Dynamic Imputation Methodology for Multi-source Streaming Mobility Data

Michiel Dhont[1,2(✉)], Elena Tsiporkova[1], and Nicolás González-Deleito[1]

[1] EluciDATA Lab of Sirris, BluePoint Brussels, Bd A. Reyerslaan 80, 1030 Brussels, Belgium
{michiel.dhont,elena.tsiporkova,nicolas.gonzalez}@sirris.be
[2] Department of Electronics and Information Processing (ETRO), Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussels, Belgium

Abstract. The road network is becoming increasingly equipped with a multitude of sensors, monitoring a wide range of operating and contextual parameters. The availability of real-time sensor data enables the realisation of diverse data-driven applications, e.g., anomaly detection, identification of insightful patterns, monitoring the evolution of relevant trends in time and delivery of actionable decision support. However, such streaming data might contain vast amounts of missing values depending on the application. This makes it very challenging, if not impossible, to fully exploit the potential of data analysis and machine learning for these data sources, and in particular real-time analysis is not feasible. We propose in this paper an imputation methodology dedicated to multi-source streaming data, with a focus on the mobility domain. The proposed approach is based on spatio-temporal profiling of the streaming behaviour derived from historical data via non-negative matrix factorisation. The profiling method takes advantage of an adaptive segmentation strategy splitting the data into rolling time windows (chunks) allowing to use the limited non-missing data as optimally as possible. The identified profiles allow to devise a dynamic and scalable imputation strategy, which is able to reliably estimate incoming missing values in streaming data as soon as they arrive.

Keywords: Data imputation · Matrix factorisation · Streaming data · Vehicle counts

1 Introduction

There is an increasing trend of connecting devices (e.g., smart watches, smart household appliances and ANPR cameras) and industrial machinery (e.g., compressors, fleet tracking and melting furnaces) to the internet [20]. Since these

This research was subsidised through the project MISTic by the Brussels-Capital Region - Innoviris and received funding from the Flemish Government (AI Research Program).

Y. Bie et al. (Eds.): KES-STS 2022, SIST 304, pp. 184–198, 2022.
https://doi.org/10.1007/978-981-19-2813-0_19

assets are connected over a network, their data can be inspected in real-time. This real-time aspect opens a wide range of opportunities as it allows for early and continuous monitoring of trends and anomalies. By deriving an accurate view of the latest state of affairs at all time, real-time data-driven decision support applications can be developed.

Unfortunately, exploiting directly streaming data on the fly is not straightforward since it is often suffering from poor quality mostly due to incompleteness (e.g., in the mobility domain [16,25]). In practice, data capturing implementations are often rapid/experimental, i.e., sensors are quickly deployed to gather data. Moreover, these deployments can be continuously expanding in terms of additional sensors which are installed in order to capture supplementary information. Last, the devices may often be located in difficult to access or widely scattered locations (e.g., inductive loops under the asphalt of a busy road), making it difficult to do good maintenance (e.g., replace broken sensors). Due to these reasons, it is almost inevitable to have some flaws in the data capturing process because of technical failures. The latter may result in various types of missing data values: randomly scattered missing values, a single sensor missing over a longer period, and relatively short moments in time when no values are available at all.

Particularly in the case of streaming data, it is often important to be able to impute missing values as soon as a limited amount of values are available. However, due to the highly dynamic nature of the data and the frequent occurrences of long sequences of subsequent missing values, such an imputation is very challenging. For this reason, imputation approaches need to be able to learn from the past, capturing prototypical behaviour via adequate profiling mechanisms. Although there are many imputation algorithms proposed in the literature, some interesting opportunities have not yet been explored. Especially in the context of continuously incoming multi-source streaming (mobility) data, there is a lot of room for improvement.

Some imputation algorithms are developed with the specific goal to work on a continuous stream of data, while other algorithms try to capture the spatial and/or temporal patterns on a static data set. Although some of these imputation algorithms yield good results, a hybrid combination of both would be more relevant in practice since data is often both continuously incoming and exhibits multi-source correlations. In this paper, we propose a novel imputation methodology for multi-source data, capable of handling continuous streaming data and validated on real-life vehicle counting data. This approach is partially inspired by the spatio-temporal fingerprinting approach which was proposed in [6] and was originally used for the purpose of performance profiling. In this paper, we exploit cleverly its characterisation potential to dynamically detect latent spatial and temporal structures in multi-source data for the purpose of missing data imputation.

The remaining of this paper is organised as follows: Section 2 focuses on existing related approaches in the literature, while Sect. 3 outlines in detail the proposed novel imputation methodology. Section 4 covers the obtained validation

results of a use case on mobility data. Finally, in Sect. 5, some concluding remarks are given.

2 Related Work

In this section, the state of the art in recent imputation strategies and approaches is discussed. First, in Sect. 2.1, existing imputation techniques for time series are briefly summarised. Next, in Sect. 2.2, the most relevant latest developments in streaming data imputation approaches are discussed.

2.1 Time Series Data Imputation

Existing imputation approaches for time series data can be divided into three different classes: interpolation, statistical learning and prediction [15]. The first class, interpolation, is the most straightforward imputation method, as it approximates the missing data by fitting a curve on top of the available data. The curve aims to define the sequence of data points by a linear or polynomial function, allowing to estimate unknown values [24]. Secondly, statistical learning-based imputation approaches aim to encapsulate statistical features of the data into a model. The latter could be achieved for example by applying the k-nearest neighbours approach [4,15,17], where estimations are made based on the k most similar situations, or a principal component analysis [18,22], where an expectation-maximisation algorithm is used to estimate values of missing data points. Finally, prediction-based imputation approaches aim to capture the temporal relationship within time series. These imputation algorithms are developed to find long term and short term relations, giving an idea about what value to expect next. The autoregressive integrated moving average method [13,28] and Bayesian networks [9,26] are two methods that belong to the category of prediction-based imputation approaches.

More recently, factorisation techniques are used to impute missing values in matrix-like data sets. Completion of matrix-like data has shown to be relevant in many applications, such as image inpainting [14] and recommender systems [21]. For example, singular value decomposition can be used for matrix completion [7]. Bao et al. [2] illustrate how this approach can be applied on a multi-variate time series data set, where each row represents a time series for a different sensor and each column captures exactly one value for each sensor at a particular moment in time. Apart from the good imputation results, the non-parametric approach allows for reliable data imputation without user intervention. Note that existing factorisation imputation techniques are only designed to fill in gaps within a matrix. In case of streaming data, new unseen columns (i.e., moments in time) are continuously added to the data matrix. Consequently, the full factorisation must be recalculated each time new columns with missing values are added to the back of the matrix. For high-frequency multi-variate data streams, such an approach would be highly inefficient and therefore unusable in real-time. Section 2.2 gives an overview on the latest developments on streaming data imputation.

2.2 Streaming Data Imputation

The increasing availability of real-time sensor data, opens a wide range of new opportunities to instantaneously analyse the data and provide data-driven decision-making. However, in the case of low-quality data, this means that data imputation strategies also need to be adapted to run efficiently on new chunks of incoming data. Coupled to this, data imputation algorithms should be able to define a level of certainty in order to express how reliable the imputation results are.

Depending on the use case, different streaming imputation strategies are appropriate. In situations where computational power is limited (think of data imputation on the edge) or a short latency is crucial, it is good practice to construct a fixed imputation model on beforehand which allows for efficient imputation on continuously incoming data. Following this approach, Fountas and Kolomvatsos propose an ensemble correlation approach to identify the pairwise similarity between a number of different sensors (i.e., multi-variate time series). Missing values are imputed based on the values of the top-k correlated sensors, weighted by their correlation [8]. In [6], a similar approach is proposed, imputing missing values in an incremental way. First, missing data is imputed based on the top-k moments in time with the most similar non-missing values. Next, the remaining gaps are filled by use of the most similar larger periods of time (e.g., days).

If the data is more complex (e.g., new unseen or diverging data patterns may arise over time) or there is initially only a limited amount of historical data available, a continuous learning imputation approach is expected to be more appropriate. To give an example, in [19] a single factorisation to identify temporal features for historical missing data imputation is exploited. Then, an incremental learning scheme based on an autoregressive model is proposed, allowing for response forecasting based on the temporal features. In the study of Halder et al. [10], some problems with imbalanced data during data stream imputation are considered. To overcome these problems, an adaptive imputation approach is proposed which includes an oversampling method per chunk of streaming data and a fuzzy decomposition method to determine the interrelationship among instances. Despite the good results on imbalanced data sets, this approach has some performance issues in the case of noisy data, which is rather crucial in a real-wold context. Furthermore, none of these methods are able to store and recognise historically occurred relations between sensors. For instance, imagine a multivariate time series that counts the number of vehicles on a number of streets close to a charging bridge over a canal. Whether this bridge has opened (and influenced the traffic flow) during a gap of missing data, is impossible to know (i.e., impute) based on only the time aspect of one sensor. In such a case, the imputation method should be able to dynamically recognise the situation based on the other sensors at that moment in time. The novel imputation method we propose in this work is able to deal with such situations and also tackles most of the other shortcomings of the related works discussed above. More specific, our approach can efficiently impute multi-variate streaming

data, while still considering both temporal and spatial relations using a dynamic profiling methodology.

3 Materials and Methods

In this section the building blocks of our novel dynamic profiling and imputation methodology are discussed. Section 3.1 outlines the concept of matrix factorisation, while Sect. 3.2 is devoted to the description of the spatio-temporal profiling. Next, Sect. 3.3 explains how to exploit the profiles to impute missing values. Section 3.4 indicates how to deploy this approach on real-time streaming data. Finally, Sect. 3.5 provides a description of the test data set and the used computer code.

3.1 Matrix Factorisation

Matrix factorisation is a discipline of linear algebra allowing to decompose a matrix into a product of matrices. One popular example of this approach is the **singular value decomposition** (SVD). Consider a matrix $\boldsymbol{X} \in \mathbb{C}^{M \times N}$. By the use of SVD, $\boldsymbol{X}$ can be factorised into $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, with $\boldsymbol{U}$ a unitary matrix $\in \mathbb{C}^{M \times M}$, $\boldsymbol{\Sigma}$ a rectangular diagonal matrix $\in \mathbb{R}_+^{M \times N}$ and $\boldsymbol{V}$ a unitary matrix $\in \mathbb{C}^{N \times N}$. SVD has many applications, such as solving homogeneous linear equations (e.g., [1]), total least squares minimisation (e.g., [27]) and low-rank matrix approximations (e.g., [23]).

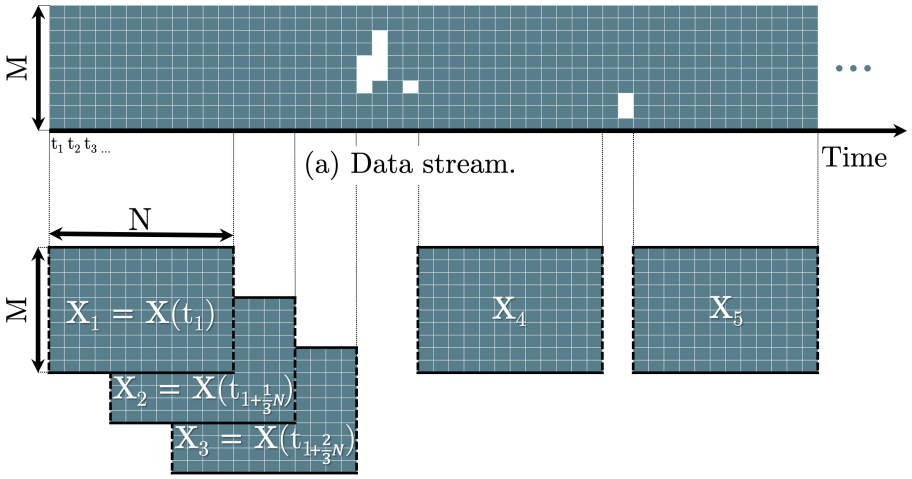
Another factorisation approach is **non-negative matrix factorisation** (NMF). As the name reveals, this approach is designed to work with matrices containing only positive values. Consider a matrix $\boldsymbol{X} \in \mathbb{R}_+^{M \times N}$. NMF is able to approximate $\boldsymbol{X}$ by a product of two factors $\boldsymbol{W}\boldsymbol{S}$, where $\boldsymbol{W} \in \mathbb{R}_+^{M \times R}$, $\boldsymbol{S} \in \mathbb{R}_+^{R \times N}$ and $R < \min(M, N)$. The smaller the value of parameter R, the greater the dimensionality reduction performed, at the expense of the reconstruction error for $\boldsymbol{X}$. In contrast to SVD, NMF is an approximation for which no exact solution exists. There are multiple heuristic algorithms developed to find $\boldsymbol{W}$ and $\boldsymbol{S}$ [11]. In our approach, the fast hierarchical alternating least squares (Fast HALS) algorithm is used [5]. Compared to SVD, the added value of NMF is the constraint of having only positive values in matrices $\boldsymbol{W}$ and $\boldsymbol{S}$. Depending on the nature of the data from the original matrix, it often is a more natural process to decompose it into positive factors [12]. To reconstruct the original matrix $\boldsymbol{X}$, no element-wise subtractions need to be performed.

3.2 Spatio-Temporal Profiling

Spatio-temporal profiling is an essential prerequisite for our imputation strategy. It allows to extract latent spatial and temporal patterns from historical data, which are subsequently used by the imputation algorithm. Consider a matrix representing data from a multi-variate time series as visualised in Fig. 1(a). In such a matrix, each row represents one of the M different parameters (e.g., sensor

values for different locations). Each column represents a consecutive timestamp (e.g., one value per minute). Considering we are working with streaming data, the time dimension is infinite.

Adaptive Dynamic Segmentation. To obtain resilient profiles, gaps of missing data are avoided during the profiling procedure. The first step in that procedure is to extract chunks of data with a fixed time window of N timestamps. The width N of each time window should be large enough so that meaningful temporal patterns can be identified, but small enough so that enough chunks of data without missing values can be found. The fulfilment of these requirements are dependent on the use case of interest. The data chunk extraction happens by chronologically looping over all timestamps $(t_1, t_2 \cdots, t_T)$, with t_1 the oldest timestamp and t_T the most recent timestamp. A chunk is only selected if it contains no missing values. Each time a chunk is selected, a number of timestamps is skipped before proceeding with the selection of the next chunk in order to avoid excessive overlap between chunks. As a rule of thumb, we do a forward jump of $\frac{1}{3}N$ timestamps in order to have a maximum overlap of $\frac{2}{3}N$ timestamps between chunks. In Fig. 1(b), the selected chunks from Fig. 1(a) are visualised. The overlap between the different chunks is essential for being able to capture transitions between different patterns in time.



(a) Data stream.

(b) Selected data chunks.

Fig. 1. Selection of data chucks for profiling.

In the second step, all selected data chunks are stacked on top of each other:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \\ \vdots \\ \boldsymbol{X}_L \end{bmatrix} \in \mathbb{R}_+^{(ML)\times N}, \tag{1}$$

where $\boldsymbol{X}_i$ represents the ith selected chunk and L the total number of selected chunks.

Temporal Profiles Extraction. Temporal profiles are extracted from the stacked matrix $\boldsymbol{X}$ by applying a decomposition method. In this paper, NMF is used due to the non-negative properties that are suitable for the use case data and to facilitate interpretation. In this way, matrices $\boldsymbol{W}$ and $\boldsymbol{S}$ are obtained as shown in Eq. (2). Conceptually, each row of matrix $\boldsymbol{S}$ can be interpreted as a temporal profile while matrix $\boldsymbol{W}$ represents the weights, which can be used to reconstruct $\boldsymbol{X}$ thanks to the temporal building blocks from $\boldsymbol{S}$.

$$\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{S}, \tag{2}$$

$$\boldsymbol{X} \approx \begin{bmatrix} \boldsymbol{W}_1 \\ \boldsymbol{W}_2 \\ \vdots \\ \boldsymbol{W}_L \end{bmatrix} \boldsymbol{S}. \tag{3}$$

These matrices are as follows: $\boldsymbol{W} \in \mathbb{R}_+^{(ML)\times R}$ and can be evenly split into L sub-matrices $\boldsymbol{W}_i \in \mathbb{R}_+^{M\times R}$, with $i = 1, 2, \ldots, L$ (see Eq. (3)), $\boldsymbol{S} \in \mathbb{R}_+^{R\times N}$, and $R \in \mathbb{N}_+$ being a hyperparameter representing the amount of temporal profiles, as explained in Sect. 3.1. Moreover, each chunk $\boldsymbol{X}_i$ can be approximated by a weighted sum of the temporal profiles as shown in Equation (4).

$$\boldsymbol{X}_i \approx \boldsymbol{W}_i\boldsymbol{S}, \quad \text{with } 1 \leq i \leq L. \tag{4}$$

Alternatively, SVD or any other decomposition technique could be used depending on the properties of the use case (e.g., if values can be negative). Note that in the case of SVD three matrices are obtained (see Sect. 3.1). However, in this situation, matrices $\boldsymbol{U}$ and $\boldsymbol{\Sigma}$ should be multiplied to replace the weight matrix $\boldsymbol{W}$, while $\boldsymbol{V}^T$ can directly be used as the temporal profile matrix $\boldsymbol{S}$.

Spatial Profiles Extraction. The decomposition of matrix $\boldsymbol{X}$ above via NMF resulted in latent temporal profiles and corresponding weights. The weight matrix $\boldsymbol{W}$ is of interest for further decomposition since it contains useful relational information between the different data sources. In situations where each parameter represents a sensor at a different location, this can be interpreted as the spatial relationship. To extract those relations, each individual weight matrix $\boldsymbol{W}_i$ is first transposed. Then, a modified weight matrix $\boldsymbol{W}'$ is constructed that

vertically stacks all individual transposed matrices:

$$\boldsymbol{W'} = \begin{bmatrix} \boldsymbol{W}_1^T \\ \boldsymbol{W}_2^T \\ \vdots \\ \boldsymbol{W}_L^T \end{bmatrix} \in \mathbb{R}_+^{(RL)\times M}. \tag{5}$$

Next, we approximate this modified weight matrix $\boldsymbol{W'}$ (by using a suitable factorisation approach) as the product of two matrices $\boldsymbol{V}$ and $\boldsymbol{U}$, as shown in Eq. (6). Assuming we again use NMF, both matrices will be non-negative. Similarly as above, the rows of the resulting matrix $\boldsymbol{U}$ can be interpreted as a set of prototypical spatial profiles, which can be used to reconstruct $\boldsymbol{W'}$ by the weights of matrix $\boldsymbol{V}$.

$$\boldsymbol{W'} \approx \boldsymbol{VU}, \tag{6}$$

$$\boldsymbol{W'} \approx \begin{bmatrix} \boldsymbol{V}_1 \\ \boldsymbol{V}_2 \\ \vdots \\ \boldsymbol{V}_L \end{bmatrix} \boldsymbol{U}, \tag{7}$$

with $\boldsymbol{V} \in \mathbb{R}_+^{(RL)\times Q}$, $\boldsymbol{V}$ can be evenly split into L sub-matrices $\boldsymbol{V}_i \in \mathbb{R}_+^{R\times Q}$, with $i = 1, 2, \ldots, L$, $\boldsymbol{U} \in \mathbb{R}_+^{Q\times M}$, and $Q \in \mathbb{N}_+$ being a hyperparameter representing the amount of spatial profiles. Similarly, Eq. (7) can be split as follows:

$$\boldsymbol{W}_i^T \approx \boldsymbol{V}_i\boldsymbol{U}, \quad \text{with } 1 \leq i \leq L. \tag{8}$$

3.3 Estimation of Missing Values

Spatial and temporal profiles, as extracted in Sect. 3.2, contain very valuable information which can be used to estimate missing values. To do that, we combine Eq. (4) and (8) as follows:

$$\boldsymbol{X}_i \approx (\boldsymbol{V}_i\boldsymbol{U})^T\boldsymbol{S}. \tag{9}$$

In this equation, any chunk $\boldsymbol{X}_i$ is expressed as the weighted combination of the temporal and spatial profiles using the weight matrix $\boldsymbol{V}_i$. Since the temporal and spatial profiles contain the latent building blocks for any period in time, this equation is assumed to also hold for any chunk with missing data ($\boldsymbol{X}_{missing}$). In that case, some values of $\boldsymbol{X}_{missing}$, as well as $\boldsymbol{V}_{missing}$, are unknown. Consequently, these unknown values can be heuristically found as a continuous optimisation problem. Technically, this can be achieved by minimising the squared error between values from the resulting matrices from the left and right sides of Eq. (9), as shown in Eq. (10):

$$
\begin{aligned}
\min_{\chi, \boldsymbol{V}_{missing}} \quad : \quad & \left(\boldsymbol{X}_{missing[1,1]} - \widehat{\boldsymbol{X}}_{[1,1]}\right)^2 + \cdots + (\boldsymbol{X}_{missing[1,N]} - \widehat{\boldsymbol{X}}_{[1,N]})^2 \\
& + \left(\boldsymbol{X}_{missing[2,1]} - \widehat{\boldsymbol{X}}_{[2,1]}\right)^2 + \cdots + (\boldsymbol{X}_{missing[2,N]} - \widehat{\boldsymbol{X}}_{[2,N]})^2 \\
& \vdots \\
& + \left(\boldsymbol{X}_{missing[M,1]} - \widehat{\boldsymbol{X}}_{[M,1]}\right)^2 + \cdots + (\boldsymbol{X}_{missing[M,N]} - \widehat{\boldsymbol{X}}_{[M,N]})^2 \\
\text{s.t.} : \quad & \widehat{\boldsymbol{X}} = (\boldsymbol{V}_{missing}\boldsymbol{U})^T \boldsymbol{S} \\
& \chi \coloneqq \{\, \boldsymbol{X}_{missing[i,j]} \mid [i,j] \; is \; missing \; in \; \boldsymbol{X}_{missing} \,\} \\
& x \geq 0 \qquad\qquad for \; x \in \chi \\
& \boldsymbol{V}_{missing[i,j]} \geq 0 \quad for \; 1 \leq i \leq Q \;\; and \; 1 \leq j \leq R,
\end{aligned}
\tag{10}
$$

with indices between squared brackets representing the coordinates of one value within a matrix, e.g., $\boldsymbol{X}_{missing[i,j]}$ being the value in matrix $\boldsymbol{X}_{missing}$ at row i and column j.

3.4 Imputation Strategy

The design of our novel imputation approach allows for data imputation on both historical and streaming data. The approach is focused on mobility data due to its strong spatial and temporal dependencies. However, it can be used in other domains that exhibit such strong dependencies. The imputation workflow's steps are as follows:

1. **Composition of training data repository.** Starting from historical data covering a sufficiently long time period allowing to capture all possible temporal and spatial patterns, a representative training data repository of only complete data chunks is composed following the segmentation approach in Sect. 3.2.
2. **Extraction of spatio-temporal profiles.** Following the two-step process described in Sect. 3.2, prototypical spatio-temporal profiles are extracted from matrix $\boldsymbol{X}$, constructed by stacking vertically the data chunks from the training repository.
3. **Imputation.** To impute data, chunks with missing data are extracted in the same way as in Fig. 1. However, a candidate data chunk $\boldsymbol{X}(t_i)$ is now only selected if it contains at least one missing value. Subsequently, the missing values in each data chunk are estimated as outlined in Sect. 3.3. Since chunks are allowed to overlap with $\frac{2}{3}N$ timestamps, each missing data point occurs in exactly 3 chunks. The relative position of a particular missing value in a chunk has an influence on the matched temporal profiles since each time window captures a different part in time, giving thus slightly deviating estimations. To obtain the most resilient imputation, the average of all three estimations is used to finally impute the missing value.

- For *historical data*, all missing values are imputed at once following the process described above.
- For real-time *streaming data*, at any moment a chunk with missing data is detected and selected, estimations are immediately computed for missing values in that chunk. Because the average of the estimation for the 3 overlapping incomplete chunks is used as final imputation value, the real-time imputation faces a latency of up to one time window (N timestamps). In parallel, one should monitor for concept drift since streaming data might be capturing a deviating or changing phenomenon. In that case, the spatial and temporal profiles need to be updated. To ease this update, chunks with no missing values should be identified as they are encountered and stored in the training data repository in order to be used later on.

3.5 Data and Computer Code

To illustrate the novel imputation approach proposed in this work, experiments have been carried out on a real-world data set from the mobility domain. More specifically, vehicle counts from 16 automatic number-plate recognition (ANPR) cameras were used. As shown in Fig. 2, the cameras are situated in 8 different locations on a circumferential urban highway (i.e., the small ring of Brussels, Belgium), while each camera monitors traffic in one direction. The data covers a period of 20 months, from February 2020 until the end of 2021. Within this period, the amount of vehicles that passed by per minute is provided for each location. This data has been collected using the real-time open API of Brussels Mobility[1], the public administration responsible for the mobility infrastructure in Brussels.

An interesting aspect of this specific data set is that the 16 ANPR cameras are situated along one single road, half of them in each direction (see Fig. 2). Therefore, many vehicles traverse several, if not all, of the 8 locations in one direction, creating a flow of vehicles. Note however that, as only aggregated information about vehicle counts is available, it is not possible to track the trajectory followed by an individual vehicle. It is important to understand that the quality of this real-world data set is not very high. Over 23% of all values are missing, making further advanced analysis of this data not really feasible, unless an appropriate data imputation method could increase the completeness.

The implementation of the proposed methods was done in Python. The Python code can be provided on request.

4 Results and Benchmarking

To validate the proposed imputation method, vehicle counting data for 16 locations, as described in Sect. 3.5, is used. Section 4.1 explains the construction of the training and validation data sets. Next, in Sect. 4.1, the imputation results on the validation data sets are given. Finally, the imputation results are benchmarked in Sect. 4.2.

[1] https://data.mobility.brussels/traffic/api/counts/.

Fig. 2. Map of the ANPR cameras in Brussels, Belgium.

4.1 Validation Strategy

The original real-world data set as described in Sect. 3.5 is first smoothed such that a continuous flow of data points is obtained. The latter is achieved by assigning per minute the mean value of the (known) values for the last 15 min. In this way, time gaps of up to 14 min are filled while the data set's original granularity is preserved.

Validation Data Sets. To allow for a qualitative and objective validation of our imputation approach, the ground truth of the missing values needs to be known. Since the missing data rate of our original data set (>23%) is too high to introduce additional missing values, several validation data sets are constructed as follows. After smoothing, only the 34 days with no missing values are retained. Next, 5 validation data sets with missing values are composed by randomly removing data values from these 34 days. Each data set has a different level of missing values: 5%, 10%, 15%, 20% and 25%.

Training Data Sets. To derive representative spatio-temporal profiles, we need more than 34 full days. For this reason, a dedicated training data set is composed for each of the 5 validation data sets, containing the smoothed data of all 20 available months. The 34 full days are contained with their different amounts of added missing values across the different data sets.

To validate our imputation approach on *historical data*, the training data set allows for the derivation of spatio-temporal models from the full time period, while the validation data set is used to validate the imputation accuracy based on the ground truth of the 34 full days. For the validation of our approach on *streaming data*, we split both data sets in two parts. The first 9 months of the training set are used to derive the spatio-temporal models. Next, the (16)

days in the validation set that fall after those 9 months are used to validate the imputation approach. In this way, we can test how well our historical spatio-temporal patterns can be used to impute future data.

Profile Extraction. As explained in Sect. 3.2, first the spatial and temporal profiles are extracted from the training set. The chunks $\boldsymbol{X_i}$ have a spatial dimension (M) of 16 and a temporal dimension of 3 h ($N = 180$). The latter was chosen as a trade-off between a higher chance to segment a complete time window (no missing values), while still capturing sequences that represent a meaningful tendency. In addition, the lower the value of N, the larger the training set of chunks becomes. Since we use NMF as decomposition approach, the rank hyperparameter has to be chosen for both the extraction of the spatial and the temporal profiles. As validation method for the rank of the temporal (R) and spatial (Q) profile extraction we used the explained variance, i.e., the ratio between the variance after reconstruction and the original variance. We considered a rank resulting in an explained variance of over 99% as fulfilling, although we experimented with some other levels of explained variance. To obtain an explained variance of 99%, an R value of 64 and a Q value of 14 were chosen. To estimate the imputation values, the limited-memory version with bound constrains of the Broyden-Fletcher-Goldfarb-Shanno algorithm (L-BFGS-B) was used [3]. As initial solution for $\boldsymbol{V}^T_{missing}$, a linear interpolation per vehicle counting location was used. As shown in Fig. 3, the amount of latent temporal and spatial profiles has a big impact on the root mean squared error (RMSE) for all of the 5 training and validation data sets. Note that the RMSE values are only based on the artificially removed data values in 34 out of the 708 days since we only know the ground truth of these values. Remarkably, the imputation approach gives better results in the cases where more data is missing. The latter might be attributed to the increased degrees of freedom, avoiding the optimisation algorithm to overfit on the non-missing values. This statement will of course not hold for more extreme ratios of missing values.

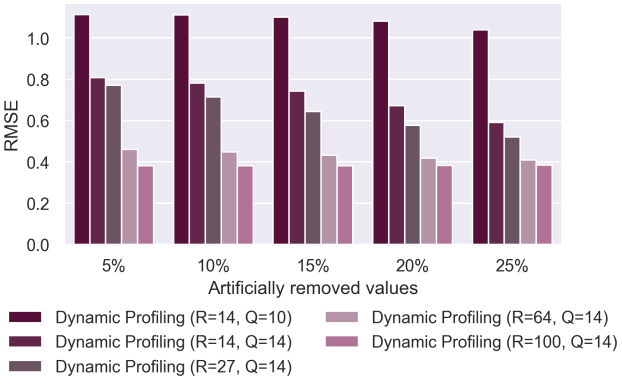
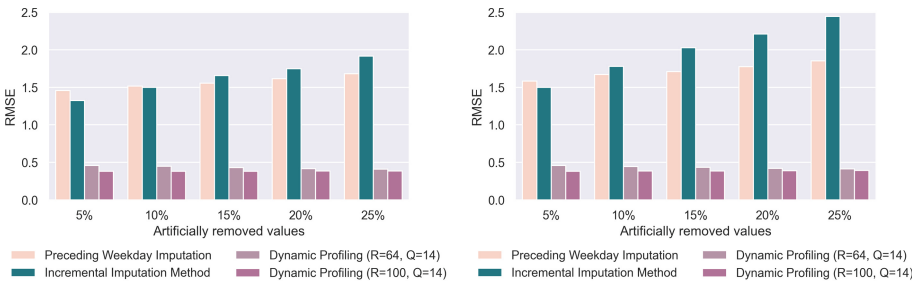


Fig. 3. RMSE values for various hyperparameters on different degrees of missing values.

4.2 Imputation Results

Figure 4a compares the RMSE of the two imputation approaches, in combination with the two best parameter settings from Fig. 3 for the validation strategy on historical data. As baseline imputation algorithm, we simply impute a missing value by copying the value from one week back at the same time. If that value would also be missing, we trace back in past weeks until a value is found. Traffic has a very clear weekly pattern and therefore this imputation method produces quite accurate imputation results as it can be witnessed in Fig. 4a (see "Preceding Weekday Imputation"). To compare the results with a state of the art imputation algorithm, the incremental spatio-temporal imputation method from [6] was chosen. Since the same data set was used in that paper, we reused the same hyperparameters. Compared to the baseline algorithm, only a small improvement was obtained for missing value rates of 5% and 10% as depicted in Fig. 4a (see "Incremental Imputation Method"). Figure 4a also illustrates that the two different versions of our dynamic profiling approach strongly outperform both alternative imputation approaches. Moreover, the strength of the dynamic profiling approach for higher missing rates is also very clearly demonstrated.

In Fig. 4b, the RMSE values for the validation strategy on streaming data are shown. Although this is a more difficult task, as confirmed by the increased RMSE values of the "Preceding Weekday Imputation" and "Incremental Imputation Method", the performance of the dynamic imputation methodology is equally as good as during the benchmarking on historical data imputation. This illustrates that our novel dynamic imputation method is able to robustly extract the latent spatio-temporal structures even from a reduced historical data set.



(a) Historical data benchmarking.

(b) Streaming data benchmarking.

Fig. 4. Comparative results for various imputation strategies on different degrees of missing values.

5 Conclusion

In this work, we introduced a novel data imputation approach for multi-source streaming data. The method relies on spatio-temporal patterns, which are

extracted via a double factorisation approach, and are able to encapsulate latent information structures in historical data. Real-world vehicle counting data has been used for the validation phase. The obtained results show that the approach performs extremely well for data sets with high rates of missing values (20–25%). The latter are very often detected in mobility data.

As future research we plan a further validation of the imputation approach by considering more advanced and realistic patterns for missing values, including a single sensor missing over a longer period and relatively short moments in time where no values are available at all. Our expectations are that the usage of spatio-temporal profiles might be even superior to alternative imputation methods as our approach can exploit patterns from both spatial and temporal dimensions simultaneously. Finally, we will try to improve the dynamic imputation methodology by experimenting with more intelligent initialisation strategies, as these can help the L-BFGS-B algorithm to converge faster and find better estimations.

References

1. Akritas, A., Malaschonok, G., Vigklas, P.: The SVD-fundamental theorem of linear algebra. Nonlinear Anal. Model. Control **11**(2), 123–136 (2006)
2. Bao, Z., Chang, G., Zhang, L., Chen, G., Zhang, S.: Filling missing values of multi-station GNSS coordinate time series based on matrix completion. Measurement **183**, 109862 (2021)
3. Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comput. **16**(5), 1190–1208 (1995)
4. Chen, J., Shao, J.: Nearest neighbor imputation for survey data. J. Official Stat. **16**(2), 113 (2000)
5. Cichocki, A., Phan, A.H.: Fast local algorithms for large scale nonnegative matrix and tensor factorizations. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. **92**(3), 708–721 (2009)
6. Dhont, M., Tsiporkova, E., González-Deleito, N.: Deriving spatio-temporal trajectory fingerprints from mobility data using non-negative matrix factorisation. In: 2021 International Conference on Data Mining Workshops (ICDMW), pp. 750–759. IEEE (2021)
7. Feuerverger, A., He, Y., Khatri, S.: Statistical significance of the Netflix challenge. Stat. Sci. **27**(2), 202–231 (2012)
8. Fountas, P., Kolomvatsos, K.: A continuous data imputation mechanism based on streams correlation. In: 2020 IEEE Symposium on Computers and Communications (ISCC), pp. 1–6. IEEE (2020)
9. Ghosh, B., Basu, B., O'Mahony, M.: Bayesian time-series model for short-term traffic flow forecasting. J. Transp. Eng. **133**(3), 180–189 (2007)
10. Halder, B., Ahmed, M.M., Amagasa, T., Isa, N.A.M., Faisal, R.H., Rahman, M., et al.: Missing information in imbalanced data stream: fuzzy adaptive imputation approach. Appl. Intell., 1–23 (2021)
11. Langville, A.N., Meyer, C.D., Albright, R., Cox, J., Duling, D.: Algorithms, initializations, and convergence for the nonnegative matrix factorization. arXiv preprint arXiv:1407.7299 (2014)
12. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. Nature **401**(6755), 788–791 (1999)

13. Lee, S., Fambro, D.B.: Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting. Transp. Res. Rec. **1678**(1), 179–188 (1999). https://doi.org/10.3141/1678-22
14. Li, X.P., Liu, Q., So, H.C.: Rank-one matrix approximation with l p-norm for image inpainting. IEEE Signal Process. Lett. **27**, 680–684 (2020)
15. Li, Y., Li, Z., Li, L.: Missing traffic data: comparison of imputation methods. IET Intel. Transport Syst. **8**(1), 51–57 (2014)
16. Nikfalazar, S., Yeh, C.-H., Bedingfield, S., Khorshidi, H.A.: A hybrid missing data imputation method for constructing city mobility indices. In: Islam, R., et al. (eds.) AusDM 2018. CCIS, vol. 996, pp. 135–148. Springer, Singapore (2019). https://doi.org/10.1007/978-981-13-6661-1_11
17. Oehmcke, S., Zielinski, O., Kramer, O.: KNN ensembles with penalized DTW for multivariate time series imputation. In: 2016 International Joint Conference on Neural Networks (IJCNN), pp. 2774–2781. IEEE (2016)
18. Qu, L., Li, L., Zhang, Y., Hu, J.: PPCA-based missing data imputation for traffic flow volume: a systematical approach. IEEE Trans. Intell. Transp. Syst. **10**(3), 512–522 (2009)
19. Ren, P., Chen, X., Sun, L., Sun, H.: Incremental Bayesian matrix/tensor learning for structural monitoring data imputation and response forecasting. Mech. Syst. Signal Process. **158** (2021)
20. Shi, Z., Chen, J., He, S.: DIY smart house: exploration and practice of IoT MOOC education. In: 2020 15th International Conference on Computer Science & Education (ICCSE), pp. 557–560. IEEE (2020)
21. Sun, S., et al.: Joint matrix factorization: a novel approach for recommender system. IEEE Access **8**, 224596–224607 (2020)
22. Tipping, M.E., Bishop, C.M.: Mixtures of probabilistic principal component analyzers. Neural Comput. **11**(2), 443–482 (1999)
23. Ye, J.: Generalized low rank approximations of matrices. Mach. Learn. **61**(1–3), 167–191 (2005)
24. Yin, W., Murray-Tuite, P., Rakha, H.: Imputing erroneous data of single-station loop detectors for nonincident conditions: comparison between temporal and spatial methods. J. Intell. Transp. Syst. **16**(3), 159–176 (2012)
25. Zafar, A., Kamran, M., Shad, S.A., Nisar, W.: A robust missing data-recovering technique for mobility data mining. Appl. Artif. Intell. **31**(5–6), 425–438 (2017)
26. Zhang, C., Sun, S., Yu, G.: A Bayesian network approach to time series forecasting of short-term traffic flows. In: Proceedings, The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749), pp. 216–221. IEEE (2004)
27. Zhang, C., Chen, Q., Wang, M., Wei, S.: Optimised two-dimensional orthogonal matching pursuit algorithm via singular value decomposition. IET Signal Proc. **14**(10), 717–724 (2021)
28. Zhong, M., Sharma, S., Lingras, P.: Genetically designed models for accurate imputation of missing traffic counts. Transp. Res. Rec. **1879**(1), 71–79 (2004)