
Spatiotemporal forecasting with convolutions and tensor decomposition

A Preprint

David S. Hippocampus*
Department of Computer Science
Cranberry-Lemon University
Pittsburgh, PA 15213
hippo@cs.cranberry-lemon.edu

Elias D. Striatum
Department of Electrical Engineering
Mount-Sheikh University
Santa Narimana, Levand
stariate@ee.mount-sheikh.edu

Abstract

Tensor based time series decomposition methods based on singular spectrum analysis showed great results in both denoising and interpretability. Several forecasting techniques based on them were already explored, yet none provided simultaneously accurate, stable and computationally cheap inferring. After an in-depth study of well known models we facilitated a new one comprising all three requirements for non-stationary quasi-periodic time series. The model was then tested on real-life data of electricity consumption and other well-explored datasets.

Keywords First keyword · Second keyword · More

1 Introduction

Singular spectrum analysis (SSA) is a method widely used in the past decades in different areas, from economics to biology and social science [1]. One of main advantages is its ability to extract underlying frequencies from complex and multidimensional data, resulting in variable number of components. SSA consists of two main stages: decomposition and reconstruction, both adjustable in terms of methods and hyperparameters used.

Its tensor modification (TSSA) offers [2] a more robust and accurate results by converting a series into a tensor and using parallel factor analysis (PARAFAC) decomposition instead of the usual SVD. It optimizes usage of information initially available to a model in cost of working with more multidimensional data SSA does. The problem of reconstruction is left untouched however.

Empirical mode decomposition (EMD) can then be used to supervise TSSA, giving us TSSA-EMD[3]. It provides a better way to identify the number of frequency components within each subspace. With such enhancement algorithms achieves [4] a distinguishable growth in accuracy of signal reconstruction with denoising, leaving other methods far behind in particular tasks.

The problem of forecasting time series has not yet being mentioned. Basic SSA shows [5] adequate results when working with series of constant-limited variation function. However it becomes highly unstable in two basic cases. With several outliers already false frequencies are extracted at decomposition stage, what does not lead to reconstruction defects but enforces unacceptable error even at earlier points of prediction. A variation growth affects SSA the same way, usually creating frequencies of much higher amplitude than are expected. That makes the early predictions seem accurate, yet giving unrealistic forecast long-wise.

To conclude, SSA is a powerful but limited method.

*Use footnote for providing further information about author (webpage, alternative address)—not for acknowledging funding agencies.

At the same time SSA does not utilize spatial information. Given a set of parallel time series it is meant to decompose each separately. TSSA instead can show better performance by working with them as with a whole dataset. Knowing this we experiment to determine the difference between SSA and TSSA forecasting and try to create a robust model of signal decomposition, reconstruction and prediction.

2 Problem

2.1 Time series

Let $x = [x_1, x_2, \dots, x_n]$, $x_i \in \mathbb{R}$ be a 1D time series, namely a vector. We suppose x has no trend, is quasi-periodic and it's phase trajectory is stationary.

[do we need to describe phase trajectory here?]

2.2 Hankelization

One of the SSA steps requires Hankelization operator \mathcal{H} to be described. With 2D matrix $i \times j$ it works as follows:

$$\mathcal{H}M = \begin{pmatrix} \tilde{m}_1 & \tilde{m}_2 & \dots & m_i \\ \tilde{m}_2 & \tilde{m}_3 & \dots & m_{i+1} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{m}_j & \tilde{m}_{j+1} & \dots & \tilde{m}_{i+j-1} \end{pmatrix}$$

$$\tilde{m}_k = \sum_{i,j \in D_k} m_{i,j} / \text{num} D_k \quad (1)$$

$$D_k = \{(\hat{i}, \hat{j}) : 1 \leq \hat{i} \leq i, 1 \leq \hat{j} \leq j, \hat{i} + \hat{j} = k + 1\}$$

For a higher dimensional tensor \mathcal{H} performs first index wise.

2.3 SSA

The basic SSA consists of two stages. The first is decomposition, it maps time series to a number of components. The second is reconstruction, it combines components to frequencies of interest.

Decomposition: given a 1D vector x of size n and window size l it first creates a 2D embedding of it, a trajectory matrix X of size $l_x \times l$:

$$X = [X_1, X_2, \dots, X_l] = \begin{pmatrix} x_1 & x_2 & \dots & x_l \\ x_2 & x_3 & \dots & x_{l+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{l_x} & x_{l_x+1} & \dots & x_{l_x+l-1} \end{pmatrix}$$

$$l_x = n - l + 1 \quad (2)$$

The actual value of l is chosen arbitrarily to include enough information about the signal variance.

Singular value decomposition (SVD) is then applied to X to obtain its singular values and vectors:

$$X = \sum_{i=1}^d X_i = \sum_{i=1}^d \sqrt{\lambda_i} u_i v_i^T \quad (3)$$

Where λ_i is the i th eigenvalue, u_i is the corresponding left singular vector and v_i is the i th right singular vector; d is the total number of eigenvalues.

Reconstruction: given a set of components X_i it first splits them into disjoint set and sums independently:

$$I = \{I_1, I_2, \dots, I_q\}, \bigcup_{i=1}^q I = \{1, 2, \dots, d\}$$

$$X = \sum_{i=1}^q \hat{X}_i = \sum_{i=1}^q \sum_{j \in I_i} X_j \quad (4)$$

The choice of I is one of the most important aspects of SSA in terms of retrieving the desired frequencies. Hankelization is then applied to X and a time series is reconstructed from it (described later).

2.4 TSSA

TSSA is a generalization of SSA to work with a set of parallel time series. It is based on the idea that the time series are not independent and that the information contained in one time series can be used to improve the reconstruction of another.

2.5 Forecasting models based on SSA and TSSA

Copypaste from [Time series SSA forecasting] tutrial link from LinkReview could be found here.

No idea on how to predict with TSSA - we are to explore this I suppose

3 SSA and TSSA comparison

SSA with window width l is taken as a baseline, with optimum l found by gridsearch for each particular dataset.

For TSSA the time series is first converted into matrix by disjoint windowing:

$$X = [X_1, X_2, \dots, X_m] = \begin{pmatrix} x_1 & x_2 & \dots & x_m \\ x_{m+1} & x_{m+2} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{k_x+1} & x_{k_x+2} & \dots & x_{k_x+m} \end{pmatrix}, k_x = \lfloor \frac{n}{m} \rfloor \quad (5)$$

Where the hyperparameter m is gridsearched for each dataset to lower the resulting reconstruction error.

The matrix is then Hankelized with window width l and the PARAFAC is applied to it, resulting in a set of components in Khatri-Rao form. They are then sorted by an individual loss optimization so that the components are in the order of decreasing importance.

For both SSA and TSSA the first- k reconstruction loss is calculated for each k . Area under curve (AUC) is then calculated for the loss curves and the method with the lowest AUC is chosen as the best for a dataset.

As a standalone solution a TSSA with both m and l gridsearched to achive the best results. Compared to SSA, TSSA is computationally expensive, but more information about a time series as whole is saved in each local area inside a hankelized matrix, giving it more opportunities to find the best reconstruction. Its ability to utilize spatial information also makes it applicable to a set of parallel time series, which is not possible with SSA. This results in a better reconstruction error and higher robustness.

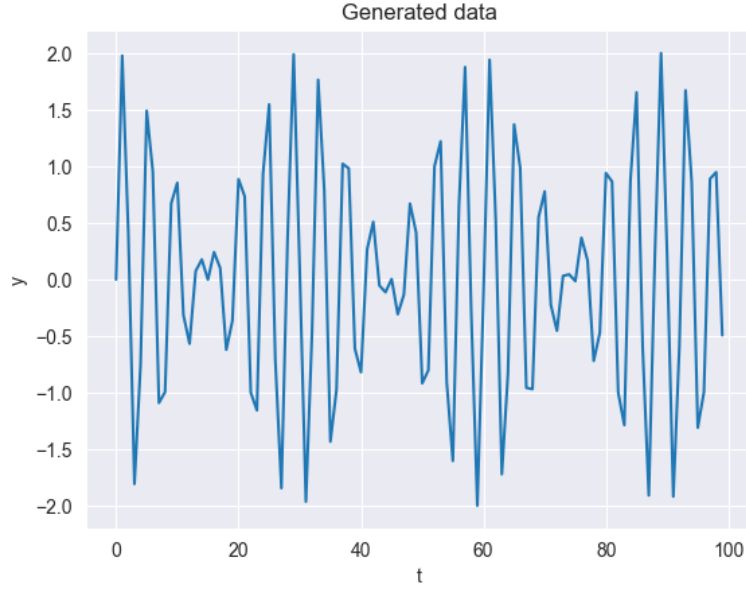
For a real-time application TSSA hyperparameters are estimated on a training set and then used for a suffix of a particular time series realization.

4 Experiment

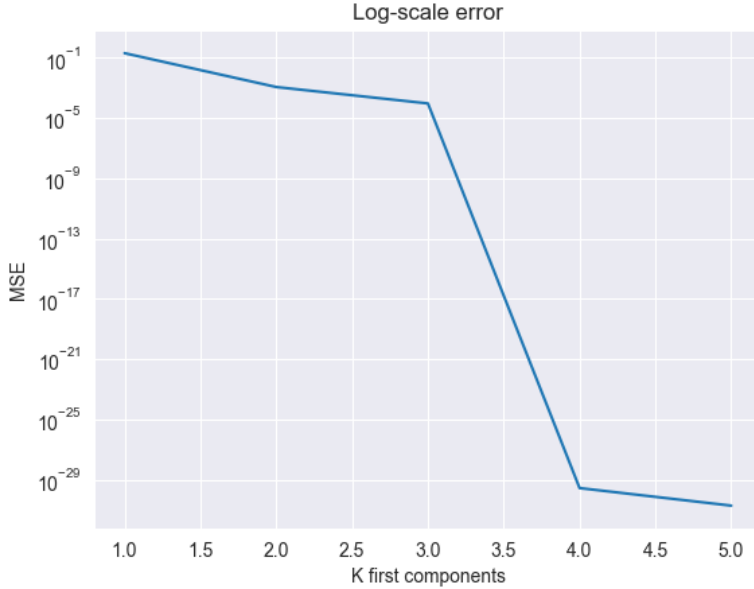
Thought different data exploration multiple reconstruction strategies were tested: first-k SSA, first-k TSSA, [katrutsa-isachenko]. Means squared error (MSE) was used as a measure of error, minimizing it is an optimization task for which optimum method is to be found. Several synthetic time series were generated to test the methods. Then the real data of electricity consumption was used to test the methods.

Preliminary experimets are currently described here. The section will be updated when the real data will be utilized.

We use $f(t) = \sin(\pi t) + np.\sin(et)$ for $t \in (0, 100)$, sampling it uniformly to 100 points to create a time series. The data generated looks as follows:

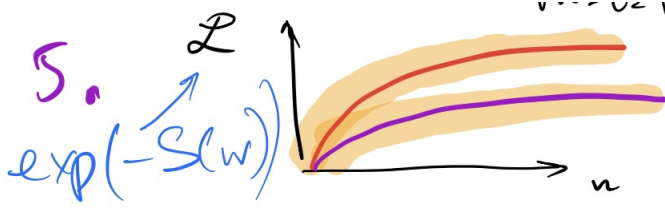


SSA is then applied with different numbers of components taken at reconstruction step. The actual results for log-MSE error for window size 5 are:



With first 4 components taken at reconstruction step the error is non-significant already. More complex data must be taken to see the difference when applying PARAFAC.

The results are in full agreement with the awaited groth of likelihood of error with the number of components taken at reconstruction step.



5 Real data experiments

Example of the experiment on a real-data time series is described here: usage of first- k SSA reconstruction, hankelization of the data and PARAFAC decomposition for it. The results on different datasets are listed in the end.

Electricity consumption dataset consists of 8760 samples - one for each hour of the year. For SSA it is given with window size 24, acquiring that number of base frequencies. For each $k \in [1, 24]$ components are then summed and a reconstruction error MAPE is calculated. For PARAFAC the data is hankelized and then decomposed into Khatri-Rao form with $k = 24$ components. Same prefix sum operation is applied to the components and the reconstruction error is calculated. The results are shown on the figure below.

