with Armijo rule will find the globally minimum value of (9) regardless of initialization. Therefore, by solving the two subproblems alternatively, the objective (5) will decrease gradually in each iteration. Finally, considering (5) is lower-bounded with zero, we can say that Algorithm 1 is convergent according to the “Cauchy’s convergence rule” [44]. In the experimental section, we will show that Algorithm 1 converges very fast.

3.5. Time complexity analysis

Now, we briefly analyze the computational complexity of the main steps involved in Algorithm 1. Here, we suppose $N > M$, namely, the amount of samples is larger than the number of features.

The time complexity of S1 depends on the specific model and optimization algorithm, thus varying over a wide range. For example, if we use LRMC as initialization model and solve it by traditional singular value thresholding (SVT), the time complexity can be estimated as $O(tM^2N)$, where t is the number of iteration. On the other hand, if mean imputation (MI) is adopted, the time complexity is just $O(M)$. The time complexity of S2, including pairwise distance computation and sorting, also depends on specific algorithm. By using quick select algorithm, the time complexity can be estimated as $O(N^2M)$. In S6, we need to solve a series of linear system of equations, just like (8). It seems that the complexity for this step is very high when confronting large sample set. However, we can immediately notice that this equation solely relies on the neighbors of each sample instead of the entire sample set. Let K be the number of neighbors, then the complexity of solving least squares problem is $O(K^3)$, thus the complexity of S6 is $O(NK^3)$. For S7–S12, the time complexity for gradient computation (10) is $O(MN^2)$. Let t_1 be the number of iterations required in S7–S12, then the time complexity for the inner gradient descent takes $O(t_1MN^2)$ in complexity. Then the total time complexity for S5–S14 is $O(t_2(K^3 + t_1MN^2))$ where t_2 is the total number of iterations.

4. Experiments

4.1. Traffic sensor data and setting

We first apply the proposed GRLSR approach to a real-world traffic flow volume data in order to evaluate its imputation performance. The data comes from Interstate 205 (I205) interstate highways, serving the Portland—Vancouver metropolitan area in the U.S. states of Oregon and Washington. The data contains the vehicle volume counts recorded by 10 inductive loop sensors and can be downloaded from the website (http://portal.its.pdx.edu/). The data collection is conducted from Mar. 1st to Aug. 31st in the year 2015. After excluding all holidays and date with MVs, we get volume data of 97 days for experiments. The aggregation period of the number of passing-through vehicles is 15 min, thereby generating a total of 96 samples in each day. Thus, the total number of traffic volumes is $96 \times 97 \times 10 = 93{,}120$.

Fig. 2 shows the traffic flow profile in the same day for each of the 10 loop sensor. As can be seen, different sensors are subject to daily traffic flow profiles with large variations. For some sensors, the traffic peaks have higher amplitude at noon while for others, the peaks might have lower amplitude. From another viewpoint, for some sensors, the traffic peaks occur earlier while for others, the peaks occur later, although this type of time-shift is relatively small.

To reflect the complex distribution of MVs, we simulate three MVs scenarios that arise commonly [10,45]. (i) Missing completely at random (MCAR) where the propensity for a data point to be missing is completely random, i.e., independent of the observed data and the other missing data. In this pattern, MVs appear as a set of isolated points randomly distributed. (ii) Missing at random (MAR) where the occurrence of MVs depends on its neighboring MVs. As a result, this pattern looks like a group of successive MVs. (iii) A mixture of MCAR and MAR (MIXED), where the mixing ratio for MCAR and MAR is 0.5, indicating half of the MVs are from

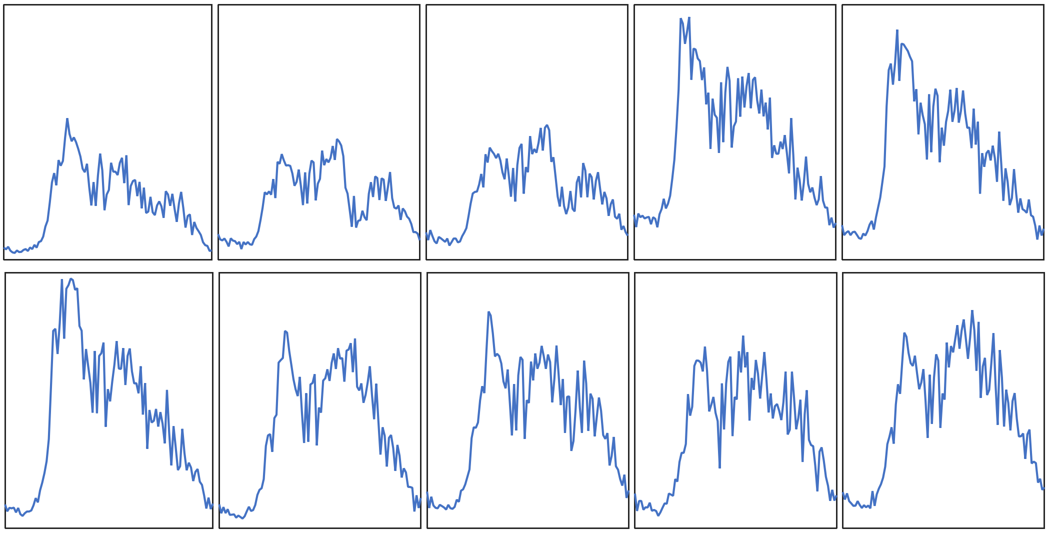


Fig. 2. Illustration of traffic flow profiles from 10 sensors in the same day.

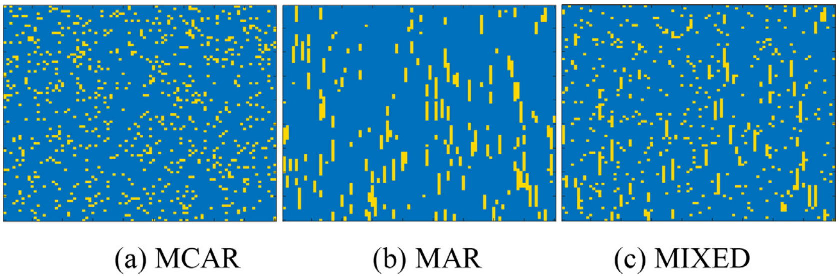


Fig. 3. Simulated examples by three missing patterns. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

MCAR while the other half are from MAR. To clearly illustrate their difference, we show an example of these simulated missing patterns in Fig. 3 with each column denoting a traffic sample and each row denoting a variable. The blue and yellow squares stand for the observed and missing values, respectively.

To comprehensively evaluate the effectiveness of GRLSR, we compare it with several closely related methods, including MI, KNN, SVD, LLS, PPCA, LRMC, and SRSp. Among them, MI is usually regarded as the baseline for MVs imputation, while the other belongs to different classes of methods (according to the taxonomy described in Section 2), e.g., regression model, probabilistic model, and matrix completion model. SRSp is selected because it shows best performance among their proposal models by following [14]. All methods are implemented in MATLAB 2015a on a PC with Core i7 processor and 12 GB RAM. In experiments, missing entries are generated artificially and then different methods are used to get an estimation. Intuitively, it is highly desirable that the estimated values approximate the real values as accurately as possible. Therefore, root mean squared error (RMSE), a widely employed evaluation metric, is calculated for quantitatively measuring the recovery performance of MVs imputation method:

$$RMSE = \sqrt{\frac{1}{|\Omega|}\left\| X_{\Omega}^{true} - X_{\Omega}^{impu} \right\|^2} \tag{11}$$

where Ω denotes the set of missing entries, $|\Omega|$ stands for the total number of MVs, X_{Ω}^{true} and X_{Ω}^{impu} denote the real value and imputed value, respectively. Clearly, smaller RMSE means better imputation performance. We also define missing ratio δ as the ratio of the number of MVs to the total number of values, and change the value of δ from 0.1 to 0.5 with step 0.1 so as to simulate imputation problem with varying difficulty. As for the selection of initial imputation algorithm, we apply LRMC because it has shown better performance comparing with other algorithms [6,15].

There are some parameters need to be set in each method. Following previous works [14], the parameters in KNN, SVD, LLS, PPCA, LRMC, and SRSp are optimized to achieve the best recovery performance by grid searching. For our algorithm, three parameters (the neighbor parameter K, balance parameters λ_1 and λ_2) are involved. Unless otherwise mentioned, we empirically let $K = 20$, $\lambda_1 = 5e + 5$, and $\lambda_2 = 0.01$ for this data.

4.2. Imputation error

Tables 1–3 list the imputation errors of different algorithms under MCAR, MAR, and MIXED missing patterns, respectively. We have repeated each test for 10 times and calculated the average errors of each method. The best results in these tables are marked in bold. We can see some interesting points from these tables. Firstly, MCAR and MAR are, respectively, the easiest and hardest situation among three missing patterns, regardless of concrete MVs imputation algorithm. The imputation error of each method increases with the missing ratio. Secondly, the baseline MI is the worst in terms of imputation accuracy because it relies on the normal distribution assumption while ignoring complex intrinsic structure of traffic samples. Thirdly, KNN, SVD, LLS, LRMC, PPCA, and SRSp all outperform MI in varying degrees because they attempt to explicitly model the inherent structure of data through adopting more sophisticated techniques. Among these five

Table 1
Imputation error obtained by different methods under MCAR missing pattern.

δ	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
0.1	206.83 ± 1.27	98.17 ± 1.38	94.11 ± 1.00	93.39 ± 1.40	85.35 ± 1.08	84.93 ± 1.06	82.89 ± 0.99	**73.33 ± 1.19**
0.2	206.45 ± 0.63	105.88 ± 1.09	95.89 ± 0.89	98.48 ± 0.92	87.62 ± 0.74	88.72 ± 0.66	84.95 ± 0.72	**77.28 ± 0.78**
0.3	206.67 ± 0.70	112.77 ± 0.98	96.83 ± 0.69	106.56 ± 1.02	90.04 ± 0.70	90.60 ± 0.71	87.56 ± 0.63	**81.27 ± 1.13**
0.4	206.82 ± 0.47	119.98 ± 0.38	98.25 ± 0.70	117.36 ± 1.21	93.11 ± 0.61	93.04 ± 0.60	90.77 ± 0.57	**87.97 ± 0.63**
0.5	206.75 ± 0.42	128.58 ± 0.64	100.89 ± 0.48	129.67 ± 1.07	96.65 ± 0.46	95.84 ± 0.53	94.26 ± 0.43	**91.39 ± 0.88**

Table 2
Imputation error obtained by different methods under MAR missing pattern.

δ	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
0.1	211.55 ± 3.17	119.41 ± 2.35	105.38 ± 2.04	155.11 ± 4.59	98.61 ± 1.74	93.21 ± 1.47	93.27 ± 1.05	**80.78 ± 1.73**
0.2	211.76 ± 1.78	124.34 ± 1.21	106.57 ± 1.24	163.24 ± 4.06	100.77 ± 1.00	97.72 ± 0.82	96.06 ± 0.77	**85.21 ± 1.32**
0.3	210.82 ± 1.28	128.79 ± 1.47	108.38 ± 1.57	172.24 ± 5.11	102.48 ± 1.11	99.34 ± 0.89	98.17 ± 0.89	**89.32 ± 1.16**
0.4	210.00 ± 0.60	133.54 ± 0.96	110.20 ± 0.98	179.85 ± 0.80	104.61 ± 1.18	100.79 ± 1.14	100.52 ± 1.11	**93.41 ± 1.25**
0.5	209.62 ± 0.42	139.85 ± 1.06	111.34 ± 1.11	181.12 ± 0.79	107.30 ± 1.20	102.92 ± 1.12	103.44 ± 1.20	**97.84 ± 1.14**

Table 3
Imputation error obtained by different methods under MIXED missing pattern.

δ	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
0.1	208.49 ± 2.13	108.83 ± 1.96	99.68 ± 0.60	126.11 ± 2.30	91.75 ± 1.03	88.99 ± 1.22	87.57 ± 0.93	**76.91 ± 1.51**
0.2	209.09 ± 2.06	116.00 ± 1.80	101.40 ± 0.98	131.30 ± 3.27	94.44 ± 0.80	92.95 ± 0.80	90.69 ± 0.63	**80.86 ± 1.47**
0.3	209.18 ± 0.85	122.02 ± 0.83	102.39 ± 0.78	133.46 ± 1.29	96.88 ± 0.93	94.96 ± 0.62	93.22 ± 0.73	**85.29 ± 1.19**
0.4	208.52 ± 0.81	127.70 ± 0.97	104.16 ± 0.74	139.78 ± 2.06	98.97 ± 0.79	97.04 ± 0.47	95.76 ± 0.68	**89.70 ± 0.71**
0.5	208.13 ± 0.78	135.51 ± 1.05	107.79 ± 0.89	149.22 ± 1.07	102.76 ± 0.64	99.97 ± 0.48	99.61 ± 0.40	**94.78 ± 0.76**

algorithms, SRSp achieves superior performance, further verifying the conclusions drawn in [14]. It also indicates the complexity of data in real-world applications. Fourthly, our proposed GRLSR works much better than the other competing methods. Actually, it reduces RMSE achieved by the best comparison method by 3–11%.For clear illustration, we show in Fig. 4 some imputation results obtained by different methods in the case of mixed missing pattern and $\delta = 0.3$.We can see that the imputation residual obtained by GRLSR is smaller than that of other methods.

4.3. Influence of parameters on imputation error

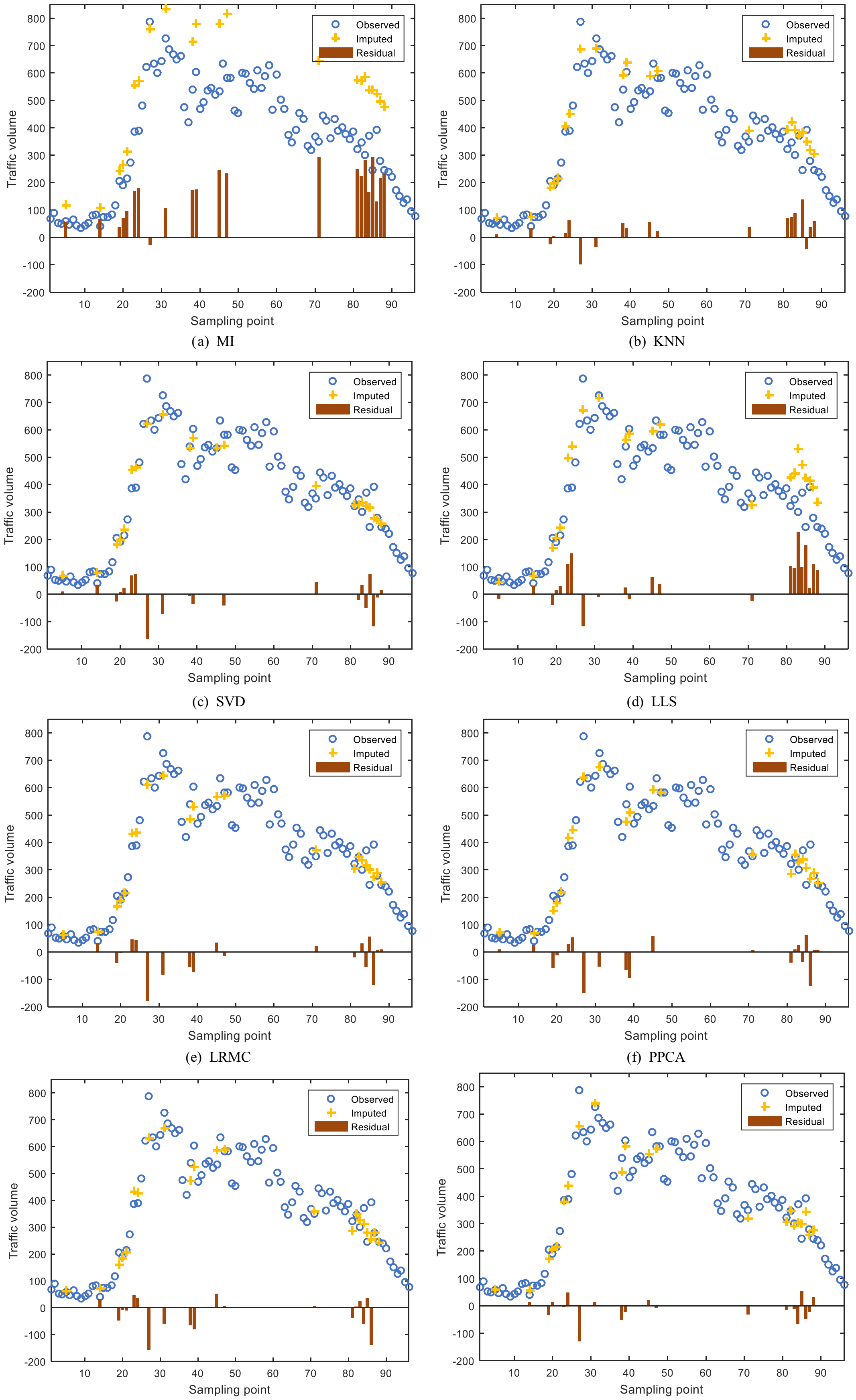
As in many other MVs imputation approaches, our proposed GRLSR also requires several parameters K, λ_1, and λ_2 to be set in advance. As discussed in the Methodology section, K determines the number of neighbors used to reconstruct each sample, λ_1 controls the strength of l_2-norm regularization on the coefficient matrix, λ_2 controls the impact of local proximity based regularization. In order to investigate the variation of imputation error with respect to these parameters, we change one parameter each time, while holding the other two fixed. The experimental results under different settings are shown in Fig. 5. From these results, we can observe that small or large parameters both lead to degraded performance. For instance, when K becomes smaller than 20 or larger than 30, the imputation errors increase accordingly. In contrast, when $K \in [20, 30]$, the imputation errors are kept to a low level. This phenomenon can be understood from two aspects. On one hand, when K is too small, very few neighboring samples are chosen to represent the target sample, thus lacking of enough information to recover MVs. On the other hand, when K is too large, those samples faraway from (dissimilar to) the target sample are involved in the reconstruction, which inevitably distorts the recovery of MVs. We can draw similar conclusions regarding the influence of λ_1 and λ_2.

4.4. Influence of initialization on imputation error

As can be noted from the step 1 of GRLSR algorithm, an existing method should be chosen to obtain an initial estimation of MVs, based on which a graph characterizing the neighboring relationship between samples can be built. As a result, this step will have a certain impact on the final imputation error of GRLSR. In this section, we discuss the sensitivity of GRLSR to different initialization methods. Taking MIXED missing data pattern as an example, the experimental results using MI, KNN, SVD, LLS, LRMC, and PPCA as initialization algorithm are presented in Fig. 6. As can be seen, different initialization does influence the recovery performance of GRLSR to a certain extent. For instance, better initialization generally leads to better performance. Nevertheless, the difference of performance using different initialization is not very large. Two factors may contribute to the robustness of GRLSR under different initialization. First, we apply weighted Euclidean distance, assigning smaller weight to the estimated values than to the observed values. Second, the proximity relationship between samples is relatively stable when varying pairwise distance within a certain range.

4.5. Results on UCI benchmark data

In this section, we further compare different MVs imputation methods on 6 publicly available UCI datasets, widely applied in the evaluation of various machine learning algorithms. LLS was employed to obtain an initial estimation of MVs for each dataset. Table 4 summarizes the basic information of these datasets, where



(a) MI

(b) KNN

(c) SVD

(d) LLS

(e) LRMC

(f) PPCA

(g) SRSp

(h) GRLSR

Fig. 4. Illustration of the imputation results obtained by different methods.

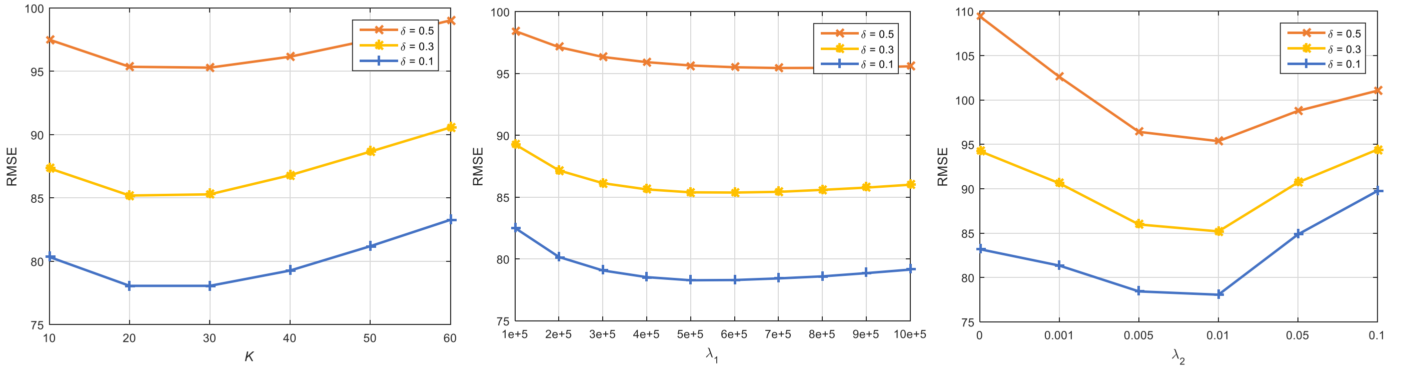


Fig. 5. Performance variation of the proposed method *w.r.t.* different values of the parameters K, λ_1, and λ_2.

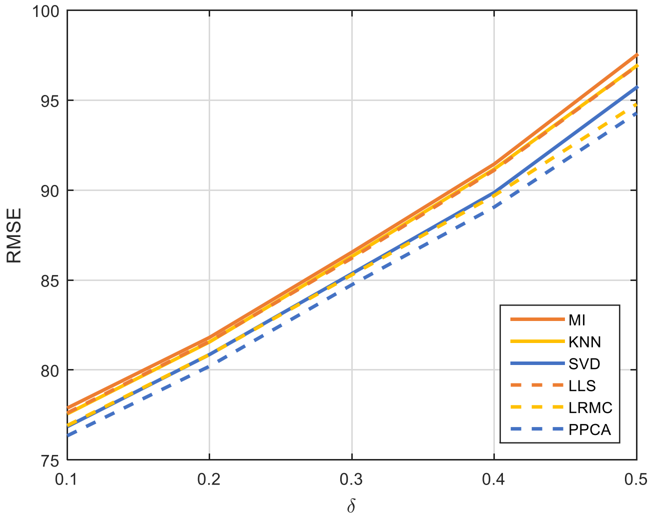


Fig. 6. Performance variation of the proposed method *w.r.t.* different initial imputation algorithms.

Table 4
UCI dataset description.

Property	Data					
	heart	iono	liver	pima	spectf	votes
n	270	352	345	768	267	435
d	13	32	6	8	44	16

n and d denote the number of samples and features, respectively. We follow similar evaluation procedure as the traffic sensor data and show the imputation errors obtained by each method in Tables 5–9. It can be found that MI produces poor results on all datasets, confirming that the variables probably follow a complex distribution instead of normal distribution. KNN, SVD, LLS, LRMC, PPCA, and SRSp all improve imputation performance to a large extent. Overall, the proposed GRLSR achieves smallest imputation errors on most datasets. These results demonstrate that GRLSR is an effective MVs imputation method under varying missing ratios.

4.6. Study of computational time and convergence

In this section, we compare the average CPU-time of the involved methods in each independent run under MCAR, MAR and MIXED patterns. Traffic flow data is used for evaluation. Note that for a fair comparison, the computational time of our method includes the total time consumed in all steps as shown in Fig. 1, not just the time taken in solving model (3). The experimental results are reported in Fig. 7(a)–(c). We can see from these figures that for the proposed GRLSR (and other competing methods), the computational time gradually increases with the missing ratio δ. This is mainly because large value of δ results in more missing entries, thus rendering the MVs imputation problem more difficult to solve. Comparing different MVs imputation approaches, we find that MI, KNN, SVD, LLS, LRMC, and PPCA all perform fast. Nevertheless, as shown in the above section, their imputation performance is limited. The recently proposed SRSp is able to achieve smaller imputation error, however, with remarkably increased computational load. As mentioned before, SRSp applies l_1-norm based sparse representation on all samples and uses ADMM as optimization algorithm. There is general consensus that [46] though ADMM can be used to solve large-scale sparse optimization problem with modest accuracy, it can be very slow to converge. Lastly, although our proposed GRLSR is still slower than MI, KNN, SVD, LLS, LRMC, and PPCA in terms of speed, it performs about 2–9 times faster than SRSp, depending on different combinations of missing patterns and missing ratios. We attribute this acceleration to the advantages of our algorithm which integrates local representation and l_2-norm based regularization, thus avoiding time-consuming computation on the whole dataset.

In this work, an iterative algorithm with guaranteed convergence is developed in Section 3.3 to solve the proposed GRLSR model (3). Now we experimentally study the convergence behavior of this algorithm under different missing patterns and missing ratios. Specifically, we run the algorithm by changing missing ratio

Table 5
Imputation RMSE ($\times 10^{-2}$) on UCI datasets when $\delta = 0.1$.

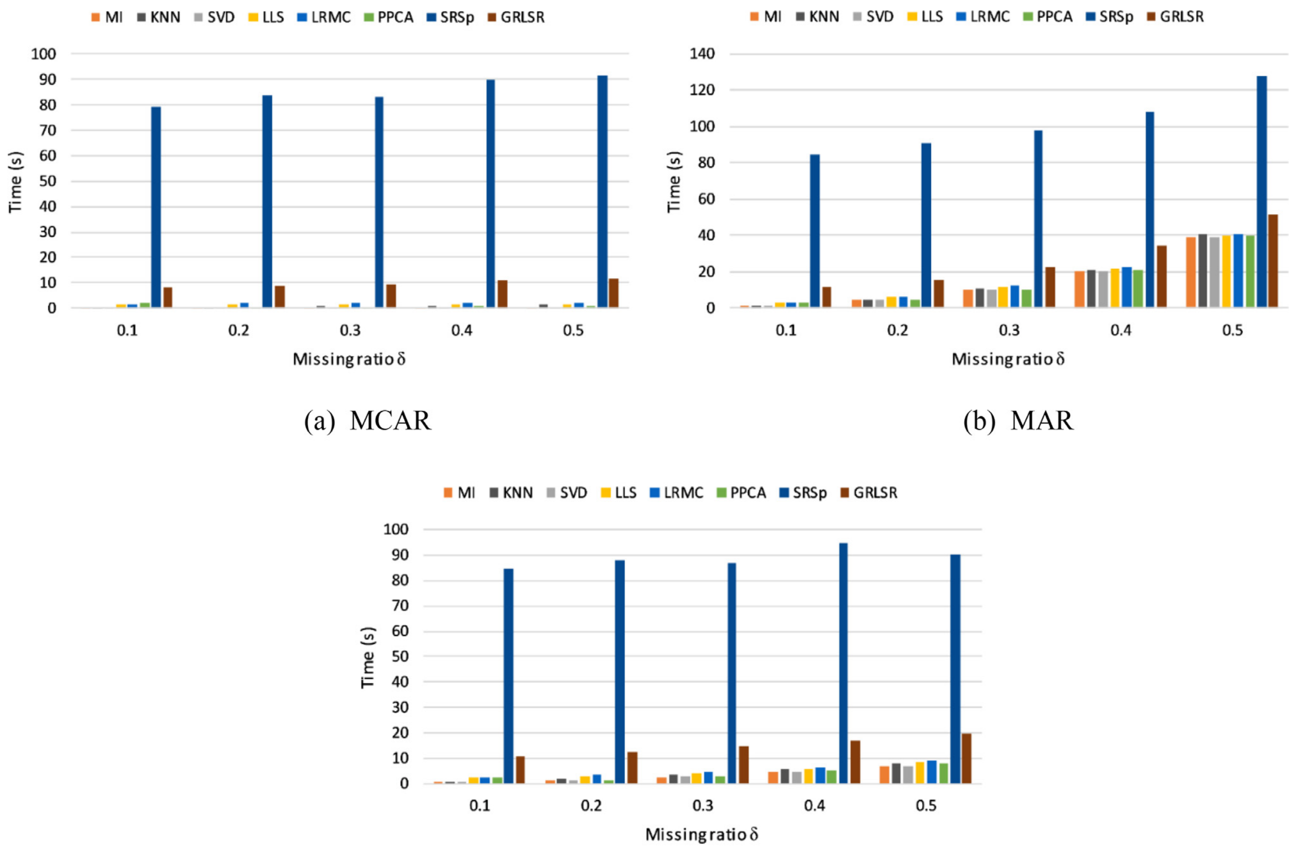
Data	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
heart	68.81 ± 3.00	65.76 ± 4.00	65.89 ± 3.14	66.19 ± 3.32	66.67 ± 1.94	**63.85 ± 3.44**	66.18 ± 3.31	64.18 ± 2.98
iono	53.62 ± 1.21	38.53 ± 1.70	40.53 ± 1.72	44.32 ± 4.56	40.56 ± 1.61	39.59 ± 1.55	43.94 ± 4.60	**37.82 ± 1.60**
liver	28.22 ± 2.89	26.84 ± 2.58	26.49 ± 2.34	27.20 ± 2.31	29.46 ± 1.64	**25.72 ± 2.36**	27.22 ± 2.38	25.81 ± 2.35
pima	31.92 ± 1.63	29.56 ± 1.74	29.11 ± 1.28	28.61 ± 1.66	33.23 ± 1.22	28.16 ± 1.43	28.59 ± 1.66	**27.55 ± 1.47**
spectf	31.37 ± 1.33	23.82 ± 1.13	20.19 ± 1.10	18.92 ± 1.16	18.54 ± 0.87	**18.45 ± 0.96**	18.91 ± 1.15	19.26 ± 0.87
votes	98.72 ± 0.44	77.10 ± 2.38	79.98 ± 1.95	77.05 ± 2.22	79.06 ± 1.36	76.63 ± 2.19	77.03 ± 2.20	**76.10 ± 2.22**

Table 6
Imputation RMSE ($\times 10^{-2}$) on UCI datasets when $\delta = 0.2$.

Data	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
heart	68.35 ± 1.80	65.93 ± 2.47	66.15 ± 1.59	65.66 ± 1.76	66.76 ± 0.94	64.62 ± 2.17	65.66 ± 1.76	**64.25 ± 1.83**
iono	53.44 ± 0.82	39.11 ± 0.83	40.84 ± 0.91	41.46 ± 1.20	40.75 ± 0.75	40.04 ± 1.05	41.15 ± 0.97	**37.91 ± 0.87**
liver	28.67 ± 1.81	27.21 ± 1.83	26.69 ± 1.65	28.73 ± 1.32	29.85 ± 1.42	**25.80 ± 1.82**	28.65 ± 1.41	25.85 ± 1.82
pima	31.87 ± 1.01	29.97 ± 1.15	29.65 ± 0.90	30.19 ± 0.68	34.21 ± 0.78	28.89 ± 1.01	30.10 ± 0.70	**28.15 ± 0.82**
spectf	31.20 ± 0.62	24.76 ± 0.68	21.39 ± 0.69	20.02 ± 0.42	19.39 ± 0.46	**19.36 ± 0.63**	20.00 ± 0.42	19.96 ± 0.34
votes	98.91 ± 0.37	77.88 ± 0.98	80.90 ± 0.90	78.09 ± 1.27	79.32 ± 0.60	77.30 ± 1.08	78.15 ± 1.17	**76.72 ± 0.97**

Table 7
Imputation RMSE ($\times 10^{-2}$) on UCI datasets when $\delta = 0.3$.

Data	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
heart	68.35 ± 1.80	65.93 ± 2.47	66.15 ± 1.59	65.66 ± 1.76	66.76 ± 0.94	64.62 ± 2.17	65.66 ± 1.76	**64.25 ± 1.83**
iono	53.44 ± 0.82	39.11 ± 0.83	40.84 ± 0.91	41.46 ± 1.20	40.75 ± 0.75	40.04 ± 1.05	41.15 ± 0.97	**37.91 ± 0.87**
liver	28.67 ± 1.81	27.21 ± 1.83	26.69 ± 1.65	28.73 ± 1.32	29.85 ± 1.42	**25.80 ± 1.82**	28.65 ± 1.41	25.85 ± 1.82
pima	31.87 ± 1.01	29.97 ± 1.15	29.65 ± 0.90	30.19 ± 0.68	34.21 ± 0.78	28.89 ± 1.01	30.10 ± 0.70	**28.15 ± 0.82**
spectf	31.20 ± 0.62	24.76 ± 0.68	21.39 ± 0.69	20.02 ± 0.42	19.39 ± 0.46	**19.36 ± 0.63**	20.00 ± 0.42	19.96 ± 0.34
votes	98.91 ± 0.37	77.88 ± 0.98	80.90 ± 0.90	78.09 ± 1.27	79.32 ± 0.60	77.30 ± 1.08	78.15 ± 1.17	**76.72 ± 0.97**



(a) MCAR

(b) MAR

(c) MIXED

Fig. 7. Illustration of the computational time of different methods.

δ in {0.1, 0.3, 0.5} and record the variation of objective function (3) under three missing patterns, i.e., MCAR, MAR, and MIXED. The resulting convergence curves are shown in Fig. 8(a)–(c), where the x-axis and y-axis denote the number of iterations and the value of objective function, respectively. From these figures, we observe that with the increasing of missing ratio δ, it needs to conduct the algorithm with more iterations before convergence. This observation is consistent with the above computational time analysis. We also notice that our algorithm converges in less than 15 iterations in most cases, demonstrating that this algorithm is able to converge very fast.

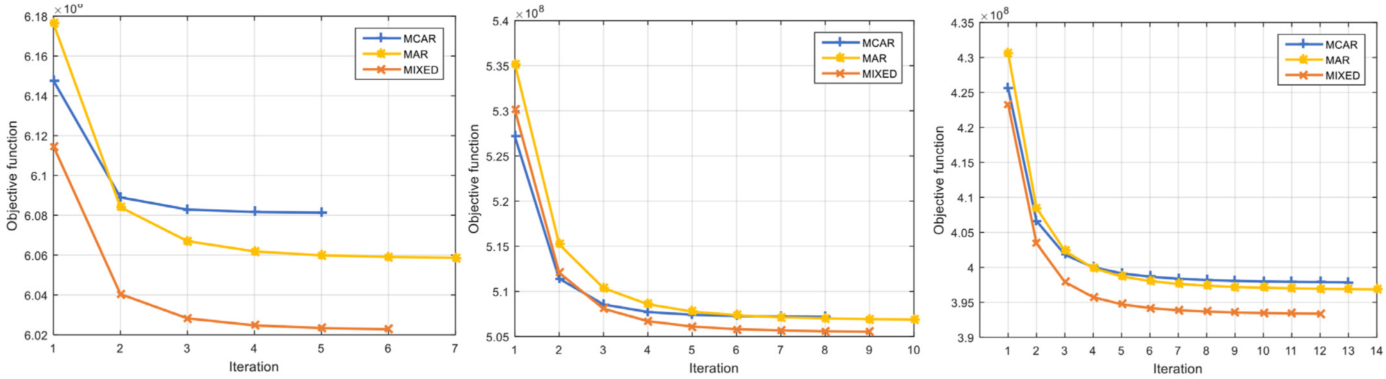
To summarize, the reason for the good performance of GRLSR is that it can incorporate the local proximity structure of data into MVs imputation problem in a natural way, thus largely favoring the accurate recovery of MV. And in our optimization algorithm, the inner two iterations (l_2-norm based least squares regression and

Table 8
Imputation RMSE ($\times 10^{-2}$) on UCI datasets when $\delta = 0.4$.

Data	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
heart	68.04 ± 1.23	66.94 ± 1.69	66.75 ± 1.67	66.83 ± 1.32	68.16 ± 0.94	66.15 ± 1.55	66.82 ± 1.35	**64.83 ± 1.34**
iono	53.51 ± 0.49	41.94 ± 0.55	42.24 ± 0.63	42.52 ± 0.86	42.31 ± 0.52	41.16 ± 0.71	42.04 ± 1.07	**38.57 ± 0.61**
liver	28.13 ± 1.32	27.33 ± 1.39	26.85 ± 1.50	37.97 ± 1.90	32.86 ± 1.18	**26.74 ± 1.42**	37.68 ± 2.14	26.98 ± 1.41
pima	32.14 ± 0.36	31.00 ± 0.37	30.83 ± 0.33	37.12 ± 0.76	37.58 ± 0.45	30.85 ± 0.28	36.78 ± 0.68	**30.03 ± 0.44**
spectf	31.24 ± 0.37	27.18 ± 0.40	23.43 ± 0.35	22.23 ± 0.34	21.27 ± 0.43	**21.14 ± 0.35**	23.09 ± 0.84	21.30 ± 0.35
votes	98.97 ± 0.42	79.04 ± 0.78	82.46 ± 0.80	80.01 ± 0.74	80.75 ± 0.48	78.28 ± 0.64	79.75 ± 0.72	**77.93 ± 0.68**

Table 9
Imputation RMSE ($\times 10^{-2}$) on UCI datasets when $\delta = 0.5$.

Data	Method							
	MI	KNN	SVD	LLS	LRMC	PPCA	SRSp	GRLSR
heart	68.19 ± 0.94	67.71 ± 1.24	67.07 ± 1.18	68.76 ± 1.57	69.26 ± 0.69	66.35 ± 1.25	68.77 ± 1.60	**65.32 ± 1.06**
iono	53.62 ± 0.39	44.52 ± 0.43	43.45 ± 0.68	43.73 ± 0.52	43.65 ± 0.45	42.30 ± 0.51	43.31 ± 0.83	**39.25 ± 0.41**
liver	28.17 ± 0.90	27.62 ± 0.89	**27.09 ± 0.98**	44.66 ± 1.46	35.15 ± 0.86	28.10 ± 0.92	44.52 ± 1.58	27.70 ± 0.83
pima	32.16 ± 0.19	31.40 ± 0.24	**31.07 ± 0.25**	42.53 ± 0.77	39.75 ± 0.56	31.82 ± 0.18	42.22 ± 0.83	31.45 ± 0.31
spectf	31.38 ± 0.25	28.51 ± 0.28	24.31 ± 0.34	23.61 ± 0.29	22.59 ± 0.30	22.81 ± 0.27	25.11 ± 1.18	22.28 ± 0.29
votes	98.92 ± 0.28	80.68 ± 0.82	83.54 ± 0.68	82.46 ± 0.87	81.78 ± 0.42	79.61 ± 0.66	81.95 ± 0.96	**78.90 ± 0.59**



(a) $\delta = 0.1$ (b) $\delta = 0.3$ (c) $\delta = 0.5$

Fig. 8. Illustration of the convergence of algorithm.

gradient descent) both run very fast. So, the proposed algorithm is also efficient.

5. Conclusions

This paper has proposed an accurate yet efficient approach for MVs imputation. In comparison with previous MVs imputation approaches, the proposed method incorporates locality constraint and graph-based regularization into the framework of sample self-representation, thus taking better advantages of local proximity structure of data. We also provide an effective iterative algorithm to solve the proposed model, and the convergence and computational complexity are also analyzed. We apply the proposed method to a real-world traffic sensor data and several UCI benchmark datasets. The experimental results demonstrate the effectiveness of the proposed method.

In this work, we mainly attempt to improve the recovery performance and implement the algorithm in a batch manner. It needs to memorize all samples especially when estimating missing values using projected gradient descent method. In addition to recovery performance, the scalability of algorithm is an interesting yet nontrivial issue for the practical uses. In the case of large scale data, it is prohibited to load all the data into memory during processing. There are some possible solutions to increase the scalability of this algorithm. For example, the representation of each sample which is solved separately can be easily parallelized. On the other hand, in the case of massive data, it is possible to select a few informative samples instead of using the whole dataset as dictionary because of information redundancy. However, how to design proper criterion to choose those informative samples without a sacrifice on recovery performance deserves dedicated research.

In addition, this work mainly focuses on MAR and MCAR missing pattern frequently arising in real-life. However, missing not at random (MNAR) is also a type of important missing pattern. For MNAR data, the underlying mechanism for observed data and for missing data might be different, and thus probably incurs biased estimation when applying most methods without accounting for such discrepancy. For example, in E-commerce recommendation [47,48], some users may intentionally hide (or give misleading) preferences/ratings towards some products, due to some unknown reasons. As a result, accurate recovery of MVs from MNAR data is more difficult than from either MCAR or MAR data. How to extend the presented algorithm for coping with MNAR data also deserves deep research in the future.

Acknowledgment

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions. This work was partially supported by the National Natural Science Foundation of China (Grant nos. 61773184, 61772254, 61601203, U1564201, 61401214, U1664258), the National Key Research and Development Program of China (2017YFB0102603), Six talent peaks project of Jiangsu Province (Grant no. 2017-JXQC-007), Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, NJUPT (BDSIP1802), Fujian Provincial Key Laboratory of Information Processing and Intelligent Control (Minjiang University) (MJUKF201724), Key Research and Development Program of Jiangsu Province (BE2016149), Natural Science Foundation of Jiangsu Province (BK2017153, BK20161516), Key Project for the Development of Strategic Emerging Industries of Jiangsu Province (2016-1094, 2015-1084), and the Talent Foundation of Jiangsu University, China (No. 14JDG066).

References

[1] F. Schimbinschi, X.V. Nguyen, J. Bailey, C. Leckie, H. Vu, R. Kotagiri, Traffic forecasting in complex urban networks: leveraging big data and machine learning, in: Proceedings of the 2015 IEEE International Conference on Big Data (Big Data), 2015, pp. 1019–1024.

[2] X. Chen, Z. Wei, X. Liu, Y. Cai, Z. Li, F. Zhao, Spatiotemporal variable and parameter selection using sparse hybrid genetic algorithm for traffic flow forecasting, Int. J. Distrib. Sens. Netw. 13 (2017) 1550147717713376.

[3] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (1995) 273–297.

[4] X. Chen, J. Yang, Q. Ye, J. Liang, Recursive projection twin support vector machine via within-class variance minimization, Pattern Recognit. 44 (10–11) (2011) 2643–2655.

[5] J. Luengo, S. García, F. Herrera, On the choice of the best imputation methods for missing values considering three groups of classification methods, Knowl. Inf. Syst. 32 (2012) 77–108.

[6] X. Chen, Z. Wei, Z. Li, J. Liang, Y. Cai, B. Zhang, Ensemble correlation-based low-rank matrix completion with applications to traffic data imputation, Knowl.-Based Syst. 132 (2017) 249–262.

[7] X. Yi, Y. Zheng, J. Zhang, T. Li, ST-MVL: Filling Missing Values in Geo-Sensory Time Series Data, in: Proceedings of the 25th International Joint Conference on Artificial Intelligence, 2016, pp. 2704–2710.

[8] H. Kim, G.H. Golub, H. Park, Missing value estimation for DNA microarray gene expression data: local least squares imputation, Bioinformatics 21 (2005) 187–198.

[9] G. Chang, Y. Zhang, D. Yao, Missing data imputation for traffic flow based on improved local least squares, Tsinghua Sci. Technol. 17 (2012) 304–309.

[10] L. Qu, L. Li, Y. Zhang, J. Hu, PPCA-based missing data imputation for traffic flow volume: a systematical approach, IEEE Trans. Intell. Transp. Syst. 10 (2009) 512–522.

[11] L. Qu, Y. Zhang, J. Hu, L. Jia, L. Li, A BPCA based missing value imputing method for traffic flow volume data, in: Proceedings of IEEE Intelligent Vehicles Symposium, 2008, pp. 985–990.

[12] E.J. Candès, B. Recht, Exact matrix completion via convex optimization, Found. Comput. Math. 9 (2009) 717–772.

[13] J.-F. Cai, E.J. Candès, Z. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. 20 (2010) 1956–1982.

[14] J. Fan, T. Chow, Matrix completion by least-square, low-rank, and sparse self--representations, Pattern Recognit. 71 (2017) 290–305.

[15] M.T. Asif, N. Mitrovic, J. Dauwels, P. Jaillet, Matrix and tensor based methods for missing data estimation in large traffic networks, IEEE Trans. Intell. Transp. Syst. 17 (7) (2016) 1816–1825.

[16] C. Li, L. Lin, W. Zuo, W. Wang, J. Tang, An approach to streaming video segmentation with sub-optimal low-rank decomposition, IEEE Trans. Image Process. 25 (5) (2016) 1947–1960.

[17] C. Li, X. Wu, Z. Bao, J. Tang, "ReGLe: spatially regularized graph learning for visual tracking," *Proceedings of* ACM Multimedia Conference 2017: 252–260

[18] C. Li, X. Sun, X. Wang, L. Zhang, J. Tang, Grayscale-thermal object tracking via multitask Laplacian sparse representation, IEEE Trans. Syst. Man Cybern.: Syst. 47 (4) (2017) 673–681.

[19] C. Li, H. Cheng, S. Hu, X. Liu, J. Tang, L. Lin, Learning collaborative sparse representation for grayscale-thermal tracking, IEEE Trans. Image Process. 25 (12) (2016) 5743–5756.

[20] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323.

[21] Z. Nan, Y. Jian, K nearest neighbor based local sparse representation classifier, in: Proceedings of 2010 Chinese Conference on Pattern Recognition (CCPR), 2010.

[22] X. He, P. Niyogi, Locality preserving projections, in: *Proceedings of Advance in Neural Information Processing Systems* 16, Vancouver, Canada, 2003.

[23] X. Chen, J. Yang, J. Liang, Optimal locality regularized least squares support vector machine via alternating optimization, Neural Process. Lett. 33 (3) (2011) 301–315.

[24] X. Gan, A.W.-C. Liew, H. Yan, Microarray missing data imputation based on a set theoretic framework and biological knowledge, Nucleic Acids Res. 34 (2006) 1608–1619.

[25] F. Shi, D. Zhang, J. Chen, H.R. Karimi, Missing value estimation for microarray data by Bayesian principal component analysis and iterative local least squares, Math. Prob. Eng. 2013 (2013) Article ID 162938, 5 pages.

[26] Z. Yu, T. Li, S.-J. Horng, Y. Pan, H. Wang, Y. Jing, An iterative locally auto-weighted least squares method for microarray missing value estimation, IEEE Trans. Nanobiosci. 16 (2017) 21–33.

[27] H. Tan, G. Feng, J. Feng, W. Wang, Y.-J. Zhang, F. Li, A tensor-based method for missing traffic data completion, Transp. Res. Part C: Emerg. Technol. 28 (2013) 15–27.

[28] M.E. Tipping, C.M. Bishop, Probabilistic principal component analysis, J. R. Stat. Soc.: Ser. B (Stat. Methodol.) 61 (1999) 611–622.

[29] Z. Cai, M. Heydari, G. Lin, Iterated local least squares microarray missing value imputation, J. Bioinform. Comput. Biol. 4 (2006) 935–957.

[30] F. Honghai, C. Guoshun, Y. Cheng, Y. Bingru, C. Yumei, A SVM regression based approach to filling in missing values, in: Proceedings of International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, 2005, pp. 581–587.

[31] Y. Zhang, Y. Liu, Data imputation using least squares support vector machines in urban arterial streets, IEEE Signal Process Lett. 16 (2009) 414–417.

[32] E.-L. Silva-Ramírez, R. Pino-Mejías, M. López-Coello, M.-D. Cubiles-de-la-Vega, Missing value imputation on missing completely at random data using multilayer perceptrons, Neural Netw. 24 (2011) 121–129.

[33] M.T. Asif, N. Mitrovic, L. Garg, J. Dauwels, P. Jaillet, Low-dimensional models for missing data imputation in road networks, in: Proceedings of 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, 2013, pp. 3527–3531.

[34] B. Ran, H. Tan, J. Feng, Y. Liu, W. Wang, Traffic speed data imputation method based on tensor completion, Comput. Intell. Neurosci. 2015 (2015) Article ID 364089, 9 pages.

[35] L. Chen, G. Yang, Z. Chen, F. Xiao, J. Shi, Correlation consistency constrained matrix completion for web service tag refinement, Neural Comput. Appl. 26 (2015) 101–110.

[36] P. Zhu, L. Zhang, W. Zuo, X. Feng, Q. Hu, A self-representation induced classifier, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 2016, pp. 2442–2448.

[37] J. Yang, L. Zhang, Y. Xu, J.-y. Yang, Beyond sparsity: the role of L 1-optimizer in pattern classification, Pattern Recognit. 45 (2012) 1104–1118.

[38] C.-P. Wei, Y.-W. Chao, Y.-R. Yeh, Y.-C.F. Wang, Locality-sensitive dictionary learning for sparse representation based classification, Pattern Recognit. 46 (2013) 1277–1287.

[39] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 3360–3367.

[40] D. Cai, X. He, K. Zhou, J. Han, H. Bao, Locality Sensitive Discriminant Analysis, in: Proceedings of the 20th International Joint Conference on Artificial Intelligence, 2007, pp. 708–713.

[41] F.R. Chung, Spectral Graph Theory, American Mathematical Society, 1997.

[42] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[43] X. Chen, J. Yang, L. Chen, An improved robust and sparse twin support vector regression via linear programming, Soft Comput. 18 (2014) 2335–2348.

[44] W. Rudin, Principles of Mathematical Analysis, 3, McGraw-hill, New York, 1964.

[45] J.L. Schafer, Analysis of Incomplete Multivariate Data, CRC Press, 1997.

[46] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends Mach. Learn. 3 (2011) 1–122.

[47] X. Ning, G. Karypis, Slim: sparse linear methods for top-n recommender systems, in: Proceedings of the 2011 IEEE 11th International Conference on Data Mining (ICDM), IEEE, 2011, pp. 497–506.

[48] A.N. Nikolakopoulos, M.A. Kouneli, J.D. Garofalakis, Hierarchical itemspace rank: exploiting hierarchy to alleviate sparsity in ranking-based recommendation, Neurocomputing 163 (2015) 126–136.

Xiaobo Chen received the Ph.D. degree in pattern recognition and intelligent systems from Nanjing University of Science & Technology (NJUST), China, in 2013. From March to August 2011, he served as a research assistant at the Hong Kong Polytechnic University (PolyU), Hong Kong. From April 2015 to August 2017, he served as a postdoctoral research associate at the University of North Carolina at Chapel Hill (UNC-CH), USA. He is currently an associate professor at automotive engineering research institute, Jiangsu University, China. His major research interests concentrate on pattern recognition and its applications. He has published more than 50 papers in numerous international journals and conference proceedings.

Yingfeng Cai received the B.S., M.S. and Ph.D. degrees all from the School of Instrument Science and Engineering, Southeast University, Nanjing, China, respectively. In 2013, she joined the Automotive Engineering Research Institute in Jiangsu University as an assistant professor. Her research interests include computer vision, intelligent transportation systems and intelligent automobiles.

Qiaolin Ye received the B.S. degree in computer science from the Nanjing Institute of Technology, Nanjing, China, in 2007, the M.S. degree in computer science and technology from Nanjing Forestry University, Nanjing, in 2009, and the Ph.D. degree in pattern recognition and intelligence system from the Nanjing University of Science and Technology, Nanjing, in 2013. He is currently an Associate Professor of the Computer Science Department with Nanjing Forestry University. He has authored over 50 scientific papers. Some of them are published in the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. His current research interests include machine learning, data mining, and pattern recognition.

Lei Chen received his Ph.D. Degree in communication and information system from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2014. He is currently an associate Professor of Nanjing University of Posts and Telecommunications, Nanjing, China. His research interests include pattern recognition, machine learning and neural computing.

Zuoyong Li received the B.S. and M.S. degrees in Computer Science and Technology from Fuzhou University, Fuzhou, China, in 2002 and 2006, respectively. He received the Ph.D. degree from the School of Computer Science and Technology at Nanjing University of Science and Technology, Nanjing (NUST), China, in 2010. He is currently a professor in Department of Computer Science of Minjiang University, Fuzhou, China. He has published over 50 papers in international/national journals. His current research interests include image processing, pattern recognition and machine learning.