Adaptive Approach for Time Series Imputation Using Matrix Estimation Methods

Abdullah Alomar
Massachusetts Institute of Technology
Cambridge, MA
alomar@mit.edu

Arwa Alanqary
King Abdulaziz City for Science and Technology
Riyadh , Saudi Arabia
aaalangari@kacst.edu.sa

Mansour Alsaleh
King Abdulaziz City for Science and Technology
Riyadh, Saudi Arabia
maalsaleh@kacst.edu.sa

Anas Alfaris
King Abdulaziz City for Science and Technology
Riyadh, Saudi Arabia
aalfaris@kacst.edu.sa

Devavrat Shah
Massachusetts Institute of Technology
Cambridge, MA
devavrat@mit.edu

ABSTRACT

The problem of imputing missing values in time series data has been addressed in many studies that proposed algorithms deemed to be robust for recovering missing values. Despite the good performance of such algorithms, there is usually a lack of theoretical guarantee on their performance. A recently proposed approach to time series imputation uses matrix estimation methods to recover missing values after transforming the time series into a matrix. This approach has exhibited superior performance and provides strong theoretical guarantees of performance for a large class of time series with random missing entries with a certain probability. In this study, we tackle the particular case of missing entries in a time series that form long blocks of consecutive values of different lengths, and we identify the effect of such scenarios, involving missing values, on the accuracy of matrix estimation algorithms for time series imputation. As the main contribution of this work, we propose an extension to the matrix estimation approach to time series imputation by introducing an adaptive algorithm for selecting the shape of the matrix based on the length of consecutive missing values in the time series. The performance of the proposed algorithm was verified by testing it on synthetic datasets consisting of a mixture of autoregressive process, finite sum of harmonics, and a linear trend, as well as real world datasets. Our empirical study shows that the proposed adaptive approach enhances the accuracy of imputation compared with the choice of a fixed matrix shape in 89% of the experiments. The improvement in performance is more significant in missing values scenarios with a more diverse lengths of consecutive missing values, and the enhancement in the performance reaches as high as 30%. Furthermore, we demonstrate that the proposed algorithm outperforms state-of-the-art R-based imputation algorithms in these experiments.

KEYWORDS

Time series, matrix estimation, imputation.

1 INTRODUCTION

Data in the form of univariate time series are encountered in a wide range of domains, such as social sciences, meteorological observations, the energy industry, and finance. After the data are measured and recorded, the problem of missing values is usually inevitable for various reasons. Possible practical scenarios include sensor malfunction, communication errors, and noise in the data. The estimation of these missing values is a common challenge when undertaking time series processing and analysis because algorithms are usually promised a complete signal. Therefore, the accurate replacement of the missing data with reasonable values, known as imputation, is essential for maintaining accuracy when inferring from and forecasting time series.

In the literature and commonly used statistical tools, many algorithms have been developed to tackle the problem of imputing time series. However, most algorithms that perform well are developed to impute multivariate time series, where additional attributes are employed to enhance imputation. Only a limited number of studies have focused on developing methods to address the special case of imputing univariate time series. Classical methods for imputing univariate time series can be divided into three categories [16]: simple statistical measures that do not use the characteristics of the time series, such as the mean, median, and the mode; univariate time series algorithms that consider the nature of the time series, such as arithmetic smoothing, linear interpolation, and imputation using structural time series models such as autoregressive integrated moving average (ARIMA) and seasonal ARIMA models; and multivariate time series algorithms, where for univariate time series lags or leads are used as the other covariates. Although such algorithms exhibit good imputation performance in many applications and are widely accepted in the data science community, they usually lack theoretical guarantees on their performance.

A recently proposed approach that outperforms standard software packages uses methods of matrix estimation to recover missing values in univariate time series [1]. This algorithm transforms the observed time series into a matrix and utilizes well-established matrix estimation methods to recover missing values. This work transforms the problem of imputing missing values in a time series into a matrix estimation problem, and provides strong performance guarantees of this approach for a large class of time series models when the time series entries are missing randomly with a certain probability.

In this work, we revisit the recent approach of imputing time series using matrix estimation, described in [1], to study its performance in the case where the time series suffer from missing data in the form of blocks of consecutive values of diverse lengths, and propose a modified algorithm that enhances the accuracy of imputation in such scenarios involving missing values. A compulsory initial step in approaching the problem of time series imputation as a matrix estimation problem is to transform the observed time series into an observation matrix. Our empirical study argues and shows that the best choice of the shape of the observation matrix for imputing missing values in a given time series is influenced by the number of consecutive missing values. It also shows, that an enhanced imputation performance can be achieved if the shape of the matrix is selected adaptively based on the number of the consecutive missing values.

The proposed algorithm clusters blocks of missing values in the time series based on their lengths and establishes a mapping between the lengths of the blocks and the best choice of the shape of the observation matrix. Once this mapping is known, this adaptive algorithm constructs multiple observation matrices, each with a unique shape that corresponds to a certain cluster. Each observation matrix is then used to recover only the missing values in its corresponding cluster. Figure 1 shows a diagram of the algorithm.

Contributions. Our contributions include the following:

- We study the effect of consecutive missing values in a time series on choosing the optimal shape of the observation matrix when the time series is converted into a matrix for imputation by matrix estimation methods.
- We propose an adaptive approach that extends the algorithm proposed in [1] for imputing missing values in time series to address the effect of consecutive missing values.
- We conduct a variety of experiments to assess the accuracy of the proposed adaptive algorithm. The experiments were conducted for both synthetic and real world time series, and the performance of the proposed algorithm was compared with that of the static imputation method and benchmarked with the R-based time series imputation package Amelia II [11]. In these experiments, the adaptive algorithm outperformed the static algorithm in 89% of the time with an improvement in the imputation error reaching as high as 30%. The algorithm also outperformed the benchmark package in most cases with an improvement in the performance reaching up to 10.9 times.

Organization. In section 2 we survey related work on three relevant topics: time series imputation, matrix completion, and a recent approach of imputing time series using matrix completion. In section 3 we describe the proposed algorithm, starting with transforming the time series into a matrix and performing matrix completion on the transformed time series, followed by the extension to the algorithm where an adaptive choice of the shape of the matrix is introduced. In section 4 we cover the design of the experiment and the datasets used to test the performance of the algorithm. Section 4 also shows the results of the empirical study by comparing the performance of the algorithm proposed in section 3 to the static imputation algorithm and with our benchmark R-based imputation package. Lastly, section 5 summarizes and analysis the results of the empirical study.

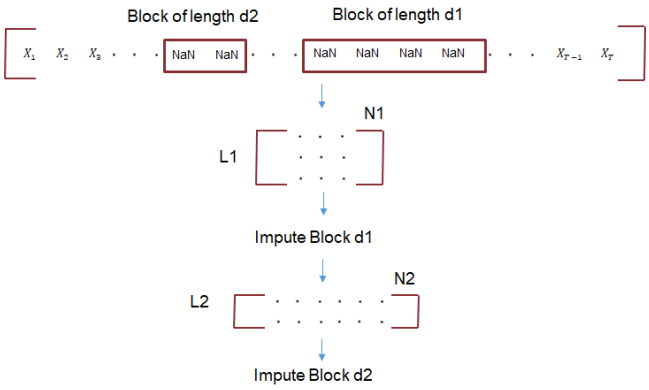


Figure 1: A diagram illustrating the proposed adaptive imputation algorithm for a time series with two missing values blocks with lengths d_1 and d_2

2 RELATED WORK

Imputing missing values in time series data is a popular area of research where many methods and algorithms have been developed to solve the problem of missing data and increase the accuracy of imputation. Examples of methods of time series imputation include the expectation maximization algorithm [9], an iterative approach for computing the maximum likelihood of data that can handle missing values. Hot deck methods are another class of methods commonly used for imputing missing data. In essence, hot deck methods involve replacing missing values with randomly selected similar records using different techniques [2]. One hot deck technique that has proved superior to others is the k-nearest neighbor method, in which the imputed value is the average of k records similar to the missing value according to a certain similarity metric [12]. In addition to developing new methods and algorithms for imputing missing data, considerable effort has been made to address the accuracy of prevalent methods and mitigate their drawbacks. One such commonly used method is the multiple imputation method, which replaces each missing value with two or more values representing a distribution of likely values and resulting in two or more completed datasets. Each dataset is then used separately for further analysis. The multiple imputation method is used to handle bias in variance in single imputation [21]. Most of these methods have been summarized and compared in recent comparison studies [10, 17].

It is worth noting that the problem of imputing univariate time series is considered a special case of the more general imputation problem. The complexity of imputing a univariate time series arises from the need to employ temporal dependencies to perform effective imputations of missing values instead of employing covariates like in multivariate data. Despite the rich literature on longitudinal data imputation, methods designed for imputing univariate time series are limited. The problem of dealing with missing values in univariate time series is often converted into a multivariate imputation by introducing lags or leads of the variable as covariates [16].

In an effort to address the problem of univariate time series imputation, Walter et al. utilized the box Jenkins techniques, including SARIMA and ARIMA, to impute missing values in univariate non-stationary seasonal time series. They also examined the use of direct linear regression models to impute missing values. Working in a similar direction, Chandrasekaran et al. developed the seasonal moving window algorithm to handle missing data in seasonal time series. The algorithm performs seasonal and trend-related decompositions using Loess, and deals with each component of the time series in the imputation step. It linearly interpolates the trend component and searches through observed data from the seasonal component to identify the best cyclic pattern to fill in the missing values [6].

More recently, methods that expand on classical methods and combine different, well-known techniques to improve accuracy have been developed. For example, FLk-NN is a method that combines two imputation methods, Fourier transform imputation and lagged k-nearest neighbor imputation. The average of the estimation provided by these two methods is used as the final estimate of the missing value. This method outperforms commonly used imputation methods, including MICE [16] and several EM approaches [18, 22]. Moreover, a recently proposed paper [19] attempted to mitigate the impact of consecutive missing values on the accuracy of imputation. This involved creating an ensemble of models based on weighted kNN while using dynamic time warping (DTW) as the distance measure. This method performs well even when there is a large number of consecutive missing values, which is mainly due to the penalty function applied to linear interpolation preprocessing step as it increases with the number of consecutive missing values.

This recent development in the field of imputing missing data also resulted in numerous tools and software packages that implement various imputation algorithms. The development of tools for imputation in the statistical environment R prevails in terms of the number of packages developed. The most popular of these packages are ones that implement multiple implementation techniques, such as Amelia that implements expectation maximization with bootstrapping algorithm [11] and MICE, which implements multiple imputation by chained equations [4]. Many other imputation techniques have also been implemented in R. Examples include the Yaimpute packages for KNN imputation [8], mtsdi package for imputation with expectation maximization [13], and missForest for imputation based on random forest [23]. One of the few packages that specialize in univariate time series imputation is imputeTS [15]. It provides an implementation framework for univariate time series covering several algorithms. ImputeTS implements four simple imputation methods: last observation carried forward, missing value imputation by mean value, missing value imputation by random sample, and replacing missing values by defined values. It also implements five sophisticated imputation methods: imputation by linear, spline and stuntman interpolation; imputation by structural model and ARIMA state space representation with Kalman smoothing; imputation by seasonal decomposition; imputation by seasonal splitting; and imputation by simple, linear, and exponentially weighted moving average.

In a recent paper [1], an algorithm that imputes missing data through matrix estimation methods was presented. The algorithm transforms the time series into a matrix and applies matrix estimation on the constructed matrix to impute the missing values. The algorithm, which establishes strong links between univariate time series and matrix estimation, is model agnostic, and is applicable to a wide range of times series models. These include finite sum of harmonics, linear time-invariant systems, and their additive mixtures. The authors provided theoretical guarantees of the performance of this algorithm for both univariate time series imputation and forecasting. Furthermore, they demonstrated the viability of using matrix estimation, in particular, the universal singular value thresholding (USVT) algorithm for imputing univariate time series through multiple experiments on synthetic and real world datasets. The results of these experiments show a superior performance of this algorithm when compared to standard software packages, even when these packages were aware of the underlying model of the time series.

The establishment of such a link between imputing univariate time series and matrix estimation opens the door to applying many matrix estimation algorithms to recover missing data in time series. This problem of matrix estimation has received considerable attention recently owing to its various applications, including collaborative filtering, remote sensing, and computer vision. The takeaway from these efforts is the ability to reconstruct a matrix from few noisy entries by a low-rank approximation of the observed data. Examples of recent work that tackles this problem include an algorithm that uses the concept of local approximation [3]. The relevant studies have proposed methods that estimate missing values by determining the data points nearest to the given one in terms of a specific distance metric, and computing the neighborhood average to estimate the final estimate. Alternatively, singular value thresholding has been used extensively for matrix estimation, including singular value thresholding (SVT) [5], universal singular value thresholding [7], SVD-IMPUTE [24], and Soft-Impute [14]. Mazumder et al. [14] compared many major matrix estimation algorithms, including the newly introduced Soft-Impute, SVT, and MMMF [20].

3 METHODOLOGY

The methodology used is an extension of that proposed in [1], where the imputation of missing values in a time series is transformed into a matrix estimation problem.

3.1 Set-up

An essential initial step in this algorithm is to transform the univariate time series $X(t)$ into an observation matrix M. This transformation process is dependent on parameter L, which denotes the number of rows in the newly constructed matrix.

(1) Denote the observations of the time series at time t by $X(t)$, where $t \in [1, T]$.
(2) For a certain $L \geq 1$ and $N = \lfloor T/L \rfloor$ the element m_{ij} in the $L \times N$ observations matrix M can be expressed as

$$m_{ij} = X(i + (j-1)L)$$

where $i \in [1, L]$ and $j \in [1, N]$

The time series is transformed into a matrix by filling an $L \times N$ matrix with its observations $X[1 : NL]$ column by column.

3.2 Static Imputation Algorithm

Once the time series has been transformed into matrix M, the task of imputing the missing values reduces to a matrix estimation problem. We use an iterative singular value thresholding (SVT) algorithm inspired by iterative SVD-Impute described in [24]. The algorithm performs SVT iteratively to estimate the missing values in matrix M by performing the following steps:

(1) Define an $L \times N$ matrix W with $(i,j)^{th}$ entry w_{ij} and initiate it as

$$w_{ij} = \begin{cases} m_{ij} & \text{if } m_{ij} \text{ is observed} \\ 0 & \text{otherwise} \end{cases}$$

(2) Choose a rank K as the rank of the finally reconstructed matrix and define a small number ϵ as the convergence threshold. Then, in each iteration $q \in [1, Q]$,
   (a) Select the gradually increasing rank k_q defined as

$$k_q = min(2q, K)$$

   (b) Perform singular value decomposition on matrix W

$$W = \sum_{i=1}^{L} s_i u_i v_i^T$$

   (c) Define S as a subset of the singular values of matrix W such that

$$S = \{i : i \leq k_q\}$$

   (d) Reconstruct the reduced rank matrix $\hat{W}$, whose entries are $\hat{w}_{ij}$, as

$$\hat{W} = \sum_{i \in S} s_i u_i v_i^T$$

   (e) Update the entries of matrix W such that

$$w_{ij} = \begin{cases} \hat{w}_{ij} & \text{if } m_{ij} \text{ is missing} \\ w_{ij} & \text{otherwise} \end{cases}$$

   (f) The method converges when the mean difference between the estimated missing values in two consecutive iterations is less than the threshold ϵ. The last iteration is denoted by Q.

(3) Declare the estimated complete time series

$$X(i + (j-1)L) = w_{ij}$$

Assumptions and Parameters. The algorithm assumes that the $L \times N$ observation matrix M is constructed with $L < N$. If L is selected such that $L > N$, the algorithm should be applied to the transpose matrix M^T. This algorithm depends on two parameters: the rank of the final reconstructed matrix K and the number of rows L of the observation matrix M. It has been shown that the most effective way of selecting the values of parameters K and L is through cross-validation.

The choice of parameter L contributes significantly to the error bound of the estimated matrix W. Theorem 3.1 in the work done by Agarawal et al. [1], which gives the accuracy of using the SVT algorithm for imputing time series, suggests that L should be as large as possible. This iterative SVD algorithm, with a single choice of L, is herein referred to as the static imputation algorithm.

3.3 Adaptive Imputation Algorithm

One of the main contributions of this paper is the proposal of an extension to the iterative SVT algorithm to perform it with an adaptive selection of parameter L. The main motivation for this adaptive choice is the effect of consecutive missing values on the optimal shape of the observation matrix.

In the time series $X(t)$, define missing values blocks $b_u, u \in [1, B]$ as continuous chains of consecutive missing values, each of length d_u, where B is the number of missing blocks. This adaptive algorithm constructs multiple observation matrices, each with a unique choice of L, and uses each of these constructed matrices to impute missing values blocks of similar lengths.

In practice, finding the optimal choice of L for a given number of consecutive missing values d_u in a specific time series is performed empirically for the time series of interest. Artificial masking with diverse block lengths d_u is performed on the observed parts of the time series to map the length of the blocks to the optimal choice of L. This mapping is then applied to the missing values in the time series, where the optimal choice of L is used to impute each block of missing values. The steps below describe the adaptive SVT algorithm in more detail.

(1) Group the blocks b_u into Z clusters $\psi_z, z \in [1, Z]$. The blocks are grouped based on block length d_u such that

$$\psi_z = \{b_u : \alpha_z < d_u < \beta_z\}$$

where α_z and β_z are positive integers that define the range of lengths of consecutive missing values d_u of blocks b_u belonging to each of the z clusters.

(2) Initialize time series $\hat{X}_0(t)$ as

$$\hat{X}_0(t) = \begin{cases} X(t) & \text{if } X(t) \text{ is observed} \\ 0 & \text{Otherwise} \end{cases}$$

(3) For each cluster ψ_z
   (a) Construct the observation matrix $M^{(z)}$ from the time series $\hat{X}_{z-1}(t)$ with number of rows L_z, where L_z is the optimal choice of number of rows corresponding to β_z.
   (b) Execute the iterative SVT algorithm on the constructed observation matrix $M^{(z)}$ to produce the estimated matrix $W^{(z)}$.
   (c) Update the missing values belonging to the blocks in cluster ψ_z only such that

$$\hat{X}_z(i+(j-1)L_z) = \begin{cases} w_{ij}^{(z)} & if\, w_{ij}^{(z)} \in \{b_u : b_u \in \psi_z\} \\ \hat{X}_{z-1}(i + (j-1)L_z) & Otherwise \end{cases}$$

(4) Define the final estimated complete time series $\hat{X}_Z(t)$ as

$$\hat{X}_Z(i+(j-1)L_Z) = \begin{cases} w_{ij}^{(Z)} & if X(i + (j-1)L_z) missing \\ X(i + (j-1)L_z) & Otherwise \end{cases}$$

4 EXPERIMENTS

To illustrate the use of the adaptive imputation algorithm described above and test its performance, we conducted experiments on both synthetically generated and real world time series data. The performance was evaluated at different levels of missing values and for diverse lengths of missing values blocks, and was compared against that of the static SVT algorithm and benchmarked against AMELIA

II [11], which is an R-based package that is believed to exhibit excellent imputation performance. The accuracy of imputation was measured using the root-mean-square error (RMSE) calculated as

$$RMSE = \sqrt{\frac{1}{(1-p)LN} \sum_{i \in \Omega} (\hat{X}(t) - X(t))^2}$$

where p is the fraction of observed values in the time series, $(1-p)LN$ is the number of missing values in the time series, and $\Omega = \{i_1, i_2 ... i_{(1-p)NL}\}$ is the set of indices of the missing values.

Masking. To select the parameters of the adaptive algorithm and test its performance, artificial masking was applied to both the training and testing datasets. Masking was performed by selecting multiple blocks of various lengths to remove from the time series. This was performed by first selecting a random index in the time series to identify the location of the missing value block, and then drawing the length of the block from a half-normal distribution with mean zero and variance σ and ceiling the drawn value. In this experiment, three variances for the distribution of the lengths of the missing value blocks were selected: 4, 15, and 25. For each variance, the time series was masked multiple times with different fraction of observed values ranging from 0.9 to 0.4. The different variances were chosen to create diverse block lengths to test the performance for different numbers of consecutive missing values.

Selecting the algorithm's parameters. Our algorithm takes into consideration that the length of consecutive missing values can influence the optimal choice of parameter L. Hence, it is essential to find the optimal choice of L that corresponds to the different block lengths before imputing missing values using the adaptive algorithm. To demonstrate how the optimal choice of parameter L is influenced by the number of consecutive missing values in a time series, we created multiple artificial masks each having identically-sized blocks of consecutive missing values. We then impute each of these masks using different choices of L and observe the associated imputation error. Figure 2 shows how the imputation error (in RMSE) changes with the different choices of L when imputing 5 different masks with blocks lengths ranging from 1 to 300 consecutive missing values. The figure shows a unique optimal choice of L for each of the tested block lengths.

In the following experiments, we find the optimal choice of L that corresponds to each block size d by grouping the missing values blocks into 10 equal clusters based on their lengths. The missing values are imputed using different choices of L to select the best one that yields the minimum error in each of the 10 cluster. Figure 3 shows the mapping between the optimal choice of L and the length of the blocks d for a given time series. This mapping is then used in the imputation process when testing the algorithm on the same time series.

4.1 Synthetic time series experiment

Model of time series. In this experiment we test the performance of our algorithm in comparison with the static algorithm for a time series model consisting of mixture of an autoregressive process (AR) and finite sum of harmonics with a trend component. The synthetic time series is constructed by first generating signals of

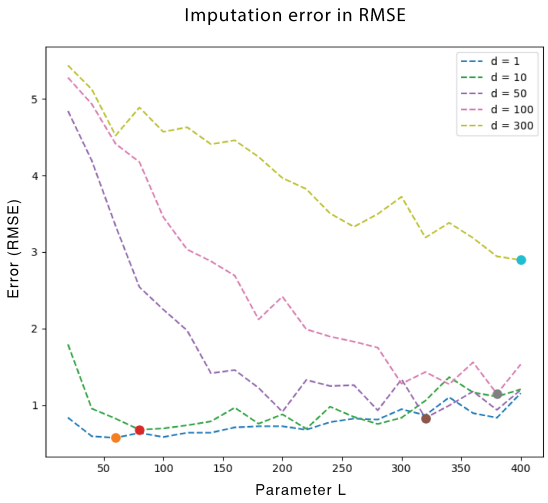


Figure 2: Imputation error in RMSE using different choices of L for five different lengths of missing value blocks d. The bold dots in the figure indicate the value of L that results in the lowest RMSE

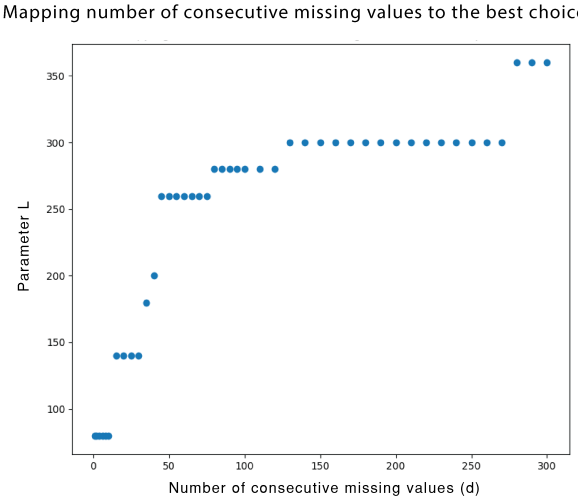


Figure 3: Mapping the lengths of consecutive missing values blocks d to the corresponding optimal choice of parameter L

sum of harmonics as

$$f_1(t) = \sum_{i=1}^{I} A_i sin(2\pi\omega_i t) + B_i cos(2\pi\omega_i t)$$

where i is the number of finite harmonics and ω_i is the frequency of the signal. The AR processes are then generated as

$$f_2(t) = \sum_{i=1}^{I} \alpha_i f(t-1)$$

The two aforementioned signals are mixed, and a linear trend component is added as follows:

$$f(t) = f_1(t) + f_2(t) + \alpha t$$

Finally, the synthetic time series $X(t)$ is modeled as mixtures of signals $f(t)$ with an additive Gaussian noise $\epsilon(t)$ with mean zero and a unit variance.

$$X(t) = f(t) + \epsilon(t)$$

We generated a distribution of 50 synthetic time series with their parameters randomly selected. In particular, each time series has a number of harmonic terms selected uniformly at random from the range $[5, 20]$, with periods $(1/\omega_i)$ ranging from 10 to 30 and coefficients (A_i, B_i) ranging between 0 to 10. The parameters of the AR process are drawn from the uniform distribution $U(0, 1)$, and are chosen to ensure the stationarity of the process. The number of lags is also randomly selected from the uniform distribution $U(3, 10)$.

Imputation. Figure 4 summarizes the performance of the adaptive algorithm in comparison with the static algorithm for different fractions of observed values p and three masking variances σ. For each combination of masking variance and percentage of observed values we report the median relative improvement in the performance (meaured in RMSE) of all 50 time series. It is evident from Figure 4 that the adaptive algorithm performs better than the static choice of L in all masking scenarios. The improvement in the performance is more significant at higher masking variances, as the sizes of missing values blocks become larger and more diverse. The median improvement reaches 27% for $\sigma = 25$. Furthermore, the adaptive imputation algorithm outperformed the static algorithm in 89% of all experiments.

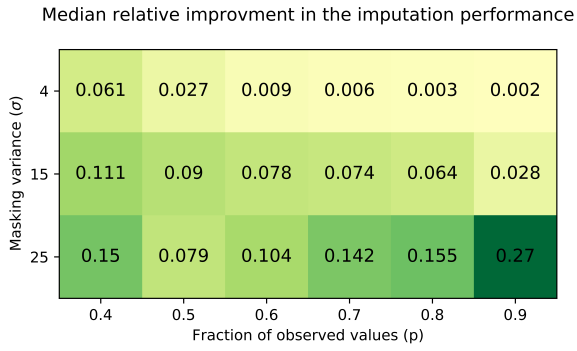


Figure 4: Median relative improvement in RMSE between the adaptive and static algorithms showing a superior performance of the adaptive algorithm in all scenarios of missing values

4.2 Real world time series experiment

Here, the adaptive algorithm is applied to two real world datasets to test its performance. In practice, information concerning the model of the time series, and the type and level of noise in the data are not easily determined. This experiment was undertaken to test the performance of the algorithm in such situations. As the true mean for these time series is not known, we used the observations to compute the error metric (RMSE).

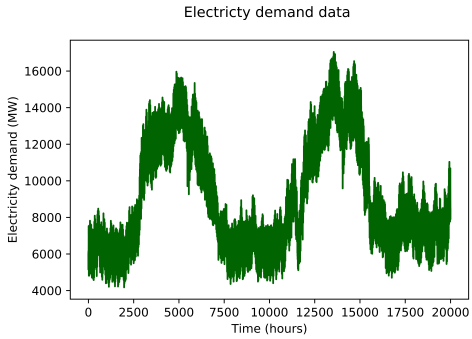


Figure 5: Hourly electricity demand data

Electricity demand. Figure 5 shows a plot of the hourly electricity demand dataset that we used in this experiment. Figure 7 shows the performance in terms of RMSE of the adaptive and static algorithms as a function of the fraction of observed values for the three choices of masking variance. The adaptive approach outperformed all static choices of L at every level of missing values. In Figure 7a, where the masking variance is 4, the enhancement in accuracy over the best static choice of L reached as high as 8%. Further improvement was noticed at higher variances as shown in Figure 7b, where performance was enhanced by up to 18% at σ= 15. The enhancement was even higher as variance increased to 25 to reach as high as 30% compared with that of the static algorithm as shown in Figure 7c. Another noticeable advantage is that the best L in the static method varies when we change the masking variance and the fraction of observed values. This illustrates the adaptive method's resilience to different scenarios of missing values.

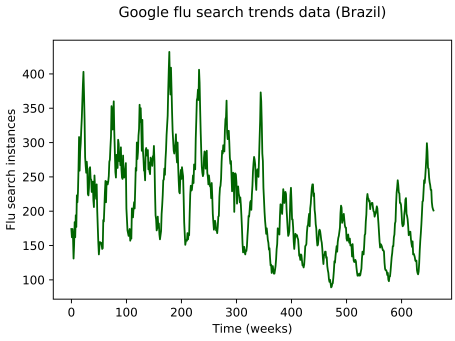
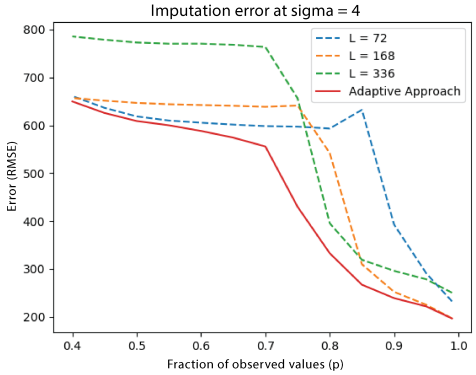
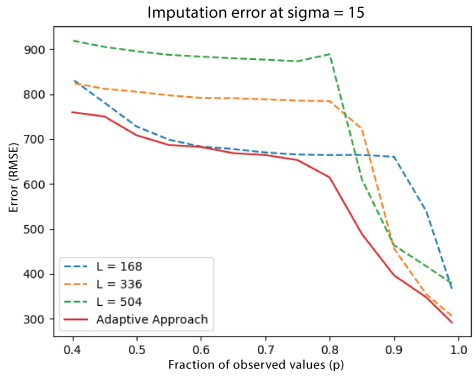


Figure 6: Weekly Google flu search trends data (Brazil)

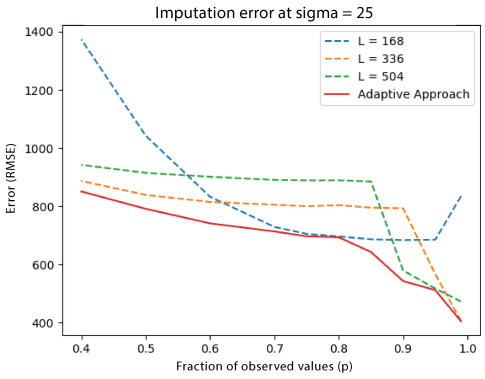
Google flu search trends data (Brazil). Figure 6 shows a plot of the Google flu search trends data in Brazil at weekly intervals. The data exhibited strong seasonality with a period of 52 weeks. Figure 8 shows that applying the adaptive algorithm to this time series did not improve imputation performance, and the best choice of L, which corresponded to the period of the time series, performed as well as the adaptive algorithm at all levels of missing values.



(a) Masking variance $\sigma = 4$

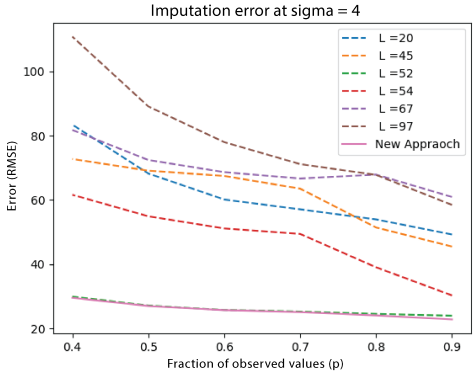


(b) Masking variance $\sigma = 15$

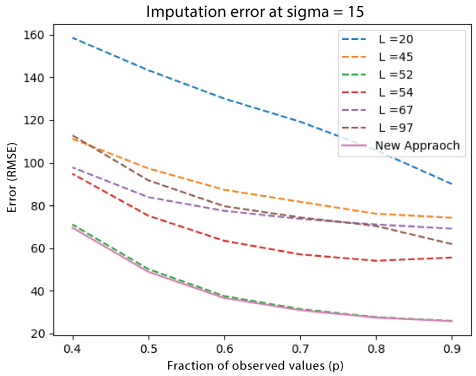


(c) Masking variance $\sigma = 25$

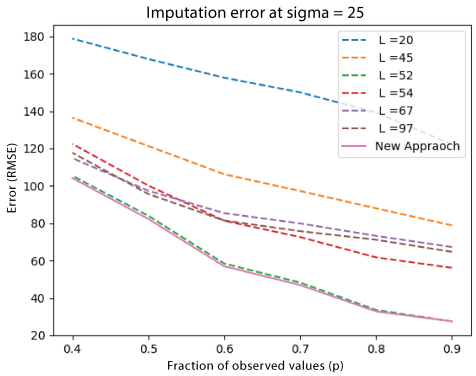
Figure 7: Electricity demand data: the performance of the adaptive imputation algorithm relative to different static choices of L for different masking variances



(a) Masking variance $\sigma = 4$



(b) Masking variance $\sigma = 15$



(c) Masking variance $\sigma = 25$

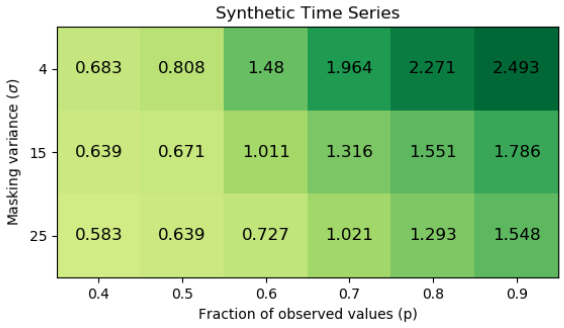
Figure 8: Google flu search trend data: the performance of the adaptive imputation algorithm relative to different static choices of L for different masking variances

4.3 Benchmarking

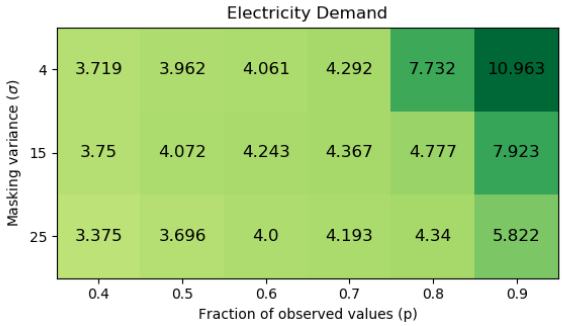
The performance of the proposed adaptive algorithm is benchmarked with the R imputation package Amelia II [11]. Figure 9 shows the relative improvement in RMSE between the adaptive algorithm and Amelia for different fractions of observed values p and three masking variances σ. Positive values (shown in green) mean that the adaptive algorithm performed better while negative values (shown in red) show the opposite. Figure 9 shows how the adaptive algorithm performed better in almost all situations, with a small exception in the flu time series with a high percentage of missing values.

In particular, the adaptive algorithm performs 58% to 249% better than Amelia when tested one of the aforementioned synthetically generated time series. The improvements are even more pronounced in the electricity demand data, where the adaptive algorithm performs 3.4 to 10.9 times better than Amelia.

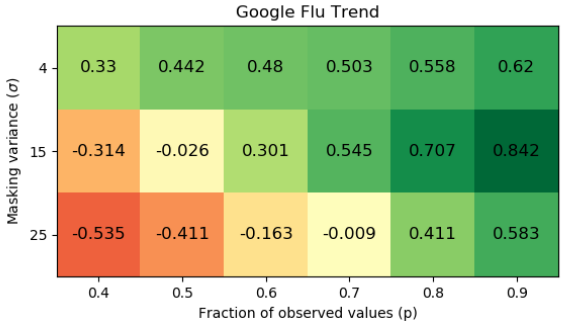
While the performance of the proposed algorithm is often superior when tested on the flu trend data, it is not always the case. Specifically, when the fraction of observed values gets below 60%, our algorithm performs up to 54% worse than Amelia. This is expected given the few data points in the flu trend time series, as our model agnostic method needs more data points than methods built specifically to fit certain models.



(a) Synthetically generated time series



(b) Electricity demand data



(c) Google flu search trends (Brazil)

Figure 9: Benchmarking with Amelia: relative improvement in RMSE between Amelia and the proposed adaptive algorithm shows an up to 10x improvement in performance.

5 CONCLUSION

In this paper, we propose an extension to the time series imputation algorithm developed in [1] that can handle a large number of consecutive missing values in the time series by adaptively choosing the optimal parameters of the algorithm for different lengths of consecutive missing values. Using both synthetic and real world data, we compared the performance of this adaptive approach with the static choice of the algorithm's parameters as described in [1]. Our experimental results suggest that the adaptive approach enhances the performance of imputation when applied to the specified model of time series. The experiments on real world data supports this claim as the algorithm performed significantly better than the static algorithm on one of the time series. On the other experiment on real world data, the adaptive algorithm showed no advantages over the static algorithm. This might be due to the strong seasonality of the time series which creates a strong correspondence between the optimal choice of L and its seasonal period. The algorithm was also compared with the R-based imputation package Amelia and delivered superior performance at most levels of missing values and variances. The only exception was when performed on the Google's flu search trend data where Amelia performed better in case of large numbers of missing values.

REFERENCES

[1] Anish Agarwal, Muhammad Jehangir Amjad, Devavrat Shah, and Dennis Shen. 2018. Time Series Analysis via Matrix Estimation. *arXiv preprint arXiv:1802.09064* (2018).

[2] Rebecca R Andridge and Roderick JA Little. 2010. A review of hot deck imputation for survey non-response. *International statistical review* 78, 1 (2010), 40–64.

[3] Christian Borgs, Jennifer Chayes, Christina E Lee, and Devavrat Shah. 2017. Thy friend is my friend: Iterative collaborative filtering for sparse matrix estimation. In *Advances in Neural Information Processing Systems*. 4715–4726.

[4] S van Buuren and Karin Groothuis-Oudshoorn. 2010. mice: Multivariate imputation by chained equations in R. *Journal of statistical software* (2010), 1–68.

[5] Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen. 2010. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization* 20, 4 (2010), 1956–1982.

[6] Sowmya Chandrasekaran, Martin Zaefferer, Steffen Moritz, Jörg Stork, Martina Friese, Andreas Fischbach, and Thomas Bartz-Beielstein. 2016. *Data Preprocessing: A New Algorithm for Univariate Imputation Designed Specifically for Industrial Needs.* Bibliothek der Technischen Hochschule Köln.

[7] Sourav Chatterjee et al. 2015. Matrix estimation by universal singular value thresholding. *The Annals of Statistics* 43, 1 (2015), 177–214.

[8] Nicholas L Crookston and Andrew O Finley. 2008. yaImpute: an R package for kNN imputation. *Journal of Statistical Software. 23 (10). 16 p.* (2008).

[9] Arthur P Dempster, Nan M Laird, and Donald B Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society.*

Series B (methodological) (1977), 1–38.

[10] Jean Mundahl Engels and Paula Diehr. 2003. Imputation of missing longitudinal data: a comparison of methods. *Journal of clinical epidemiology* 56, 10 (2003), 968–976.

[11] James Honaker, Gary King, Matthew Blackwell, et al. 2011. Amelia II: A program for missing data. *Journal of statistical software* 45, 7 (2011), 1–47.

[12] Per Jonsson and Claes Wohlin. 2004. An evaluation of k-nearest neighbour imputation using likert data. In *Software Metrics, 2004. Proceedings. 10th International Symposium on*. IEEE, 108–118.

[13] W Junger and APde Leon. 2012. mtsdi: Multivariate time series data imputation. *R package 0.3* 3 (2012).

[14] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. 2010. Spectral regularization algorithms for learning large incomplete matrices. *Journal of machine learning research* 11, Aug (2010), 2287–2322.

[15] Steffen Moritz and Thomas Bartz-Beielstein. 2017. imputeTS: time series missing value imputation in R. *The R Journal* 9, 1 (2017), 207–218.

[16] Steffen Moritz, Alexis Sardá, Thomas Bartz-Beielstein, Martin Zaefferer, and Jörg Stork. 2015. Comparison of different methods for univariate time series imputation in R. *arXiv preprint arXiv:1510.03924* (2015).

[17] Michikazu Nakai, Ding-Geng Chen, Kunihiro Nishimura, and Yoshihiro Miyamoto. 2014. Comparative study of four methods in missing value imputations under missing completely at random mechanism. *Open Journal of Statistics* 4, 01 (2014), 27.

[18] Shigeyuki Oba, Masa-aki Sato, Ichiro Takemasa, Morito Monden, Ken-ichi Matsubara, and Shin Ishii. 2003. A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics* 19, 16 (2003), 2088–2096.

[19] Stefan Oehmcke, Oliver Zielinski, and Oliver Kramer. 2016. kNN ensembles with penalized DTW for multivariate time series imputation. In *Neural Networks (IJCNN), 2016 International Joint Conference On*. IEEE, 2774–2781.

[20] Jasson DM Rennie and Nathan Srebro. 2005. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning*. ACM, 713–719.

[21] Donald B Rubin and Nathaniel Schenker. 1986. Multiple imputation for interval estimation from simple random samples with ignorable nonresponse. *Journal of the American statistical Association* 81, 394 (1986), 366–374.

[22] Tapio Schneider. 2001. Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values. *Journal of climate* 14, 5 (2001), 853–871.

[23] Daniel J Stekhoven and Peter Bühlmann. 2011. MissForestâĂŤnon-parametric missing value imputation for mixed-type data. *Bioinformatics* 28, 1 (2011), 112–118.

[24] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. 2001. Missing value estimation methods for DNA microarrays. *Bioinformatics* 17, 6 (2001), 520–525.