



Particularities and commonalities of singular spectrum analysis as a method of time series analysis and signal processing

Nina Golyandina

St. Petersburg State University, St. Petersburg, Russia

Correspondence

Nina Golyandina, St. Petersburg State University, Universitetskaya nab. 7-9, St. Petersburg 199034, Russia.
Email: n.golyandina@spbu.ru

Abstract

Singular spectrum analysis (SSA), starting from the second half of the 20th century, has been a rapidly developing method of time series analysis. Since it can be called principal component analysis (PCA) for time series, SSA will definitely be a standard method in time series analysis and signal processing in the future. Moreover, the problems solved by SSA are considerably wider than that for PCA. In particular, the problems of frequency estimation, forecasting and missing values imputation can be solved within the framework of SSA. The idea of SSA came from different scientific communities, such as that of researchers in time series analysis (Karhunen–Loève decomposition), signal processing (low-rank approximation and frequency estimation) and multivariate data analysis (PCA). Also, depending on the area of applications, different viewpoints on the same algorithms, choice of parameters, and methodology as a whole are considered. Thus, the aim of the paper is to describe and compare different viewpoints on SSA and its modifications and extensions to give people from different scientific communities the possibility to be aware of potentially new aspects of the method.

This article is categorized under:

Data: Types and Structure > Time Series, Stochastic Processes, and Functional Data

Statistical Learning and Exploratory Methods of the Data Sciences > Exploratory Data Analysis

Statistical Learning and Exploratory Methods of the Data Sciences > Modeling Methods

KEY WORDS

singular spectrum analysis, time series, signal processing, decomposition, forecasting

1 | INTRODUCTION

1.1 | References

The origin of singular spectrum analysis (SSA) is usually referred to the papers (Broomhead & King, 1986a; Fraedrich, 1986); although, the algorithm of SSA can be found for example, in Colebrook (1978). SSA became widely known in climatology after publication of Vautard and Ghil (1989) and Vautard, Yiou, and Ghil (1992). After several years, the book (Elsner & Tsonis, 1996) summarized the basic information about SSA existing to that moment. In parallel, SSA (named “Caterpillar”) was created in Russia; the results were published in Danilov and Zhitlavsky (1997). The history of the “Caterpillar”

method starts from Belonin et al. (1971), where O. M. Kalinin is indicated as the author of the ideas underlying the method (it is difficult to find the access to this review; therefore, we refer to the book Belonin, Golubeva, and Skublov (1982, chapter 3, section 8), where this review is cited together with the algorithm of the first stage of SSA). Another source of the Russian branch was the paper (Buchstaber, 1994).

A breakthrough in the theory of SSA was made in the fundamental book (Golyandina, Nekrutkin, & Zhigljavsky, 2001), where the theory is presented together with examples. The next book is Golyandina and Zhigljavsky (2013) in the series Briefs in Statistics; it contains a brief description and some updates from 2001, including description of SSA as a set of filters. From 2013, a large jump was performed, when SSA became a method for analysis of objects of different dimensions and shapes. Also, the R-package **Rssa** (Korobeynikov, Shlemov, Usevich, & Golyandina, 2017) was developed with a very fast implementation of SSA for different kinds of objects. The proposed structured approach to SSA, its multivariate extensions (multivariate [MSSA] and two-dimensional [2D-SSA]) together with algorithms and description of the implementation in **Rssa** are contained in the recent book (Golyandina, Korobeynikov, & Zhigljavsky, 2018).

Three mentioned monographs of Golyandina and coauthors cover a very wide range of problems solved by SSA; however, they only briefly discuss practical applications of SSA to stationary processes. At the same time, the applications of SSA to stationary time series were developed by the team from the University of California, Los Angeles (UCLA) (starting from Yiou, Sornette, & Ghil, 2000), mostly for climatic data. Some practical applications, in particular, in economics and biomedicine, are considered in the works of Hassani, Sanei and their coauthors (see, e.g., the book by Sanei and Hassani (2015); and the review by Hassani and Thomakos (2010)). A separate branch is related to real-world problems in geophysics, where traces in the form of straight lines should be extracted; a preliminary processing is performed by the discrete Fourier transform of the image rows and then complex SSA is applied to the Fourier coefficients (Oropeza, 2010; Trickett, 2003). These branches seem to be developing somewhat independently; therefore, it would be very helpful to enrich one another.

1.2 | Sketch of the algorithm

The SSA algorithm consists of two stages. The first stage is called decomposition, where the studied object (e.g., a time series) is transformed into a trajectory matrix (a Hankel matrix) and then the singular value decomposition (SVD) is applied to the trajectory matrix to obtain a decomposition into elementary rank-one matrix components. The second stage, which is called reconstruction, creates grouped matrix components in a clever way and transforms the grouped matrix decomposition back to a decomposition of the initial object by the so-called diagonal averaging.

Since the idea of considering the subseries of one time series as different observations and then applying principal component analysis (PCA) or Karhunen–Loëve transform (KLT) to the obtained sample is straightforward, Decomposition stage of SSA can be found in many papers; it is hard to detect which paper was first. For example, the mentioned above papers (Belonin et al., 1971; Broomhead & King, 1986a; Fraedrich, 1986; Vautard & Ghil, 1989) contain description only of Decomposition stage. Also, we can cite Basilevsky and Hum (1979) and Efimov and Galaktionov (1983) as references related to the first stage of SSA. “Diagonal averaging” from the second stage is used in Buchstaber (1994), Cadzow (1988), Colebrook (1978), Ghil and Vautard (1991), and Tufts, Kumaresan, and Kirssteins (1982). Nowadays, Reconstruction stage is considered as an essential part of SSA.

Another origin of SSA traces back to properties of Hankel matrices (Gantmacher, 1959). Sometimes, an origin of SSA is drawn from de Prony (1795), where the modeling of the signal in the form of a sum of exponential series was considered; this origin is related to the parameter (frequency) estimation problem.

We suggest to call the method “SSA” if both Decomposition and Reconstruction stages are involved. The methods based on Decomposition stage only are called subspace-based methods. Although many subspace-based methods were developed before SSA, these methods may be called SSA-related.

1.3 | Motivation

Depending on the area of applications, different points of view on the same algorithms, the choice of parameters and the methodology as a whole are considered. The aim of the present paper is to describe and compare different points of view on SSA and its modifications and extensions to give people from different scientific communities the possibility to be aware of potentially new aspects of the method.

1.4 | Structure

Section 2 contains the description of SSA starting from its algorithm and basic ideas. We briefly discuss SSA from different viewpoints to show that the origins of SSA have connections to a very wide range of known methods. Particular cases and extensions of SSA are described. The emphasis is placed upon the differences in descriptions of SSA-related methods in various papers. Section 3 contains specific problems of time series analysis and their solution by SSA in comparison with other methods. In several subsections of Section 3 we touch the same abilities of SSA as in Section 2; however, we look at these abilities from different angles. To minimize intersections between such sections, we provide cross-references. In Section 4, we briefly describe implementations of SSA, since in the era of big data, the effective implementation of a method is a key point for its use in real-world problems. Section 5 concludes the paper.

This review paper is supported by the monograph (Golyandina et al., 2018), which describes and organizes the algorithms of SSA-related methods and their implementation in the R-package **Rssa** (Korobeynikov et al., 2017) with numerous examples. The monograph contains the description of different aspects of SSA itself; in this paper, we put an emphasis on different external viewpoints on SSA and their connection with other methods. To avoid the repetition of figures and to give the readers convenient possibility to look at the illustrative pictures, we put into footnotes the links to the examples from the companion website <https://ssa-with-r-book.github.io> to the book (Golyandina et al., 2018).

1.5 | Detailed structure

1.5.1 | A general review of SSA (Section 2)

1. Time series and digital images: common problems (Section 2.1)
2. Singular spectrum analysis (Section 2.2)
 - An idea of SSA
 - Basic SSA algorithm
 - A view from the Karhunen-Loeve transform
 - A view from stationary processes
 - A view from dynamical systems
 - A view from structured low-rank approximation
3. Decomposition (Section 2.3)
 - Separability
 - How to identify the SVD components
4. Filtering (Section 2.4)
5. Modelling (Section 2.5)
 - Subspace-based approach
 - Signal extraction via projections
6. Choice of parameters (Section 2.6)
7. Theoretical studies (Section 2.7)
8. General scheme of SSA decompositions (Section 2.8)
9. Multivariate/multidimensional extensions (Section 2.9)
 - Multivariate SSA
 - Two-dimensional SSA
 - Shaped SSA
 - Complex SSA
10. Modifications of the SVD step (Section 2.10)
 - Use of a priori information
 - Refined decompositions of signals
 - Tensor SSA

1.5.2 | SSA and different problems (Section 3)

1. SSA and nonlinearity. Is SSA a linear method? (Section 3.1)
2. SSA and autoregressive processes (Section 3.2)

3. SSA and parameter estimation (Section 3.3)
4. SSA and structured low-rank approximation (Section 3.4)
5. SSA and linear regression (Section 3.5)
6. SSA and filtering (Section 3.6)
7. SSA and independent component analysis (Section 3.7)
8. SSA and empirical mode decomposition, discrete Fourier transform and discrete wavelet transform (Section 3.8)
9. SSA: model-free method and modelling (Section 3.9)
10. SSA: forecasting and gap filling (Section 3.10)
11. SSA and signal detection: Monte Carlo SSA (Section 3.11)
12. SSA and outliers (Section 3.12)
13. SSA and a priori/a posteriori information (Section 3.13)
14. SSA: automatic identification and batch processing (Section 3.14)
15. SSA and machine learning (Section 3.15)

1.5.3 | Implementation of SSA (Section 4)

1. Software and fast implementation (Section 4.1)
2. Example of calculations in the R-package **Rssa** (Section 4.2)

2 | A GENERAL REVIEW OF SSA

2.1 | Time series and digital images: common problems

2.1.1 | Decomposition

Let us observe $X = (x_1, \dots, x_N)$, where $X = T + P + N$, T is a trend, P contains regular oscillations and N is noise. The common problem is to construct a decomposition $X = \tilde{T} + \tilde{P} + \tilde{N}$, see Figure 1, which allows one to estimate the trend T , the whole signal $S = T + P$, or periodic components P (P can consist of periodic components with different fundamental periods). Once the signal is estimated and its structure is detected, different signal-based methods such as forecasting can be applied.

If the time series does not contain a signal, then the problem of filtering can be stated; this problem is reduced again to constructing the decomposition, into low- and high-frequency components.

Many problems in digital image processing are similar to the problems stated for time series. Let

$$X = \begin{pmatrix} x_{1,1} & \dots & x_{1,N_2} \\ \dots & \dots & \dots \\ x_{N_1,1} & \dots & x_{N_1,N_2} \end{pmatrix}$$

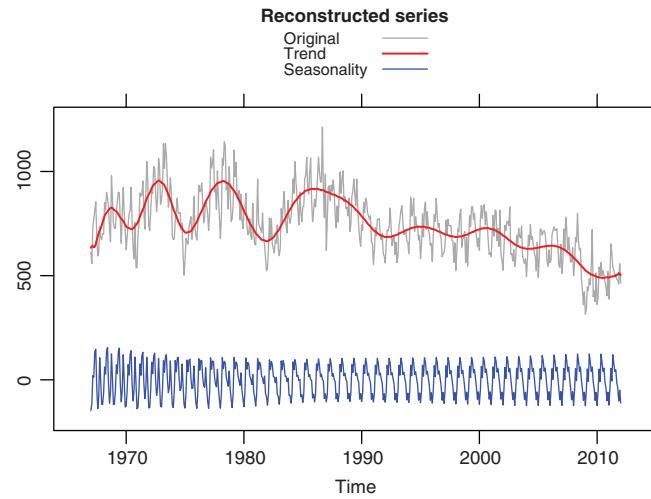
be a digital image, which is modeled as a decomposition $X = T + P + N$ into a pattern, regular oscillations (e.g., a texture) and noise. Then the problem of estimating the decomposition components arises again. See Figure 2, which demonstrates the extraction of a pattern (which is obtained after the removal of regular oscillations).

Note that “digital images” is a common name for 2D data, since one of the dimensions may be temporal. The decomposition problem for multidimensional data is also important in higher dimensions; these dimensions may again be of different nature. For example, one can consider both 3D spatial data and 2D data with the third temporal dimension as 3D data; and so on. We will call data with n dimensions n D data.

2.1.2 | Other time-series/image problems

The results of decomposition (in particular, of signal extraction) allow one to solve many problems for objects with different dimensions and shapes; among these problems are

- trend/tendency extraction;
- smoothing/filtering;

FIGURE 1 “MotorVehicle,” monthly sales: decomposition

- noise reduction;
- extraction of periodic components (including seasonal adjustment);
- frequency estimation;
- construction of a model and parameter estimation;
- forecasting/prediction (of the extracted signal);
- missing data imputation;
- change-point detection.

Thus, the decomposition serves as a starting point for solving many important problems.

2.2 | Singular spectrum analysis

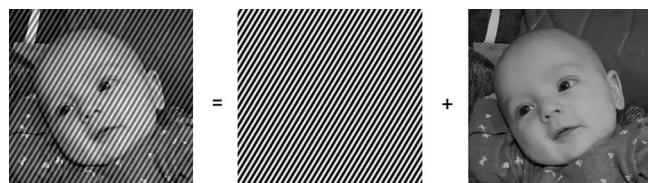
In the paper, we will consider the application of SSA for most of the problems mentioned above. The name “SSA” can be used in both a narrow and a broad sense. One can talk about SSA in a narrow sense meaning the algorithm of decomposition of an object into a sum of identifiable components. SSA in a broad sense consists of methods that use the results of the SSA decomposition. Let us start with the description of SSA as a decomposition method.

2.2.1 | An idea of SSA: To create samples of structure

In multivariate data, there are many observations, which are sampled from the same probability distribution. In data like time series or digital images, we have only one object and therefore need to create many samples of the object's structure to detect it. This can be done by a moving procedure.

Let us start with time series. Denote $X = (x_1, \dots, x_N)$ the time series of length N and choose a window length L , $1 < L < N$; then consider $K = N - L + 1$ vectors constructing from the moving subseries of length L : $X_1 = (x_1, \dots, x_L)^T$, $X_2 = (x_2, \dots, x_{L+1})^T$, ... (the procedure looks like a caterpillar is moving; this step led to one of the method names “Caterpillar”).

For digital images

**FIGURE 2** “Maya”: decomposition

$$\mathbf{X} = \begin{pmatrix} x_{1,1} & \dots & x_{1,N_2} \\ \dots & \dots & \dots \\ x_{N_1,1} & \dots & x_{N_1,N_2} \end{pmatrix},$$

the size of the moving 2D windows is $L_1 \times L_2$; the moving procedure is performed in two directions, from left to right and from top to bottom:

$$\mathbf{X}_{k,m}^{(L_1, L_2)} = \begin{pmatrix} x_{k,m} & \dots & x_{k,m+L_2-1} \\ \dots & \dots & \dots \\ x_{k+L_1-1,m} & \dots & x_{k+L_1-1,m+L_2-1} \end{pmatrix},$$

$k = 1, \dots, K_1$ and $m = 1, \dots, K_2$, where $K_1 = N_1 - L_1 + 1$, $K_2 = N_2 - L_2 + 1$.

2.2.2 | An idea of SSA: To find a common structure using the SVD

Using the obtained set of subobjects of the given object, we should find a common structure. There is a standard statistical method for this, PCA, which is applied to data in matrix form. Hence, the moving subobjects are transformed into vectors, which are then stacked in a matrix called *trajectory matrix*.

Note that PCA is applied to matrices, whose rows and columns have different nature since if the rows correspond to variables, then the columns contain observations. In the case of SSA, the rows and columns of the trajectory matrix have typically the same structure; for time series, both the rows and the columns are subseries of the same time series. Whereas PCA consists of centering or standardization of the variables (the rows of the data matrix) and then applying the SVD to the transformed matrix, SSA usually omits the centering/standardization. The version of SSA which directly applies the SVD to the trajectory matrix is called Basic SSA.

Due to the approximation properties, the SVD allows one to extract the signal (i.e., to remove noise). Due to the biorthogonality of the SVD, the separation of the signal components (e.g., trend and periodic components) one from another is possible. (We refer the reader to Golyandina et al. (2001, chapter 4), where the properties of the SVD, which are helpful for understanding SSA, are gathered.)

For the separation of the signal components one from another, the approximation properties are not necessary. Therefore, other techniques (instead of the SVD) can be applied after the signal extraction. One can see an analogy with the factor analysis of multivariate data, where latent variables should be extracted; first, a number of initial factors are found (e.g., using PCA) and then the rotation of these variables is used for better interpretability. The rotated factors do not have approximation properties.

2.2.3 | Basic SSA algorithm

Here we introduce the algorithm for SSA decomposition of time series. The general form of the algorithm is given in Golyandina et al. (2018) and Section 2.8.

Let $\mathbf{X} = (x_1, \dots, x_N)$ be a time series of length N .

Decomposition stage

Parameter: window length L ; $1 < L < N$:

1. Embedding

The trajectory matrix is constructed by means of the embedding operator \mathcal{T} , which maps a time series to an $L \times K$ Hankel matrix ($K = N - L + 1$) as follows:

$$\mathcal{T}(\mathbf{X}) = \mathbf{X} = \begin{pmatrix} x_1 & x_2 & \dots & x_K \\ x_2 & \ddots & \ddots & x_{K+1} \\ \vdots & \ddots & \vdots & \vdots \\ x_L & x_{L+1} & \dots & x_N \end{pmatrix}. \quad (1)$$

2. Singular value decomposition

The SVD is given by $\mathbf{X} = \sum_{m=1}^d \sqrt{\lambda_m} U_m V_m^T$, where $\{U_m\}_{m=1}^d$ and $\{V_m\}_{m=1}^d$ are orthonormal systems of left and right singular vectors of \mathbf{X} , respectively; $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d > 0$ are squared singular values.

Reconstruction stage

Parameter: way of grouping $\{1, \dots, d\} = \bigcup_{j=1}^c I_j$:

1. Grouping

The SVD components are grouped: $\mathbf{X} = \mathbf{X}_{I_1} + \dots + \mathbf{X}_{I_c}$, where $\mathbf{X}_I = \sum_{m \in I} \sqrt{\lambda_m} U_m V_m^T$.

2. Diagonal averaging

Each matrix \mathbf{X}_I is transferred to the nearest Hankel matrix $\tilde{\mathbf{X}}_I$ by hankelization and then $\tilde{\mathbf{X}}_I$ is transformed into a time series as $\tilde{\mathbf{X}}^{(I)} = \mathcal{T}^{-1}(\tilde{\mathbf{X}}_I)$.

Thus, the output of the SSA algorithm is the decomposition $\mathbf{X} = \tilde{\mathbf{X}}^{(I_1)} + \dots + \tilde{\mathbf{X}}^{(I_c)}$, which corresponds to the structure of the original time series \mathbf{X} if a proper grouping was used.

2.2.4 | Comments to basic SSA

Singular value decomposition

The SVD $\mathbf{X} = \sum_{m=1}^d \sqrt{\lambda_m} U_m V_m^T$ is the biorthogonal decomposition into a sum of rank-one matrices, $d = \text{rank } \mathbf{X}$. By the Eckart–Young theorem (Eckart & Young, 1936), it provides the best low-rank approximation in the Frobenius norm.

The computation of the SVD can be reduced to finding eigenvectors $\{U_m\}_{m=1}^d$ and eigenvalues $\{\lambda_m\}_{m=1}^d$ of the matrix $\mathbf{X}\mathbf{X}^T$, which is sometimes called lag-covariance matrix (despite of centering is not applied); $V_m = \mathbf{X}^T U_m / \sqrt{\lambda_m}$. The collection $(\sqrt{\lambda_m}, U_m, V_m)$ is called the *m*th eigentriple (ET).

If the trajectory matrix \mathbf{X} is considered as a (transposed) data matrix with K cases and L variables, the SVD of \mathbf{X} is closely related to PCA. If rows of \mathbf{X} are centered, the SVD exactly corresponds to PCA. Therefore, according to statistical terminology, V_m and $\sqrt{\lambda_m} V_m$ can be called *factor scores* and *principal components*, respectively.

Grouping

Let $\mathbf{X} = \mathbf{X}^{(1)} + \mathbf{X}^{(2)}$ and therefore the following equality be valid for the trajectory matrices: $\mathbf{X} = \mathbf{X}^{(1)} + \mathbf{X}^{(2)}$. In the case, when there exists a grouping $\{1, \dots, d\} = I_1 \cup I_2$ such that $\mathbf{X}^{(1)} = \mathbf{X}_{I_1}$ and $\mathbf{X}^{(2)} = \mathbf{X}_{I_2}$ (this is called “separability,” see Section 2.3.1), the eigenvectors $\{U_m\}_{m \in I_j}$ form a basis of $\text{colspace}(\mathbf{X}^{(j)})$; therefore, U_m for $m \in I_j$ repeats the behavior of $\mathbf{X}^{(j)}$ (recall that the columns of $\mathbf{X}^{(j)}$ are subseries of $\mathbf{X}^{(j)}$). Thus, we can say that an eigenvector repeats the behavior of a component that produces this eigenvector. Thereby, we can identify slowly varying U_m and gather them into the trend group; then identify regular oscillations and gather them into the periodicity group, and so on.

The elementary grouping, when each group consists of one SVD component (one ET), produces the so-called *elementary reconstructed time series*. Since the grouping and the diagonal averaging are linear operations, it does not matter what operation is performed first, the grouping and then the diagonal averaging or vice versa. Thereby, inspection of elementary reconstructed time series is helpful for performing grouping, since the grouping is simply the summation of these time series.

There is a particular case, when a noisy signal is observed and the signal should be extracted. Formally, this is the case of two groups, signal and noise ones. Since we need only the signal group, the grouping is reduced to the choice of one signal group I . Due to approximation properties of the SVD, the group I corresponds to signal usually consisting of r leading components, that is, $I = \{1, \dots, r\}$. Thus, the way of grouping is reduced to the choice of r .

2.2.5 | A view from the KLT

For a random process ξ_t , $t \in [0, T]$, the structure is contained in the autocovariance function $K(s, t) = E(\xi_s - E\xi_s)(\xi_t - E\xi_t)$; and the KLT based on the eigendecomposition of $K(t, s)$ is considered to express this structure in the form of a decomposition. Originally, the Karhunen–Loëve decomposition of ξ_t is the decomposition into an infinite sum of white noises ε_k , $k = 1, 2, \dots$, with coefficients obtained from the eigenfunctions $u_k(t)$ of the autocovariance function: $\xi_t = \sum_k u_k(t)\varepsilon_k$.

However, for discrete time and a finite time series length, the KLT of (ξ_1, \dots, ξ_N) in fact coincides with PCA of the multivariate random variable $\xi = (\xi_1, \dots, \xi_N)^T$. Hence, the KLT is the decomposition into a finite sum of white noises ε_k , $k = 1, \dots, N$, with coefficients obtained from the eigenvectors U_k of the variance–covariance matrix of ξ : $\xi = \sum_{k=1}^N U_k \varepsilon_k$. For a time series (x_1, \dots, x_N) of length N , which can be considered as a realization of (ξ_1, \dots, ξ_N) , the empirical KLT is constructed on the basis of eigenvectors of the sample covariance matrix. If the time series is centered or, alternatively, centering is applied to the rows of the trajectory matrix \mathbf{X} , then $\mathbf{X}\mathbf{X}^T$ can be considered as an estimate of the (auto)covariance matrix.

The question is how to construct the sample version of ε_k , $k = 1, \dots, N$. In PCA, we have many samples of $(\xi_1, \dots, \xi_N)^T$. Thus, the sample version of ε_k is the k th principal component of the multivariate data. For time series, we have only one realization of length N . Constructing the trajectory matrix \mathbf{X} is the way to create $K = N - L + 1$ L -dimensional samples. Then $\sqrt{\lambda_m} V_m \in \mathbb{R}^K$ can be considered as the sample version of ε_m , $m = 1, \dots, \min(L, K)$.

Thus, up to the above assumptions, the empirical KLT formally corresponds to Decomposition stage of the SSA algorithm (Basilevsky & Hum, 1979). People, who studied the theory of random processes and then looked at the SSA algorithm, sometimes say that it is nothing new, just the KLT. However, note that Reconstruction stage is not included in the KLT decomposition. Also, the point of view on SSA from random process theory leaves a mark on the methodology of applications of SSA (see, e.g., Khan & Poskitt, 2013).

2.2.6 | A view from stationary processes

Researchers who deal with stationary random processes look at SSA in a different way. For (weakly) stationary processes, the autocovariance function $K(s, t)$ depends on $|t - s|$. Therefore, one of the main characteristics of a stationary random process ξ_t , $t \geq 0$, with mean μ is its autocovariance function $C(|s - t|) = K(s, t) = E(\xi_s - \mu)(\xi_t - \mu)$ of one variable. For discrete time and a sequence ξ_1, \dots, ξ_N , this means that the $N \times N$ autocovariance matrix $\{E(\xi_i - \mu)(\xi_j - \mu)\}_{i,j=1}^N$ is Toeplitz. However, in practice, we cannot accurately estimate covariances with large lags, since there are only one pair of observations with lag $N - 1$, two pairs of observations with lag $N - 2$, and so on. In SSA, the autocovariances are estimated up to the lag L ; then not smaller than $K = N - L + 1$ pairs of observations can be used for estimating these autocovariances. Since autocovariances are proportional to autocorrelations for stationary processes, it does not matter which of them is considered.

In basic SSA, the lag-covariance matrix $L \times L$ has the form $\mathbf{C} = \mathbf{XX}^T/K$, which is close to a Toeplitz matrix if the time series is stationary (we assume that the time series is centered); however, this is not exactly a Toeplitz matrix. In Toeplitz SSA, the lag-covariance matrix is estimated in a way to obtain exactly a Toeplitz matrix. The conventional estimation

$\tilde{\mathbf{C}} = \{\tilde{c}_{ij}\}_{i,j=1}^L$ of the autocovariance matrix is $\tilde{c}_{ij} = \frac{1}{N-|j-i|} \sum_{k=1}^{N-|j-i|} x_k x_{k+|j-i|}$. In both Basic and Toeplitz SSA, the decomposition step can be expressed as $\mathbf{X} = \sum_m P_m (\mathbf{X}^T P_m)^T$. In Basic SSA, $P_m = U_m$ are the eigenvectors of \mathbf{C} , whereas in Toeplitz SSA, P_m are the eigenvectors of $\tilde{\mathbf{C}}$.

Since the intention of Toeplitz SSA is a better estimation of the autocovariances, small enough window lengths are usually chosen to obtain stable estimates of the autocorrelations, which should be estimated for lags that are not larger than the chosen window length.

Note that in Vautard and Ghil (1989), the Toeplitz version is called VG according to the author's names, while the basic version with the SVD decomposition is called BK, since it was suggested in Broomhead and King (1986a).

It is important to note that the only form of the decomposition step which is invariant with respect to transposition of the trajectory matrix (that corresponds to the change $L' = K$ and $K' = L$) is the SVD. In particular, for Toeplitz SSA the decompositions for L , K , and $L' = K$, $K' = L$ are different.

It seems that this view from stationary processes limits the application of SSA, since Toeplitz estimates of autocorrelation matrices have sense for analyzing stationary time series only, while the range of problems solved by SSA is much wider. Since the VG (Toeplitz) version was proposed for the analysis of climatic data as a default option, this version is still considered as the main one in many applications; probably, sometimes the Toeplitz version is used just through habit. However, it should be noted that the Toeplitz version of SSA is linked to spectral estimation; in such a case, this version is appropriate.

2.2.7 | A view from dynamical systems

The origin of SSA conventionally refers to the papers devoted to dynamical systems (Broomhead & King, 1986a; Fraedrich, 1986), where the problem is based on the Takens embedding theorem and the method is called singular system analysis.

It seems that the connection between SSA and the Takens theorem is mostly historical. Hence, we omit the description of this “embedology” approach. More details can be found in Sauer, Yorke, and Casdagli (1991). A discussion about the reliability of the Broomhead and King approach as a method of estimating the embedding dimension can be found in Mees, Rapp, and Jennings (1987) and the subsequent comments. Nevertheless, the terminology in SSA is partly taken from the theory of dynamical systems. The dynamical system approach introduced the term “embedding” for the first step of the SSA algorithm; also the term “trajectory matrix” is due to interpretation of columns of the trajectory matrix as a sequence of vectors (a trajectory) in a multidimensional space.

2.2.8 | A view from structured low-rank approximation

It is well-known that rank-deficient Hankel matrices correspond to time series of a certain structure, see, for example, Gantmacher (1959). A particular case of the SSA algorithm, when extracting the signal is of interest (Section 2.5.2), practically coincides with one step of the iterative algorithm suggested in Cadzow (1988) and later named as Cadzow iterations.

The matrix called “trajectory” in SSA, was called “enhanced matrix” in the Cadzow’s paper and subsequent papers (e.g., Hua, 1992 constructs the enhanced matrix in the 2D case for 2D frequency estimation).

More details related to structured low-rank approximation (SLRA) can be found in Section 3.4.

2.3 | Decomposition

Here we consider the problem of decomposition of time series into a sum of identifiable components such as a trend, periodic components and noise. The first question is whether SSA allows one to find such a decomposition; then, if the answer is “yes,” the problem is: how to identify the time series components with the help of information about the SVD components. Also, the question about the choice of window length L , which leads to a better decomposition, arises. We start with the first question.

2.3.1 | Separability

The (approximate) separability of time series components that are of interest in the study is needed to perform a grouping of the SVD components to (approximately) extract these components.

Let $\mathbf{X} = \mathbf{X}^{(1)} + \mathbf{X}^{(2)}$ be an observed time series and $\mathbf{X} = \mathbf{X}^{(1)} + \mathbf{X}^{(2)}$ be the corresponding equality for their trajectory matrices. The SVD step of Basic SSA provides an expansion $\mathbf{X} = \mathbf{X}_1 + \dots + \mathbf{X}_d = \sum_{m=1}^d \sqrt{\lambda_m} \mathbf{U}_m \mathbf{V}_m^T$. Separability means that the set of ETs $(\sqrt{\lambda_m}, \mathbf{U}_m, \mathbf{V}_m)$ in the decomposition of the sum $\mathbf{X}^{(1)} + \mathbf{X}^{(2)}$ is equal to the union of the ETs $(\sqrt{\lambda_{m,j}}, \mathbf{U}_{m,j}, \mathbf{V}_{m,j})$ produced by each time series $\mathbf{X}^{(j)}$, $j = 1, 2$. If this is true, the grouping that separates the time series $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ exists and it is sufficient to identify the corresponding ETs. When the proper grouping is fixed, the diagonal averaging step provides the decomposition of the original time series.

There are two versions of separability related to nonuniqueness of the SVD which is caused by multiple eigenvalues. *Weak separability* means that there exists an SVD decomposition of the trajectory matrix \mathbf{X} , which allows one to group its components and thereby to gather $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$. *Strong separability* means that any SVD allows the proper grouping. In practice, it is necessary to have the strong separability, since the used numerical method constructs some SVD, and therefore the chance that it will coincide with the separating one, is vanishing in the absence of strong separability.

Proposition 1 Let L be fixed. Two time series $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ are weakly separable, if their column trajectory spaces are orthogonal and the same is valid for their row trajectory spaces, that is, $(\mathbf{X}^{(1)})^T \mathbf{X}^{(2)} = \mathbf{0}_K$, \mathbf{K} and $\mathbf{X}^{(1)} (\mathbf{X}^{(2)})^T = \mathbf{0}_L$, \mathbf{L} .

Two time series $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ are strongly separable, if they are weakly separable and the sets of singular values of their L -trajectory matrices are disjoint, that is, $\lambda_{k,1} \neq \lambda_{j,2}$ for any k and j .

Thus, the condition for (approximate) weak separability is the (approximate) orthogonality of subseries of length L and the same for subseries of length K . For example (Golyandina et al., 2001, section 6.1), the time series $x_n^{(j)} = A_j \cos(2\pi\omega_j n + \phi_j)$, $n = 1, \dots, N$, $j = 1, 2$, are asymptotically weakly separable for any $0 < \omega_1 \neq \omega_2 < 0.5$ as N tends to infinity. For exact weak separability, the condition that $L\omega_j$ and $K\omega_j$ are integers should be fulfilled. For strong separability, we additionally need $A_1 \neq A_2$, since it is easily to find that $\|\mathbf{X}^{(j)}\|_F^2 = A_j^2 L K / 2$ and therefore $\lambda_{m,j} = A_j^2 L K / 4$, $m = 1, 2$, for $j = 1, 2$.

Examples of approximately separable time series are: trend and oscillations; slowly-varying components and noise; sinusoids with different periods; seasonality and noise.

Theoretically, the approximate separability is a consequence of the asymptotic separability as $N \rightarrow \infty$. Therefore, the accuracy of decomposition depends on the convergence rate as $N \rightarrow \infty$. For example, for separability of sine waves, the convergence rate is $C/\min(L, K)$, where $C \sim 1/|\omega_1 - \omega_2|$, see Golyandina et al. (2001, Example 6.7). Therefore, for weak separability of sine waves, the choice $L \approx N/2$ is recommended. Also, a worse separability occurs for sine waves with close frequencies.

Fragments 2.4.1–2.4.3¹ of (Golyandina et al., 2018) demonstrate the problem of the lack of weak separability for an artificial example and the way of improving the separability by means of a modification of SSA called iterative oblique SSA (Golyandina et al., 2018, section 2.4). Fragment 2.5.1² shows the problem of the lack of strong separability, also for an artificial example. A real-world example, which explains why we need separability, can be found in Section 2.3.4.

2.3.2 | How to identify the SVD components

The most sophisticated step of SSA is the way of grouping of elementary reconstructed components (RCs). In this section we will demonstrate how it can be done in an interactive way.

Let us enumerate the main approaches (see Golyandina et al., 2001, section 1.6 and the example in Section 2.3.3):

1. To construct the trend group, choose the ETs with slowly varying eigenvectors. The same can be done on the basis of factor vectors or elementary RCs.
2. To extract the periodicity with period T , find pairs of components similar to sine/cosine with periods T/k , $k = 1, \dots, (T-1)/2$ and one saw-tooth component, which corresponds to the period 2 if T is even. The mentioned pairs of sine/cosine can be detected in 2D scatterplots of sequential eigenvectors.
3. To group the components, look at the matrix of weighted correlations between the elementary RCs called w -correlation matrix. The elementary RCs with strong w -correlation should be put into the same group. In particular, noise produces correlated components; a sine wave with period larger than 2 produces two w -correlated elementary reconstructed time series. Note that the name “correlation” can be a bit ambiguous here, since the vectors are not centered; that is, the measure called w -correlation is, in fact, the cosine of the angle between the vectors.

The enumerated properties can be formalized to obtain methods for automatic identification. The approaches to automatic identification are considered in Section 3.14 in more detail.

A special task is the choice of components related to the signal. To extract the signal, we should choose a number of the leading components r . If the signal is used for example, forecasting, r can be chosen by the minimization of the forecasting errors for historical data.

2.3.3 | Example of identification and decomposition

Let us demonstrate how to visually identify the SSA components and to obtain the SSA decomposition, by a simple example. In the 1D scatterplots (Figure 3), one can find a slowly varying component (ET1), whereas in the 2D scatterplots (Figure 4) regular polygons say about pairs of sine-wave components (ET2–3 for $T = 12$, ET4–5 for $T = 4$, ET6–7 for $T = 6$, ET8–9 for $T = 2.4$ and ET10–11 for $T = 3$). In these figures, $U_i(k)$ denotes the k th coordinate of the

FIGURE 3 “Fortified wines,” $L = 84$: one-dimensional graphs of eigenvectors $(k, U_i(k)), k = 1, \dots, L$

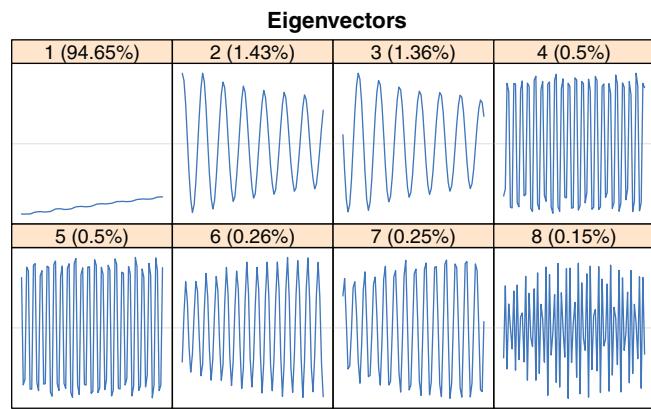


FIGURE 4 “Fortified wines,” $L = 84$: 2D scatterplots of eigenvectors $(U_i(k), U_{i+1}(k)), k = 1, \dots, L$

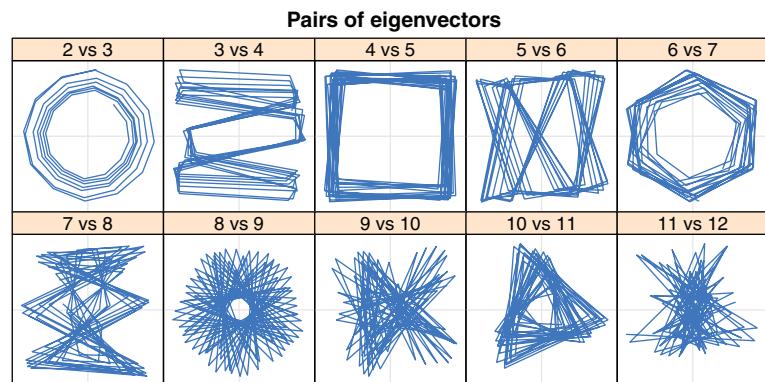
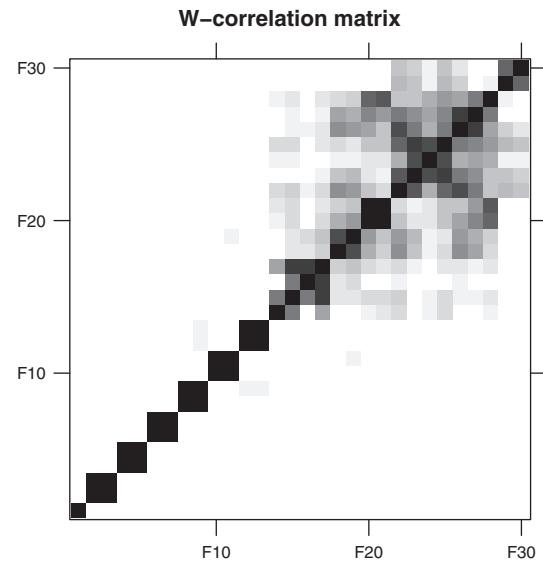


FIGURE 5 “Fortified wines,” $L = 84$: w -correlations between elementary reconstructed components



ith eigenvector obtained in the SVD step of SSA. The eigenvector numbers are indicated at the captions of the graphs. Figure 5 with the depicted w -correlations provides a guess for grouping, since strongly correlated components should be included in the same group (the black color shows correlations close to 1; the white color corresponds to zero correlations).

The resultant decomposition into the trend, the seasonality and noise is depicted in Figure 6. This example is performed by the code of Fragments 2.1.1–2.1.3³ of (Golyandina et al., 2018).

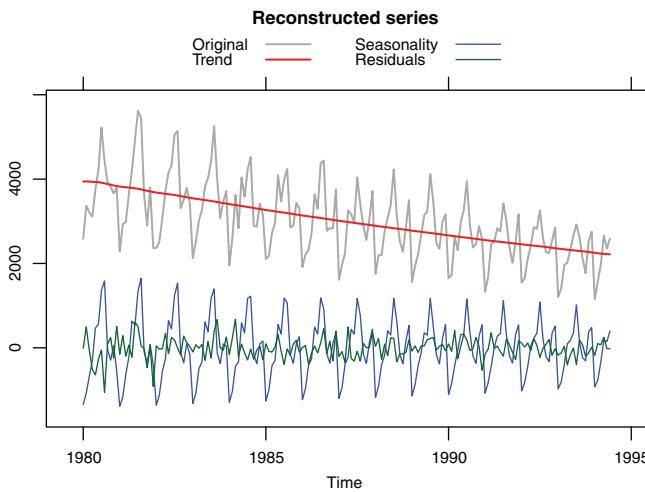


FIGURE 6 “Fortified wines,” $L = 84$: decomposition for groups ET1, ET2–11, and ET12–84

2.3.4 | Example of problems with separability

In Section 2.3.3, we considered an example with simple trend and periodic components. That example corresponds to a good separability between trend, seasonal and noise components. However, for trends of complex form which are common in time series analysis, there is a big chance that trend components of the SSA decomposition can be mixed with seasonal components. Below we consider a short time series and demonstrate the problem of the lack of separability.

Let us analyze the first five years of the time series “MotorVehicle” (Figure 1). The first decomposition is performed by Basic SSA with $L = 24$. Figure 7 depicts 11 elementary reconstructed series, that is, the time series, which are obtained with the grouping $\{1, \dots, d\} = \bigcup_{i=1}^d \{i\}$. These time series are helpful for the grouping procedure, since the reconstruction by ETs from a group I is just the sum of the elementary RCs with numbers from I .

One can see that the trend is contained in slowly varying RC1 and partly in RC2, 8, and 9; the latter elementary reconstructed time series contain a mixture of the trend and seasonality. This means that the reconstruction of the trend with ET1 is insufficient, while that with, say, ET1,2 will contain seasonality (see Figure 8).

Thus, if there is no separability of the trend from the residual, then it is impossible to extract an accurate trend. This example explains, why it is very important to try for separability by different means, such as the choice of the window length, sequential SSA, nested modifications of basic SSA (e.g., iterative oblique SSA, SSA with derivatives; see Section 2.10.2).

Let us apply SSA with derivatives to the group of the 11 leading ETs. SSA with derivatives changes the chosen components (but does not change their sum) in such a way to change their contribution and separate them in the case of the lack of strong separability; the order of the components can be changed and the trend components are typically the last in the chosen group. It is clearly seen in Figure 9 that the trend components are RC9–11. Figure 10 shows the extracted accurate trend.

2.4 | Filtering

It is known that the time series components reconstructed by SSA can be considered as a result of linear filters applied to the original time series (Hansen & Jensen, 1998; Harris & Yan, 2010; Bozzo, Carniel, & Fasino, 2010; Golyandina & Zhitljavskiy, 2013, section 3.9). Certainly, the coefficients of these filters have nonlinear dependence on the time series. Therefore, these filters are called adaptive. This approach is more natural if L is small, since then a larger part of the points of each reconstructed time series is obtained by the same linear filter (see Section 3.6 for details).

2.4.1 | Example

The following example shows that SSA can be considered from the viewpoint of decomposition into components with different frequency ranges, that is, not necessarily as a decomposition into a trend, periodic components and noise. Figure 11 shows

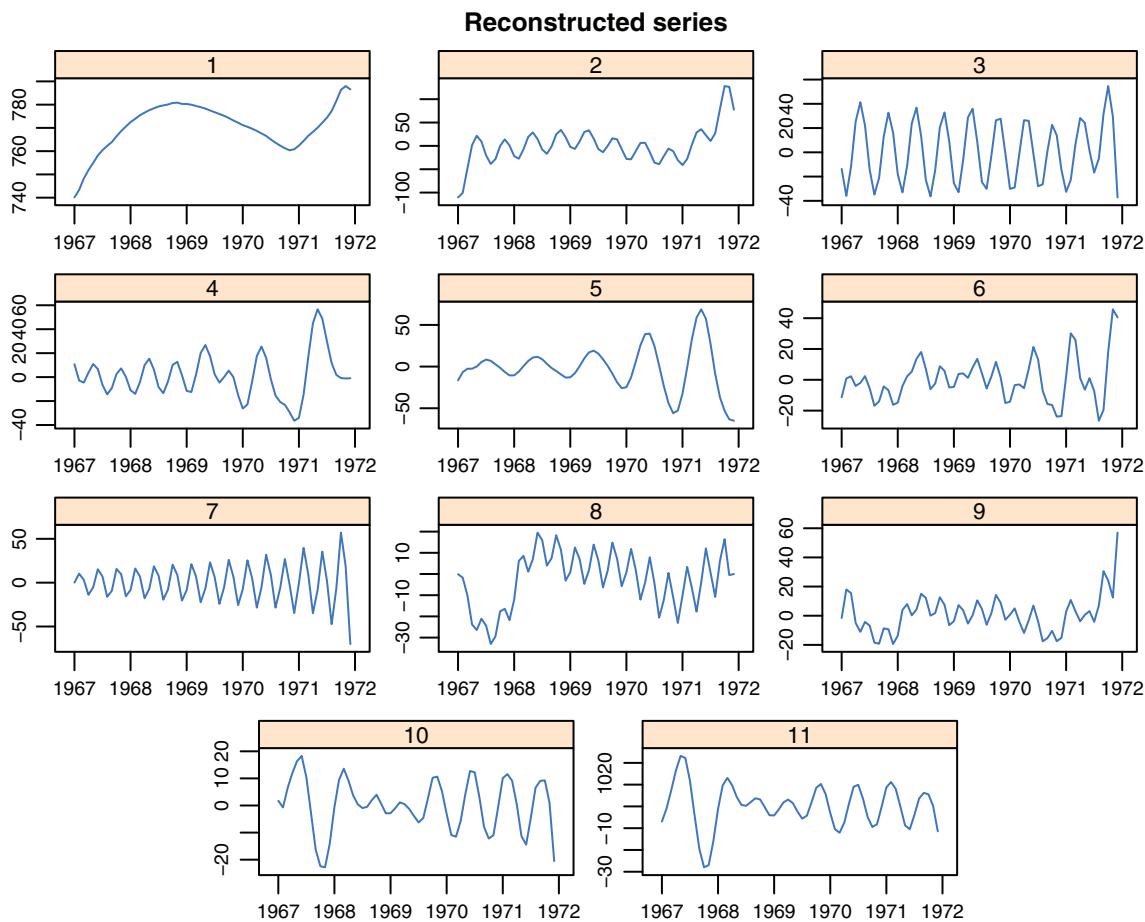
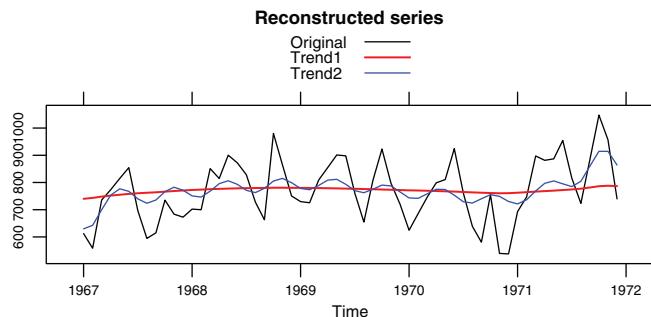


FIGURE 7 “MotorVehicle” (5 years), singular spectrum analysis with $L = 24$: elementary reconstructed components; poor separability

FIGURE 8 “MotorVehicle” (5 years), singular spectrum analysis with $L = 24$: two trend reconstructions, ET1 and ET1–2; poor separability



the decomposition, which was obtained by grouping the components according their frequency range. Figure 12 with components' periodograms depicted together confirms this. This example is performed by the code of Fragment 2.8.1⁴ (Golyandina et al., 2018).

2.5 | Modeling

We mentioned that SSA combines nonparametric (model-free) and parametric approaches; certainly, the latter is possible if a parametric model is stated. Consider the following model of signals that is used in SSA. Let $S = (s_1, \dots, s_N)$ be a signal (or, more precisely, a time series component of interest). Set a window length L , $1 < L < N$; $K = N - L + 1$. Consider the trajectory matrix:

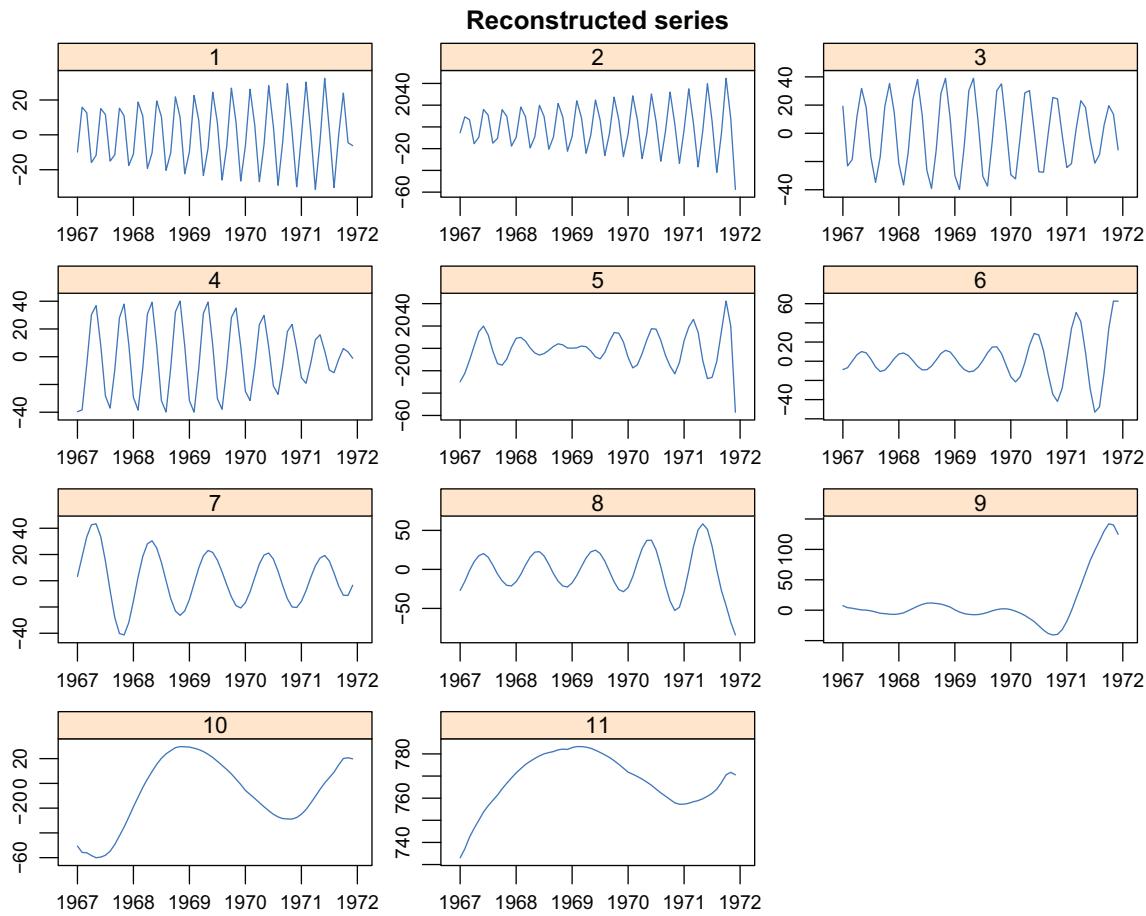


FIGURE 9 “MotorVehicle” (5 years), DerivSSA with $L = 24$: elementary reconstructed components; good separability

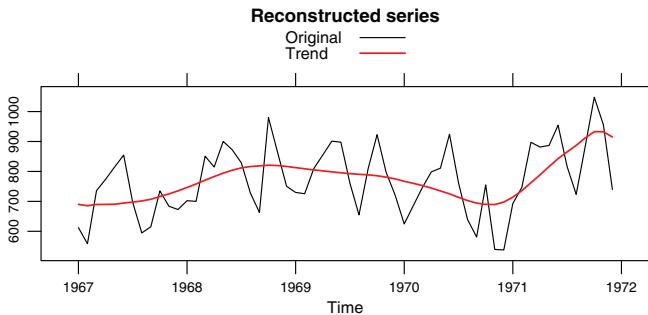


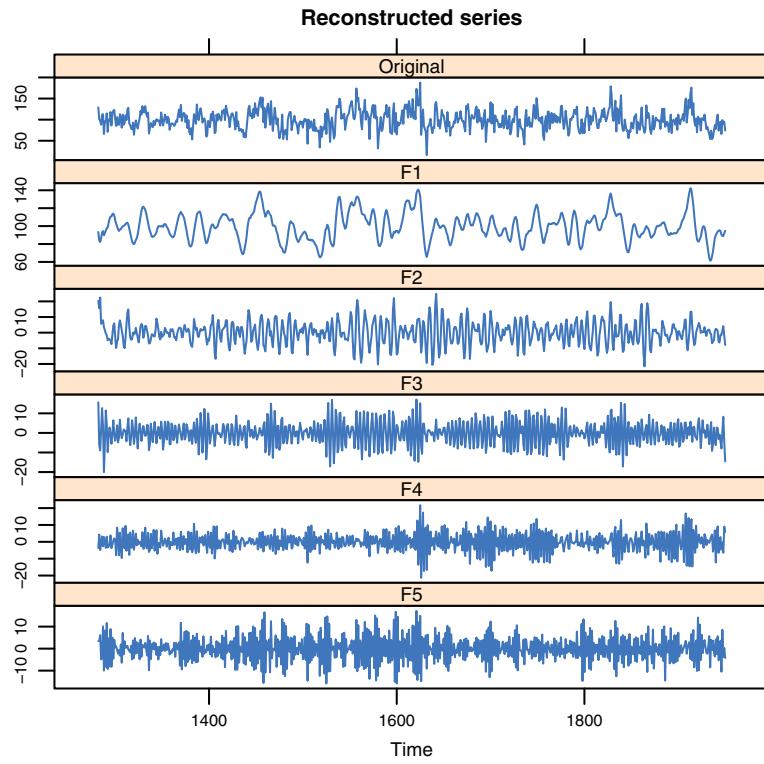
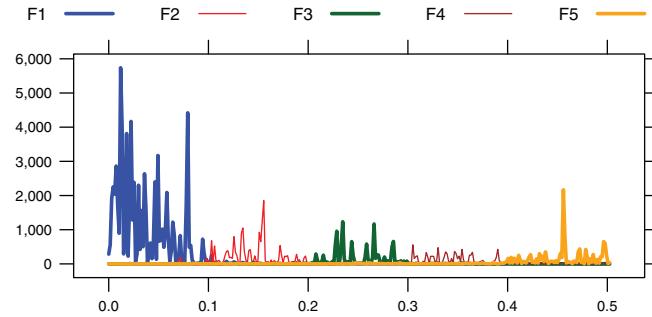
FIGURE 10 “MotorVehicle” (5 years), DerivSSA with $L = 24$: trend reconstruction, ET9–11; good separability

$$\mathbf{S} = \begin{pmatrix} s_1 & s_2 & \dots & s_K \\ s_2 & \ddots & \ddots & s_{K+1} \\ \vdots & \ddots & \vdots & \vdots \\ s_L & s_{L+1} & \dots & s_N \end{pmatrix}.$$

Let r denote the rank of \mathbf{S} .

Different forms of the model are:

- \mathbf{S} is a Hankel low-rank matrix of rank $r < \min(L, K)$; the model can be parameterized by a basis of $\text{colspace}(\mathbf{S})$ or of its orthogonal completion. Such time series are called *time series of finite rank*.
- The time series is governed by a linear recurrence relation (LRR)

FIGURE 11 “Tree rings”: frequency decomposition**FIGURE 12** “Tree rings”: periodograms of the reconstructed time series

$$s_n = \sum_{k=1}^r a_k s_{n-k}, a_r \neq 0, n = r+1, \dots \quad (2)$$

Such time series is called a *time series governed by an LRR*.

- The time series has an explicit parametric form of a finite sum:

$$s_n = \sum_j p_j(n) \exp(\alpha_j n) \sin(2\pi\omega_j n + \phi_j), \quad (3)$$

where $p_j(n)$ is a polynomial in n ; $\exp(\alpha_j n) = \rho_j^n$ for $\rho_j = e^{\alpha_j}$.

The first model is more general; however, under some nonrestrictive conditions (e.g., if the time series has infinite length), these three models are equivalent; see for example, (Hall, 1998, Theorem 3.1.1; Golyandina et al., 2001, section 5.2).

Let us describe how the minimal LRR (the irreducible LRR of order that is as low as possible), which governs the time series, determines its explicit parametric form. A more convenient form of (3) is as follows:

$$s_n = \sum_{m=1}^p \left(\sum_{l=0}^{k_m-1} c_{ml} n^l \right) \mu_m^n, \quad (4)$$

where each μ_m coincides with $\rho_j e^{\pm i 2\pi \omega_j}$ for some j . Thus, the complex parameters $\{\mu_m\}$ in (4) determine frequencies $\{\omega_j\}$ and exponential bases $\{\rho_j\}$ in (3).

The polynomial $P_r(\mu) = \mu^r - \sum_{k=1}^r a_k \mu^{r-k}$ of order r is called *characteristic polynomial* of the LRR (2). The roots of the characteristic polynomial are called *characteristic roots* of the corresponding LRR. The roots of the characteristic polynomial of the minimal LRR governing the time series determine the values of the parameters μ_m and k_m in (4) as follows. Let a time series $S_\infty = (s_1, \dots, s_n, \dots)$ satisfy the LRR (2). Consider the characteristic polynomial of the LRR (2) and denote its different (complex) roots by μ_1, \dots, μ_p , where $p \leq r$. All these roots are nonzero as $a_r \neq 0$ with k_m being the multiplicity of the root μ_m ($1 \leq m \leq p$, $k_1 + \dots + k_p = r$). We refer for an extended summary to Section 3.3 and (Golyandina et al., 2018, section 2.1.2.2).

2.5.1 | Subspace-based approach

The question is how to find the structure (the basis of $\text{colspace}(S)$, the coefficients of the LRR, the parameters of s_n) of a signal S . Within SSA, the answer is as follows. Let us apply SSA and obtain the decomposition of the trajectory matrix $S = \sum_{m=1}^r \sqrt{\lambda_m} U_m V_m^T$.

Then

- The vectors U_m , $m = 1, \dots, r$, form a basis of the *signal subspace* $\text{colspace}(S)$.
- The basis U_m , $m = 1, \dots, r$, provides the coefficients a_k in the LRR $s_n = \sum_{k=1}^r a_k s_{n-k}$, $n = r+1, \dots$. Denote π_m the last coordinate of U_m , $\underline{U}_m \in \mathbb{R}^{L-1}$ the vector U_m with the last coordinate removed, and $\nu^2 = \sum_{m=1}^r \pi_m^2$. Then the elements of the vector

$$R = (a_{L-1}, \dots, a_1)^T = \frac{1}{1-\nu^2} \sum_{m=1}^r \pi_m \underline{U}_m \quad (5)$$

are the coefficients of the *min-norm* governing LRR: $s_n = \sum_{k=1}^{L-1} a_k s_{n-k}$ (see discussion of the min-norm LRR in Golyandina and Zhitljavsky (2013, section 3.2.3)). If $L = r+1$, (5) yields the minimal LRR, which is unique. Since LRRs are directly related to forecasting, see also Section 3.10.

- The basis U_m , $m = 1, \dots, r$, determined the values of α_j and ω_j in (3). Consider the signal in the complex-valued form $s_n = \sum_{k=1}^r c_k \mu_k^n$ (we simplify the form excluding polynomials). The relation between parameters is $\alpha = \ln(\text{Mod}(\mu))$ and $\omega = \text{Arg}(\mu)/(2\pi)$. Apply a subspace method, for example, ESPRIT (Roy & Kailath, 1989) (another name of ESPRIT for time series is Hankel SVD (HSVD); Barkhuijsen, de Beer, & van Ormondt, 1987). Denote $\mathbf{U}_r = [U_1 : \dots : U_r]$ and let $\underline{\mathbf{U}}_r$ be the matrix with the last row removed and $\overline{\mathbf{U}}_r$ be the matrix with the first row removed. Then μ_k can be found as the eigenvalues of the matrix $\underline{\mathbf{U}}_r^\dagger \overline{\mathbf{U}}_r$, where \dagger denotes pseudo-inversion. See Section 3.3 for more details about the parameter estimation.

This is called *subspace-based approach*. It can be extended to the 2D, 3D, ... cases; certainly, the n D theory for $n > 1$ is much more complicated than that for the one-dimensional case.

Subspace-based approach in real-world problems

In real-world problems we observe $X = S + R$, where S is the structured component of interest (e.g., a signal), R is a residual (e.g., noise).

Suppose we have an approximate separability of S and R . Then

1. Apply SSA and obtain the set of U_m , $m = 1, \dots, L$.

2. Identify the SVD components with numbers $G = \{i_1, \dots, i_r\}$, which are related to \mathbf{S} ; this is possible due to the assumed separability.
3. Take the set $\{U_i\}_{i \in G}$ as an estimate of the basis of the signal subspace. The same formulas from the subspace-based approach, which are used in the noiseless case, are applied to $\{U_i\}_{i \in G}$ to get estimates of the LRR coefficients and the time series parameters.

2.5.2 | Signal extraction via projections

Consider a particular case of $\mathbf{X} = \mathbf{S} + \mathbf{N}$, $\mathbf{X} = (x_1, \dots, x_N)$, where \mathbf{S} is a signal of rank r , \mathbf{N} is noise, and set the parameters: the window length L and the signal rank r .

Introduce two projections in Frobenius norm: $\Pi_r : \mathbb{R}^{L \times K} \mapsto \mathcal{M}_r$, where \mathcal{M}_r is the set of matrices of rank not larger than r , and $\Pi_{\mathcal{H}} : \mathbb{R}^{L \times K} \mapsto \mathcal{H}$, where \mathcal{H} is the set of Hankel matrices. Let \mathcal{T} be defined in (1).

Scheme of SSA for signal extraction:

$$\mathbf{X} \xrightarrow{\mathcal{T}} \mathbf{X} = \begin{pmatrix} x_1 & x_2 & \dots & x_K \\ x_2 & x_3 & \dots & x_{K+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & \dots & x_N \end{pmatrix} \xrightarrow{\text{SVD: } (\sqrt{\lambda_m}, U_m, V_m), \ \Pi_r} \xrightarrow{\mathcal{T}} \begin{pmatrix} \tilde{s}_1 & \tilde{s}_2 & \dots & \tilde{s}_K \\ \tilde{s}_2 & \tilde{s}_3 & \dots & \tilde{s}_{K+1} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{s}_L & \tilde{s}_{L+1} & \dots & \tilde{s}_N \end{pmatrix}$$

$\left\{ \begin{array}{l} \mathcal{L}_r = \text{span}(U_1, \dots, U_r) \\ \text{is the signal space;} \\ \Pi_r \text{ is the projector on } \mathcal{L}_r; \\ \hat{\mathbf{S}} = \sum_{m=1}^r U_m (\mathbf{X}^\top U_m)^\top = \Pi_r \mathbf{X}. \end{array} \right.$

$$\xrightarrow{\Pi_{\mathcal{H}}} \tilde{\mathbf{S}} = \begin{pmatrix} \tilde{s}_1 & \tilde{s}_2 & \dots & \tilde{s}_K \\ \tilde{s}_2 & \tilde{s}_3 & \dots & \tilde{s}_{K+1} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{s}_L & \tilde{s}_{L+1} & \dots & \tilde{s}_N \end{pmatrix} \xrightarrow{\mathcal{T}^{-1}} \tilde{\mathbf{S}}.$$

Thus, a concise form of the SSA algorithm for the signal extraction is

$$\tilde{\mathbf{S}} = \mathcal{T}^{-1} \Pi_{\mathcal{H}} \Pi_r \mathcal{T} \mathbf{X}.$$

2.5.3 | Example of modeling

Figure 6 depicts the decomposition of the time series “Fortified wines,” which was constructed without the use of a model. Nevertheless, this decomposition helps to detect the model and estimate its parameters. The model is a sum of products of polynomials, exponentials and sine waves. The subspace-based method allows one to construct the parametric model of the signal of rank r .

The subspace-based approach gives us the following form:

$$\begin{aligned} \tilde{s}_n = & C_1 0.99679^n + C_2 0.99409^n \sin(2\pi n/12 + \phi_2) + \\ & + C_3 1.00036^n \sin(2\pi n/4 + \phi_3) + C_4 1.00435^n \sin(2\pi n/5.97 + \phi_4) + \\ & + C_5 1.00175^n \sin(2\pi n/2.39 + \phi_5) + C_6 0.98878^n \sin(2\pi n/3.02 + \phi_6). \end{aligned}$$

The coefficients C_i and the phases ϕ_i can be estimated by the linear least-squares (LS) method. The details are described in Section 3.3; the R-code for the parameter estimation can be found in Fragments 3.5.9⁵ and 3.5.11⁶ of (Golyandina et al., 2018).

2.6 | Choice of parameters

The two parameters of SSA are the window length L and the way of grouping. There are no strict recommendations for their choice. Moreover, these recommendations differ for different problems and different assumptions about the time series structure.

For example, if the signal is of finite rank, then $L \approx N/2$ is recommended (simulations provide the recommendation $L \approx 0.4N$; Golyandina, 2010).

If the signal is not of finite rank or has a complex structure (a large rank and its trajectory has a large condition number), then a smaller window length is recommended.

If the period of a periodic component of the time series is known, it is recommended to take the window length divisible by the fundamental period.

For analysis of stationary processes, the recommendation can be special. In particular, the window length should allow a good estimation of the autocovariance matrix. This means that L should be small enough.

Formalization of the grouping way is difficult. One approach is based on the separability notion (e.g., on w -correlations as a measure of separability) and on visual inspection of ETs (Section 2.3.2). Another approach can be used in the case of finite-rank signals. Then the methods of rank detection can be applied (see a brief discussion in paragraph “Signal identification” of Section 3.14).

2.7 | Theoretical studies

Below we touch on several theoretical approaches to SSA. First, the theory of separability should be mentioned, where separability conditions are formulated and proved. Recall that the (approximate) separability of time series components means the ability of the method to construct the decomposition into identifiable components; in particular, the trend should be separable from the residual to be extracted by SSA (see Section 2.3.4). This theory was developed in Golyandina et al. (2001) and subsequently different modifications of SSA were proposed to improve separability (see Section 2.10.2).

However, an advanced technique is necessary to obtain theoretical results on the accuracy of reconstruction. A possible approach to this is to use the perturbation technique. Let us observe a perturbed signal $\mathbf{X} = (x_1, \dots, x_N) = \mathbf{S} + \delta\boldsymbol{\epsilon}$ of length N , where the time series $\delta\boldsymbol{\epsilon}$ is considered as a perturbation of the signal \mathbf{S} . The method results in $\tilde{\mathbf{S}} = \mathbf{S} + \Delta_N(\mathbf{S}, \delta, \boldsymbol{\epsilon})$; then $\Delta_N(\mathbf{S}, \delta, \boldsymbol{\epsilon})$ is expanded as $\Delta_N(\mathbf{S}, \delta, \boldsymbol{\epsilon}) = \Delta_N^{(1)}(\mathbf{S}, \boldsymbol{\epsilon})\delta + \Delta_N^{(2)}(\mathbf{S}, \boldsymbol{\epsilon})\delta^2 + \dots$. Most of theoretical results assume that δ is small; hence, the first-order error $\Delta_N^{(1)}(\mathbf{S}, \boldsymbol{\epsilon})$ is studied (Badeau, Richard, & David, 2008; Hassani, Xu, & Zhigljavsky, 2011; Vlassieva & Golyandina, 2009); this technique is in fact the linearization of the error in the neighborhood of \mathbf{S} . For simplicity, the behavior of the first order error (as $\delta \rightarrow 0$) is frequently considered as $N \rightarrow \infty$. This technique allows one to obtain results, which partly help to understand the behavior of the error. Nonetheless, this technique is still insufficient, since SSA works for any level of noise (i.e., for any value of δ) as $N \rightarrow \infty$. In the series of papers (Ivanova & Nekrutkin, 2019; Nekrutkin, 2010; Nekrutkin & Vasiliyev, 2017), a step ahead in the study of perturbations without the assumption about the smallness of δ is made; the obtained results are related to the case of nonrandom perturbation and thereby to the separability of signal components.

It is also worth noting the theoretical approach for the case of a fixed small L and stationary time series, where $\mathbf{X}\mathbf{X}^T/K$ tends to the $L \times L$ autocovariance matrix as N (and $K = N - L + 1$) tends to infinity (see, e.g., Van Huffel, 1993). Note that the choice of small L is inappropriate from the viewpoint of separability.

Another strand of theory is devoted to SSA-related methods in terms of the parametric model, low-rank approximations and subspace-based methods (Section 2.5).

2.8 | General scheme of SSA decompositions

We refer to Golyandina et al. (2018, section 1.1) for discussion of the general scheme of SSA modifications and variations in detail. A general form of the SSA-family algorithms for the object decomposition can be presented in the following form. A wide range of objects, from time series to n D shapes, can be considered as the input.

Input: An object (e.g., a time series).

1. **Embedding.** The input object is transformed into a structured trajectory matrix from a set \mathcal{H} by an embedding operator \mathcal{T} (e.g., \mathcal{H} is the set of Hankel matrices).
2. **Decomposition.** The trajectory matrix is decomposed into a sum of one-rank elementary matrices (e.g., by the SVD).
3. **Grouping.** The elementary matrices are grouped in an appropriate way; the grouped matrices are obtained by summation of the elementary matrices by the groups.
4. **Return to the object decomposition.** The grouped matrices are transformed to the form of the input object (e.g., from matrices to time series) by projecting to \mathcal{H} (e.g., by hankelization) and performing \mathcal{T}^{-1} .

Output: Decomposition of the input object into the sum of identifiable objects (e.g., of a trend, oscillations and noise).

The SSA algorithm in the general form is easily extended to analyzing objects of different dimensions by the change of the embedding operator \mathcal{T} . Also, modifications of the decomposition step do not depend on the shape/dimension of the decomposed object, since they are localized in the decomposition step, whereas the specific of the shape/dimension is localized by means of the embedding operator \mathcal{T} in the first and last steps.

2.9 | Multivariate/multidimensional extensions

Multivariate/multidimensional extensions differ by the embedding step of the SSA scheme; that is, by definition of the embedding operator \mathcal{T} . Let us list different versions of the embedding step:

- SSA for time series (1D-SSA): $\mathcal{T}(\mathbf{X})$ is a Hankel matrix;
- MSSA for systems of time series: $\mathcal{T}(\mathbf{X})$ is a stacked Hankel matrix;
- 2D-SSA for digital images: $\mathcal{T}(\mathbf{X})$ is a Hankel-block-Hankel matrix;
- Shaped SSA for any shaped objects: $\mathcal{T}(\mathbf{X})$ is a quasi-Hankel matrix;
- M shaped 3D-SSA for several 3D images of some shapes provides a stacked quasi-Hankel trajectory matrix;

Shaped n D-SSA ($n > 1$) can be considered as a general n -dimensional extension of SSA for shaped objects.

2.9.1 | Multivariate SSA

Decomposition, forecasting, missing data imputation, and other subspace-based methods are performed in the same manner as for 1D-SSA, see examples from Golyandina et al. (2018, chapter 4).⁷

Within the framework of SSA, we cannot speak about causality, since the MSSA method is invariant with respect to shifts in time of the time series. However, we can say about the supportiveness of one series with respect to another series. The supportiveness of the second time series with respect to the first time series means that the second time series improves the accuracy of signal estimation or forecasting in comparison with the use of the first series only. If the time series have a large portion of common structure, their simultaneous processing is better than the separate processing of each time series. The common structure within the SSA framework means similar signal subspaces. The accuracy of the signal extraction depends on the signal-noise ratio. Even if two time series have signals of the same structure, a large noise level in the second time series can cause this series to be not supportive.

It is important to stress that in MSSA there are a lot of different notations, which are sometimes controversial. In 1D-SSA, it is not important, what we call left or right singular vectors, eigenvectors or factor vectors/principal components, since the transposed L -trajectory matrix coincides with K -trajectory matrix for $K = N - L + 1$. Usually, the longer vectors are called factor vectors (principal components). Another approach is to fix the embedding dimension L (which is equal to the window length L) and to call the right singular vectors factor vectors/principal components. Then increasing the time series length N will increase the length K of factor vectors. Since usually $L \leq K$, both approaches provide the same terminology.

However, for MSSA, this is not the case. In MSSA, the trajectory matrix is constructed from the stacked trajectory matrices of time series from the considered collection. The stacking can be either vertical or horizontal. In the case of horizontal stacking,

$$\mathcal{T}_{\text{MSSA}}(\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(s)}) = [\mathbf{X}^{(1)} : \dots : \mathbf{X}^{(s)}],$$

where $\mathbf{X}^{(i)}$ is the trajectory (and therefore Hankel) matrix of the i th time series. Therefore, the left singular vectors correspond to time subseries of length L , while the right singular vectors consist of stacked subseries of different time series from the collection. Thus, in contrast to 1D-SSA, left and right singular vectors have different structure.

The vertical stacking is not something different; left and right singular vectors interchange (after interchanging L and $K = N - L + 1$); therefore, the way of stacking influences no more than terminology, choice of parameters and possibly computational costs.

Originally, a small window length L was applied to each of s time series with the vertical stacking of trajectory matrices (Weare & Nasstrom, 1982), where $L = 3$ (rows of the matrix \mathbf{X} were reordered in comparison with the conventional MSSA

algorithm with the vertical stacking); therefore, the length K of right singular vectors is large. The left singular vectors of length L_s were called EEOFs (extended empirical orthogonal functions). The right singular vectors produce factor vectors and principal components; the latter are equal to the factor vectors multiplied by singular values. Both the factor vectors and principal components have length close to the length of the time series themselves; therefore, they can be used instead of time series in further investigations.

In Weare and Nasstrom (1982), each time series corresponds to a space point on a 2D surface, that is, s is large. Each EEOF is divided into three parts corresponding to three time lags (the number of lags is equal to $L = 3$) and then each part is depicted as a surface. In Section 2.9.3 we discuss, how to describe the algorithm from Weare and Nasstrom (1982) as shaped 3D SSA; it seems this description is more natural. Note that the algorithm from Weare and Nasstrom (1982) (and all the algorithms which result in EEOFs) corresponds to Decomposition stage and does not contain Reconstruction stage.

The vertical stacking was also used in the paper (Broomhead & King, 1986b), which is usually considered as one of the first papers with the description of MSSA (its Decomposition stage). Although the original trajectory matrices were stacked horizontally in that paper, the authors considered eigenvectors of $\mathbf{X}^T \mathbf{X}$, not $\mathbf{X} \mathbf{X}^T$; that is, the SVD of the matrix \mathbf{X}^T was in fact considered, not of the \mathbf{X} itself. To make the difference between the horizontal and vertical variant more understandable, let us note the following: to build the SVD of the trajectory matrix for two time series $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ with trajectory matrices $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$, respectively, in the horizontal case, one calculates the eigenvectors of the matrix $\mathbf{X}^{(1)}(\mathbf{X}^{(1)})^T + \mathbf{X}^{(2)}(\mathbf{X}^{(2)})^T$, whereas in the vertical case one finds eigenvectors of the matrix

$$\begin{pmatrix} \mathbf{X}^{(1)}(\mathbf{X}^{(1)})^T & \mathbf{X}^{(1)}(\mathbf{X}^{(2)})^T \\ \mathbf{X}^{(2)}(\mathbf{X}^{(1)})^T & \mathbf{X}^{(2)}(\mathbf{X}^{(2)})^T \end{pmatrix}.$$

In Golyandina, Korobeynikov, Shlemov, and Usevich (2015) and Golyandina et al. (2018), the horizontal stacking is considered to fix the number of rows (i.e., to fix the dimension of the column space of the trajectory matrix). In the case of horizontal stacking, increasing the time series lengths leads to the addition of columns to the trajectory matrix. Moreover, the lengths of different time series can differ; this does not influence the column dimension.

The above consideration that the horizontal and vertical stackings yield just different forms of the same method is valid for basic (M)SSA, where the SVD of the trajectory matrix is considered. In particular, the form of stacking in MSSA, which was suggested in Broomhead and King (1986b), does not matter. For Toeplitz (M)SSA, this is not the case. Let \mathbf{C}_i be the autocovariance (Toeplitz) matrix of the i th time series, \mathbf{C}_{ij} be the cross-covariance matrix of the i th and j th time series, $\mathbf{C}_{ii} = \mathbf{C}_i$. Then for the horizontal stacking, the decomposition is constructed on the basis of the eigenvectors of $\sum_{i=1}^s \mathbf{C}_{ii}$.

For the vertical stacking, which is usually considered if MSSA stands for multichannel SSA, the decomposition is constructed on the basis of the eigenvectors of

$$\begin{pmatrix} \mathbf{C}_{11} & \dots & \mathbf{C}_{1s} \\ \dots & \dots & \dots \\ \mathbf{C}_{s1} & \dots & \mathbf{C}_{ss} \end{pmatrix}.$$

In climate investigation, the Toeplitz version of MSSA with vertical stacking is conventional (Plaut & Vautard, 1994). However, we were unable to find papers where the comparison of the vertical and horizontal versions of Toeplitz MSSA was carried out.

2.9.2 | Two dimensional SSA

For digital images given by a 2D array $\mathbf{X} = \mathbf{X}_{N_1, N_2} = (x_{ij})_{i,j=1}^{N_1, N_2}$ of size $N_1 \times N_2$, a 2D window size $L_1 \times L_2$ should be chosen. The trajectory matrix consists of vectorized moving 2D windows. This trajectory matrix can be written in the form of a Hankel-block-Hankel matrix:

$$\mathbf{X} = \mathcal{T}_{\text{2D-SSA}}(\mathbf{X}) = \begin{pmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{H}_3 & \dots & \mathbf{H}_{K_2} \\ \mathbf{H}_2 & \mathbf{H}_3 & \mathbf{H}_4 & \dots & \mathbf{H}_{K_2+1} \\ \mathbf{H}_3 & \mathbf{H}_4 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{H}_{L_2} & \mathbf{H}_{L_2+1} & \dots & \dots & \mathbf{H}_{N_2} \end{pmatrix}, \quad (6)$$

where each \mathbf{H}_j is the $L_1 \times K_1$ trajectory (Hankel) matrix constructed from $\mathbf{X}_{\cdot, j}$ (the j th column of the 2D array \mathbf{X}).

The estimation of frequencies by 2D-ESPRIT is the most frequent use of such trajectory matrix (Sahnoun, Usevich, & Comon, 2017), see Fragments 5.3.1⁸ and 5.3.2⁹ from Golyandina et al. (2018). Also, 2D-SSA is used for the problems of smoothing and noise reduction, see Fragments 5.4.2–5.4.6.¹⁰

2.9.3 | Shaped SSA

In Golyandina et al. (2015), a general approach to SSA is suggested. Shaped SSA is the universal version of SSA, which is applicable to arbitrary shapes and dimensions of the objects. The moving window can also be of arbitrary shape. For example, both the digital image and the moving window can be circular-shaped; that is, they are not necessary to be rectangular-shaped as in 2D-SSA. Shaped SSA allows one to consider different versions of the SSA algorithms in a unified manner: SSA for objects with missing data, 1D-SSA, MSSA, 2D-SSA and their circular versions among others. See Fragment 2.6.2¹¹ for the 1D case, Fragments 5.2.2–5.2.3¹² for the example with a shaped image and a circular-shaped window; and finally Fragment 5.4.6¹³ for the decomposition of data given on a cylinder (Golyandina et al., 2018).

There is a limitation of this approach, since points within the window and the window locations within the object should be linearly ordered. Being linearly ordered, the shaped windows are transformed into column vectors and these vectors are stacked to the trajectory matrix according to the ordered locations of the shaped windows. Many objects can be considered as linearly ordered. The standard technique is to consider the object as a subset of a multidimensional box of the same dimension as the object one, with natural ordering. For example, a piece of the sphere (after its projection to the plane) can be circumscribed by a rectangle; however, there is no continuous planar projection of the whole sphere. Therefore, at the present moment, SSA for data given on the whole sphere is not elaborated.

2.9.4 | Complex SSA

Complex SSA is Basic SSA applied to complex-valued time series. The only difference in the algorithm is the change of transpose to conjugate transpose. The algorithm of complex SSA was explicitly formulated in Keppenne and Lall (1996); although SSA was applied to complex time series much earlier, without considering the difference between real-valued and complex-valued data (see e.g., Tufts et al., 1982).

Generally speaking, complex SSA is not a method for analyzing multivariate time series; although, it can be considered as a special method for the analysis of two time series.

Most applications of complex SSA are related to the so-called F-xy eigenimage filtering (Trickett, 2003). This name is related to the analysis of digital images in geophysics; first, the discrete Fourier transform (DFT) is applied to each row of the image matrix, then complex-valued series are constructed from the results of the DFTs for each frequency, and finally, the constructed series are analyzed by complex SSA. Note that the authors of the papers devoted to F-xy eigenimage noise suppression usually omit the word “complex” in complex SSA. The specific of the studied geophysical images is that they are noisy and contain lines (traces); with the help of the DFT, these lines are transformed into a sum of complex exponentials, which have rank 1 in complex SSA. Therefore, the lines can be separated from noise very well.

2.10 | Modifications of the SVD step

Almost independently from the extensions of the embedding step of the SSA algorithm (Section 2.9), different modifications of the decomposition step can be considered in a common form $\mathbf{X} = \mathbf{X}_1 + \dots + \mathbf{X}_r$, where the matrices \mathbf{X}_i are some matrices of rank one.

In Basic SSA, the SVD expansion into rank-one matrices is considered. If there is no additional information about the time series, the SVD is optimal. Thus, possible modifications of the decomposition step are related to different assumptions about the time series.

2.10.1 | Use of a priori information

The following decompositions have been proposed taking into account the information about the structure of time series:

- *SSA with projection* is used if the model of the signal is partly known; for example, if it is assumed that the signal has a linear trend, see Section 3.5. We refer to Golyandina and Shlemov (2017) for details, where the general approach with preliminary projections is described. SSA with centering (Golyandina et al., 2001, section 1.7.1) is a particular case of SSA with projection, since the centering of a vector can be considered as the projection to the subspace spanned by the vector of ones. The version of SSA, where rows of the trajectory matrix are centered, came from PCA; however, such centering is not natural for time series, since the rows and columns are subseries of the original time series and therefore have the same structure. Double (both row and column) centering is much more natural for trajectory matrices. It is shown in Golyandina and Shlemov (2017) that SSA with double centering helps to extract linear trends in the presence of periodic components (see Fragment 2.8.7¹⁴).
- *Toeplitz SSA* is used if the time series is stationary (see Section 2.2.6). In the applications related to stationary time series, Toeplitz SSA is considered as a main version. Details are described in Section 3.13.

2.10.2 | Refined decompositions of signals

Another reason to modify the SVD step is related to improving the separability. The two main properties of the SVD (the biorthogonality and the optimality) allow Basic SSA to decompose a time series into identifiable components. The optimality (we mean the approximation properties of the SVD) provides the possibility to separate signals from noise. The biorthogonality helps to separate time series components if they are (approximately) orthogonal. It turns out that many time series components such as trends and sine waves with different frequencies are asymptotically orthogonal (separable) as the time series length tends to infinity; however, we need the separability for time series of finite length. Moreover, the problem of the lack of strong separability (the problem of equal eigenvalues in the SVD) cannot be solved even asymptotically.

There are a lot of methods for matrix decompositions, which are not necessarily biorthogonal. However, most of the methods for a decomposition into rank-one matrices, which are not biorthogonal, simultaneously drop the approximation property. Therefore, the improvement of separability is performed in two steps. First, the trajectory space of the signal is estimated by means of grouping the signal elementary matrix components in the SVD step of Basic SSA. Then, the signal grouped matrix is decomposed by a refined expansion (we call such kind of refined expansions nested decompositions).

The examples of SSA modifications with nested decompositions are:

- SSA with derivatives. This modification can help if the components of the signal were mixed due to equal contributions of components (i.e., in the case, when the weak separability takes place but there is no strong separability) (Golyandina & Shlemov, 2015). The application of SSA with derivatives is demonstrated in Fragment 2.8.6¹⁵ for the 1D case. Fragment 5.2.5¹⁶ shows that the same approach works in the 2D case.
- Iterated oblique SSA. This version helps to solve the problem of no orthogonality (no weak separability; Golyandina & Shlemov, 2015); see, for example, Fragment 2.8.2.¹⁷

An approach, which uses additional rotations of mixed components, is also considered in Groth and Ghil (2011), where multichannel SSA with factor rotations is suggested to solve the problem of the lack of strong separability in MSSA. Recall that this problem is caused by equal eigenvalues in the SVD decomposition of the trajectory matrix. In the paper (Groth & Ghil, 2011), the problem of equal eigenvalues is called degeneracy of eigenvalues (this is the notion, which is sometimes used in PCA for the explanation of unstable eigenvectors for close eigenvalues). Formulation of the studied problem in terms of the strong separability may help to gain new insights.

2.10.3 | Tensor SSA

The general scheme of SSA (Section 2.8) can be further extended by means of considering the embedding operator \mathcal{T} , which maps the set of objects to the set of tensors of some order instead of the matrix set, and then using tensor decomposition instead of matrix decomposition in the decomposition step.

The shortcoming is that tensor decompositions are generally not unique and their numerical calculation is time-consuming. Sometimes, tensor decompositions can be reduced to simpler matrix decompositions. For example, the SVD of the trajectory matrix in 2D-SSA is in fact the Kronecker-product SVD of a tensor of fourth order (Golyandina & Usevich, 2010).

New algorithms can be obtained by tensor decompositions that are not reduced to matrix ones. For example, there is a version of SSA where the embedding operator transfers the initial object to a 3D array instead of a matrix (a 2D array); then the tensor decomposition (PARAFAC) is performed (see, e.g., Kouchaki, Sanei, Arbon, & Dijk, 2015; Yang et al., 2017). Note that the idea of the use of tensor decompositions arose much earlier in the subspace-based and low-rank approximation framework (Papy, De Lathauwer, & Van Huffel, 2005; see also Sidiropoulos et al., 2017).

3 | SSA AND DIFFERENT PROBLEMS

3.1 | SSA and nonlinearity. Is SSA a linear method?

In the literature, a criticism of SSA sounds as “SSA is a linear method”.

Let us explain what this means. First, note that the algorithm is nonlinear. Components, which can be extracted by SSA, are generally nonlinear. However, the class of time series, which produce rank-deficient trajectory matrices, consists of time series governed by homogeneous LRRs in the form $x_n = \sum_{i=1}^r a_i x_{n-i}$. LRRs are closely related to linear differential equations (LDEs) $\sum_k b_k s^{(k)}(t) = 0$, where $s^{(k)}(t)$ is the k th derivative of $s(t)$, since LRRs are generated by the finite-difference method applied for solving LDEs. In the theory of dynamical systems, the methods, which are related to LDEs, are called linear. For example, let $s(t) + bs'(t) = 0$, $t \in [0, \infty)$, be a LDE of order 1; consider $i = 1, 2, 3, \dots$ as the discretization of t and the finite-difference approximation $s'(i) \approx s_{i+1} - s_i$; then we obtain $s_i + b(s_{i+1} - s_i) = 0$ or, the same, $s_{i+1} = (1 - 1/b)s_i$, that is, the LRR of order 1.

The answer to the criticism of SSA linearity is as follows.

First, the class of solutions of LDEs is wide enough, since it contains any finite sums of products of polynomials, exponentials and sine waves. Then, to deal with signals governed by LRRs (e.g., to extract signals from noisy time series), a large window length (optimally, in the range from $N/3$ to $N/2$, where N is the time series length) is recommended; however, smaller windows allow taking into consideration only local finite-rank approximations. For example, a modulated sinusoid with a slowly varying amplitude is well approximated by a sinusoid with the same frequency on time intervals of size equal to several periods. Therefore, SSA with a small window length equal to several periods produces reasonable results for this example. It can be argued that for extraction by SSA, the time series component of interest should be locally governed by the same LRR at each local segment. Thus, small window lengths allow one to deal with nonlinearity of the model in some common cases, such as modulated harmonics and trends.

A very important feature of SSA is that the method does not use the explicit parametric form (3) of the time series. For example, to predict the value of an exponential series $s_n = Ae^{\alpha n}$, one approach is to estimate A and α and then to perform the forecasting by the explicit formula. However, a more flexible approach is to estimate coefficients of the governing LRR and to perform forecasting by the estimated LRR. Thus, many SSA-family methods try to avoid the estimation of parameters in (3) to make the method more robust with respect to deviations from the model.

Whereas a slowly changing amplitude of the sinusoidal signal is admissible, a changing frequency is not appropriate for SSA applied to the whole time series (except in the case of varying the frequency around one main value; this is called “phase noise”). In the case of changing frequency, methods like empirical mode decomposition (EMD) + HHT (Huang & Wu, 2008) can be used for the extraction of oscillations with changing frequency; however, they only work well if noise is small enough.

Local SSA

The standard approach to analyzing signals with a changing structure is to consider moving subseries (segments) of the original time series. In the framework of SSA, this procedure is called subspace tracking (see, e.g., Badeau, Richard, & David, 2003). In the papers about subspace tracking, the primary focus is on the construction of fast algorithms. The other use of subspace tracking is the change-point detection (Golyandina et al., 2001, chapter 3; Moskvina & Zhigljavsky, 2003), see example in Fragment 3.5.12.¹⁸

For the estimation of time series components such as trends and signals, SSA applied to moving subseries is used (see, e.g., Leles, Sansao, Mozelli, & Guimaraes, 2018). The main problem here is how to combine different local decompositions into one global decomposition. In Leles et al. (2018), central parts of the local signal reconstructions are used and then stacked. The idea to use only the central part is promising, since the reconstruction of end points is less accurate. However, the question is why not to use only one central point (by analogy with the LOESS method; Cleveland, 1979); certainly, this can be done if the computational cost is acceptable.

As the result of local SSA, we obtain a signal estimation or just a smoothing. The problem is how to forecast the extracted signal, since its local estimates may have different structures on different time intervals. In local versions of SSA, we do not obtain a common nonlinear model; we have several stacked linear models. However, many nonparametric local methods including for example, LOESS have this drawback.

SSA as linear filter

Another reason to call SSA a linear method is its connection with linear filters (see Section 2.4 for references). However, it should be noted that the coefficients of the SSA linear filters are produced by the time series in a nonlinear way.

3.2 | SSA and autoregressive processes

It is important to remark that the basic model of signals in SSA and the model of autoregressive (AR) processes are similar only at a superficial glance. The fact is that the models are totally different.

For SSA, the signal model is $s_n = \sum_{i=1}^r a_i s_{n-i}$ and the observed series is a noisy signal, which has the form $x_n = s_n + \epsilon_n$, where ϵ_n is typically noise (a nonregular oscillation; for example, a realization of a stationary random process).

For AR processes, we have noise innovations at each step: $x_n = \sum_{i=1}^r a_i x_{n-i} + \epsilon_n$. Under some conditions on the coefficients, such innovations yield that x_n , $n = 1, 2, \dots$, is a stationary stochastic process, while the series with terms $s_n = \sum_{i=1}^r a_i s_{n-i}$ is a deterministic and not necessarily stationary signal. The coefficients $\{a_i\}_{i=1}^r$ in the SSA model can be arbitrary.

In both AR and SSA analysis, the characteristic polynomials are constructed on the basis of coefficients a_i , $i = 1, \dots, r$. In terms of SSA, we are concerned about characteristic roots of the governing LRR, that is, about the roots of the characteristic polynomial $\mu^r - a_1\mu^{r-1} - \dots - a_r$ (see Section 2.5). The roots, which have moduli larger/smaller than 1, correspond to a growing/damped time series components, whereas roots with unit moduli correspond to a stationary deterministic component like undamped sine waves (see, e.g., Golyandina et al., 2001, Example 5.10).

For the AR model, the AR characteristic polynomial is reciprocal to that in SSA (the coefficients are taken in reverse order); therefore, its roots are inverse to the roots of the characteristic polynomial in SSA with the same coefficients a_i , $i = 1, \dots, r$. The stationarity in the AR model corresponds to the case when all the roots of the AR characteristic polynomial have moduli larger than 1. The case of roots with unit moduli corresponds to nonstationarity (an example is Brownian motion $x_n = x_{n-1} + \epsilon_n$ and its characteristic polynomial $\mu - 1$; the same characteristic polynomial corresponds to a constant signal in SSA).

In the context of SSA used for signal extraction, AR is mostly considered as a noise model (see, e.g., a discussion of Monte Carlo SSA in Section 3.11).

There is a common form for forecasts of signals governed by LRRs and for conditional mean forecasts of AR processes. Both are performed by the LRR with estimated coefficients. The difference is in the approach for estimating these coefficients. Note that the forecasting values in the AR model are always converging to zero (or the mean value).

Let us remark that in Keppenne and Ghil (1992) a hybrid of SSA and AR was proposed: the AR model was used for forecasting leading elementary RCs obtained by the Toeplitz (VG) version of SSA. In that paper, the proposed forecasting method was applied to climate data, which were modeled as stationary random processes.

There is one more connection between SSA and AR. For AR processes, SSA can be applied to an estimate of the autocovariance function $C(l)$ to find the AR coefficients, since the series $C(l)$ is governed by the LRR with the same coefficients: $C(l) = \sum_{i=1}^r a_i C(l-i)$ (the Yule-Walker equations).

3.3 | SSA and parameter estimation

In this section, we consider the problem of parameter estimation in the model described in Section 2.5.

Let the signal have the explicit parametric form of a finite sum

$$s_n = \sum_j A_j \exp(\alpha_j n) \sin(2\pi\omega_j n + \phi_j) \quad (7)$$

and the observed time series be $x_n = s_n + r_n$. Here we simplify the general model (3), where a polynomial multiplier can be presented in each summand. The complex-valued form of (7) is

$$s_n = \sum_{m=1}^r c_m \mu_m^n, \quad (8)$$

$\mu_m = \rho_m e^{i2\pi\omega_m}$. Here $\omega_m = \text{Arg}(\mu_m)/(2\pi)$, $\rho_m = |\mu_m|$. Recall that μ_m are the roots of the characteristic polynomial of the minimal LRR governing the signal. If s_n are real, then each complex root μ_m should have its complex conjugate $\mu_k = \mu_m^*$. Therefore, $c_m \mu_m^n + c_k \mu_k^n$ can be made real by a suitable choice of complex coefficients c_m and c_k .

A more general signal model is given by the signal rank: $\text{rank } S = r$, where r is known in advance or is estimated. In this model, we consider $s_n = A_1 e^{\alpha_1 n} + A_2 e^{\alpha_2 n}$ and $s_n = A e^{\alpha n} \cos(2\pi\omega n + \phi)$ belonging to the same model for $r = 2$, whereas for the explicit representation (7), these are two different parametric models.

The dependence on the parameters α_j (ρ_j) and ω_j is nonlinear; the dependence on A_j and ϕ_j can be considered as linear, since $A \cos(2\pi\omega n + \phi) = C_1 \cos(2\pi\omega n) + C_2 \sin(2\pi\omega n)$ for some C_1 and C_2 depending on A and ϕ .

The Cramér–Rao lower bounds (CRLB) for the variance of parameter estimates are known (see, e.g., Badeau, David, & Richard, 2008; Stoica & Moses, 2005). For the case of undamped sinusoids, the CRLB for the estimate of the frequency has order $1/N^3$, while the CRLB for the estimate of the amplitude has order $1/N$.

The common approach for parameter estimation is the nonlinear least-squares method, which implies the explicit parametric form:

$$\sum_{n=1}^N (x_n - s_n(\{\alpha_j, \omega_j, A_j, \phi_j\}))^2 \rightarrow \min_{\{\alpha_j, \omega_j, A_j, \phi_j\}}.$$

It is a very complicated time-consuming optimization problem, since the objective function has many local minima; the problem is nonlinear and therefore needs iterative methods for solution.

The second approach is called *subspace-based* (Section 2.5.1). Let us consider the complex-valued form (8). The nonlinear parameters μ_m are estimated by one of the subspace-based methods (e.g., ESPRIT), which are based on U_1, \dots, U_r obtained in Decomposition stage of SSA (Section 2.2.3). The linear parameters can be found by the conventional linear least-squares method. Consider the Vandermonde matrix generated by μ_m , $m = 1, \dots, r$:

$$\mathbf{M} = \begin{bmatrix} \mu_1 & \mu_1^2 & \dots & \mu_1^N \\ \mu_2 & \mu_2^2 & \dots & \mu_2^N \\ \vdots & \vdots & \ddots & \vdots \\ \mu_r & \mu_r^2 & \dots & \mu_r^N \end{bmatrix}.$$

Then (8) has the form $S = (s_1, \dots, s_N) = C^T \mathbf{M}$, where $C = (c_1, \dots, c_r)^T$. If s_n and μ_m are estimated (the signal S is estimated as the reconstructed time series \tilde{S} obtained by SSA; μ_m are estimated as $\tilde{\mu}_m$ obtained by ESPRIT), we come to the approximate equality $\tilde{S} \approx C^T \tilde{\mathbf{M}}$; then the estimate \hat{C} can be found by the LS method.

One can see that the described approach is very simple and all we should know about the signal is its rank. As methods for the frequency estimation, the subspace-based algorithms are called high-resolution, since they provide the frequency estimates with the variance of the same order $1/N^3$ as the CRLB has; see Badeau, Richard, et al. (2008) for the undamped case: for $s_n = C e^{i2\pi\omega n}$, the ESPRIT estimate of ω has variance $D \hat{\omega} \sim 1/N^3$ if the window length L is proportional to the time series length N .

3.4 | SSA and SLRA

Let us apply SSA to the problem of extracting a finite-rank signal from a noisy time series. The algorithm of SSA can be written down in a compact form by means of projection operators (see Section 2.5.2): $\tilde{\mathbf{S}} = \mathcal{T}^{-1} \Pi_{\mathcal{H}} \Pi_r \mathcal{T} \mathbf{X}$.

However, in practice, the estimate $\tilde{\mathbf{S}}$ is generally not of finite rank. The problem of finding the estimate $\tilde{\mathbf{S}}$ of rank r can be solved by different methods. There is a subset of methods, called SLRA, which state the problem of approximation of the time series trajectory matrix by a low-rank Hankel matrix: $\min_{\mathbf{S} \in \mathcal{M}, \cap \mathcal{H}} \|\mathbf{X} - \mathbf{S}\|_F$. The most famous method is called Cadzow iterations and was introduced in Cadzow (1988). This method consists of alternating projections and can be expressed as $\mathbf{S}^{(m)} = \mathcal{T}^{-1} (\Pi_{\mathcal{H}} \Pi_r)^m \mathcal{T} \mathbf{X}$. Thus, SSA as a method for signal extraction can be considered as the first iteration of the Cadzow iterations. Note that the Cadzow iterations were suggested in parallel with SSA.

Another approach is to use a particular parameterization of the set \mathcal{D}_r of time series of rank not larger than r . Then the problem of low-rank approximation is considered as the least-squares problem

$$\min_{\mathbf{S} \in \mathcal{D}_r} \|\mathbf{X} - \mathbf{S}\|_w^2 \quad (9)$$

and can be solved by the weighted least-squares (WLS) method. If the noise is Gaussian, then by the choice of corresponding weights, the weighted LS estimates coincide with maximum likelihood estimates (MLE) and therefore they are asymptotically the best.

The problem (9) is still called SLRA method (see Markovsky, 2019), where the Hankel structure of trajectory matrices is considered. However, the problem (9) does not depend on the window length, which determines the dimensions of the trajectory matrix. Therefore, in fact, this is not the problem of low-rank matrix approximation (Zvonarev & Golyandina, 2018).

Both SSA and Hankel SLRA can be used for signal extraction, forecasting and frequency estimation, since both methods provide estimates of the signal subspace and the signal itself. Comparing the approaches, we can say that

- SSA is fast, SLRA is time-consuming;
- SLRA can provide more accurate parameter estimates in comparison with SSA;
- for a signal which only approximately (or locally) satisfies the model, SLRA does not work; SSA does work;
- the outcome of SLRA allows simpler procedures for parameter estimation and forecasting.

Thus, SLRA is applied mostly in signal processing for engineering problems, where the signals are exactly of finite rank. In the general case of real-world time series, the presence of time series components which are exactly of finite rank is a rare case; therefore, SSA is suitable in a much greater extent. For example, slowly varying trends can be approximated by time series of finite rank but generally they are not exactly of finite rank. Also, amplitude modulations of periodic components are not exactly of finite rank for real-world data. However, this does not create a barrier for SSA to extract them. Nevertheless, a reasonable example of Cadzow iterations for extraction of an exponential trend can be found in Fragment 3.5.8.¹⁹

3.5 | SSA and linear regression (LS-estimator of linear trend)

As we discussed before, SSA can extract trends, that is, slowly-varying time series components, since they can be approximated by finite-rank time series.

Ideally (but not necessarily), trends should be series of small rank r . Linear series with terms $s_n = an + b$, $a \neq 0$, $n = 1, \dots, N$, belong to the class of time series, which are governed by LRRs, and have rank 2. However, linear functions are not natural for SSA (while exponential series are very natural). The reason is that the characteristic root for a linear series is 1 of multiplicity 2 and the minimal linear recurrence formula is $s_{n+2} = 2s_{n+1} - s_n$ (therefore, the characteristic polynomial is $\mu^2 - 2\mu + 1$). Presence of roots of multiplicity larger than 1 is very unstable. Any distortion of the coefficients of this LRR transforms the multiple unit root to two different roots, that is, a linear series to a sum of two exponentials with small exponential rates or a sinusoid with large period.

There is a modification of SSA, which was called in Golyandina et al. (2001, section 1.7.1) SSA with double centering. In that book, the correspondence between SSA with double centering and the extraction of linear trends is demonstrated. As it is shown in Golyandina and Shlemov (2017), SSA with double centering is a particular case of SSA with projection, where projections of the rows and columns of the trajectory matrix to given subspaces are produced, subtracted from the trajectory matrix, and then the SVD expansion of the residual matrix is performed. SSA with projection is positioned as SSA with the

use of some information given in advance. At the present moment, only the use of SSA with projection for the extraction of polynomial trends is analyzed.

The most common approach for estimating linear trends is the linear regression method, where the least-squares solution is used. Let us summarize the results of the comparison between SSA with double centering and linear regression. Certainly, if the time series consists of a linear trend and white noise $x_n = an + b + \epsilon_n$, the least-squares method provides the best estimate of the trend. However, if the residual includes, for example, a periodic component, this is not true. It is numerically shown in Golyandina and Shlemov (2017) that for the case $x_n = an + b + \sin(2\pi\omega n + \phi) + \epsilon_n$, where ϵ_n is white noise, the LS method applied to the trend, which was obtained by SSA with double centering, generally overcomes the conventional LS estimate applied to the original time series (i.e., overcomes the ordinary linear regression).

3.6 | SSA and filtering

Linear finite-impulse response (FIR) filters are defined as $f_n(X_\infty) = \sum_{i=-m_1}^{m_2} b_i x_{n-i}$, where $X_\infty = (\dots, x_{-1}, x_0, x_1, \dots)$ is the input time series. The main characteristic of FIR filters is the frequency response $A(\omega)$, which has a simple explanation: if $x_n = \cos(2\pi\omega n)$, then $f_n(X_\infty) = A(\omega)\cos(2\pi\omega n + \phi(\omega))$.

SSA can be considered as a set of adaptive filters (see Section 2.4 for references). For the window length L , each elementary RC for the points with numbers from L to $N - L + 1$ is obtained by a linear FIR filter applied to the original time series $X = (x_1, \dots, x_N)$. If the window length is small, almost all points of the SSA output can be considered as a result of filtering by an adaptive FIR filter. If L is large, this is not the case. Actually, each point of the reconstructed series is a linear combination of values of the original series. However, for each point with number from $[1, L]$ and $[N - L + 1, N]$ we have different linear combinations.

The following result is valid: let SSA be applied with $L \leq (N + 1)/2$ and $U = U_m$ be the m th eigenvector in the SVD of the trajectory matrix X ; $m = 1, \dots, d$. Then the m th elementary reconstructed time series on the interval $[L, N - L + 1]$ has the form

$$\tilde{x}_n^{(m)} = \sum_{j=-(L-1)}^{L-1} \left(\sum_{k=1}^{L-|j|} u_k u_{k+|j|}/L \right) x_{n-j}, \quad L \leq n \leq N - L + 1. \quad (10)$$

This filter is called middle-point filter. It is shown in Golyandina and Zhigljavsky (2013, section 3.9) that the frequency response $A(\omega)$ of the middle-point filter is determined by the periodogram of U . This explains why we can perform the grouping step based on frequency characteristics of the eigenvectors U_m , $m = 1, \dots, d$.

Note that if U consisted of equal numbers, we would obtain the Bartlett (triangle) filter. The leading eigenvector has vector components close to a constant if the time series is positive and L is small enough (the same sign of the eigenvector components follows from the Perron–Frobenius theorem).

The other helpful property is as follows. The filter bandwidth tends to be narrower as L (together with N) increases. That is, by increasing the window length L we can obtain a more refined decomposition. Recall that we discuss the behavior of RCs at $[L, N - L + 1]$.

The so-called last-point filter plays an important role in SSA, since it is used for reconstruction of the last point and therefore related to the prediction of the time series. It is not really a filter, since it is used for reconstructing one point:

$$\tilde{x}_N^{(m)} = u_L \sum_{i=0}^{L-1} u_{i+1} x_{N-i}.$$

However, it is the only reconstruction filter that is causal (see Golyandina & Zhigljavsky, 2013, section 3.9.5 for discussions).

As a rule, causal filters lead to a delay. For example, the causal moving average with the window width W has delay $W/2$. The delay in SSA depends on the separability quality. If the time series component of interest is exactly separable, then the delay is zero. This is an important advantage of SSA over the moving averaging. This advantage is a consequence of the adaptive properties of SSA.

3.7 | SSA and ICA

Independent component analysis (ICA) is introduced for random processes and the word “independent” is related to using the stochastic independence instead of the uncorrelatedness in PCA that is based on the orthogonality in the SVD. In SSA, mostly nonrandom signals are considered. Therefore, the direct use of the ICA approach in the SSA algorithm is not appropriate. In ICA, different measures of stochastic independence are considered. Once the measure of independence is fixed, the independent components are found by solving the corresponding maximization problem. The decomposition step in SSA can be modified in such a manner that the ICA optimization problem is solved instead of the SVD and thereby the separability of the signal components can be improved. By the reasons explained in Section 2.10.2, ICA is used in SSA for a nested decomposition.

SSA with the SOBI-AMUSE version of ICA is described in Golyandina and Lomtev (2016). Note that this version is very similar to the modification of SSA called SSA with derivatives (Golyandina & Shlemov, 2015), which was created using a completely different approach. The version of SSA with maximization of the entropy is described in Golyandina and Zhigljavsky (2013, section 2.5.4).

Another connection between SSA and ICA is related to applications to blind signal separation and is considered in Pietilä, El-Segaeir, Vigário, and Pesonen (2006). In that application, basic SSA is used for preprocessing, that is, for removal of noise and for dimension reduction; then ICA is applied for the extraction of independent components from their mixture in a conventional way. This is similar to the use of PCA before ICA for analysis of multidimensional data (Kato, Chen, & Xu, 2006).

3.8 | SSA and EMD, DFT, DWT

Empirical mode decomposition

EMD (Huang & Wu, 2008) is frequently compared with SSA, since both are model-free techniques. It seems that EMD is a method without explicit approximation properties, whereas SSA has both separability and approximation properties. The first components extracted by EMD is highly oscillating and the last component is referred to a trend; for SSA, it is the opposite, since the signal and the trend typically correspond to the leading components of the decomposition; this is an advantage of SSA as a decomposition method. The advantage of EMD is its ability to extract periodic components with complex amplitude and frequency modulations. It is likely that the combination of SSA and EMD can extend the range of real-world problems being solved.

Discrete Fourier transform

DFT differs from SSA by the use of a fixed basis consisting of sines–cosines with frequencies from an equidistant grid against the construction of an adaptive basis in SSA. In Bozzo et al. (2010) the relation between SSA and DFT is discussed. Note that SSA in fact coincides with DFT in the circular version of SSA (see terminology in Shlemov & Golyandina, 2014), where the data are considered given on a circle in the 1D case or on a torus in the 2D case.

From the viewpoint of frequency estimation (see Section 3.3), SSA and the related subspace-based methods allow one to estimate frequencies with a better resolution than $1/N$, where N is the time series length (see, e.g., Santamaría, Pantaleón, & Ibáñez, 2000; Stoica & Soderstrom, 1991 for comparisons of DFT, ESPRIT and MUSIC). The MUSIC method allows one to construct pseudo-spectrums similar to periodograms but with no limitation on the frequencies set. Comparing time series models that are suitable for the methods, one can say that a sum of pure sinusoids corresponds to DFT, while a sum of exponentially modulated sinusoids corresponds to SSA.

Another application of DFT is the estimation of the spectral density by means of smoothing the periodograms. SSA can be used for estimating the spectral density (see, e.g., Golyandina et al., 2001, section 6.4.3), where the results from Grenander and Szegö (1984) in terms of SSA are discussed. If the spectral density is monotonic, different eigenvectors generally correspond to different frequencies; otherwise, the eigenvectors are mixed, that is, they may include different frequencies with comparable contributions. This is an explanation why most of the eigenvectors produced by white noise (which has a constant spectral density) are irregularly oscillating. And, vice versa, the eigenvectors generated by red noise (the AR process of order one with a positive coefficient) correspond to distinguishing frequencies.

In Yiou, Baert, and Loutre (1996), the application of SSA to spectral estimation in climatology is discussed.

Discrete wavelet transform

DWT is the decomposition based on a fixed space–time basis. This yields both advantages and disadvantages in comparison with SSA. See discussion in Yiou et al. (2000).

3.9 | SSA: Model-free method and modeling

In Section 2.5 we discussed the model of time series that suits SSA. These are time series of finite rank or, almost the same class, time series governed by LRRs. The latter class is slightly narrower; although it is much more understandable in practice.

SSA is a multipurpose method, which can be whether model-free or used for modeling (this is a distinguished feature of SSA). Briefly:

1. As an exploratory method, SSA is a model-free technique, which can perform the time series decomposition and the frequency filtering without assumptions given in advance.
2. If the signal is governed by an LRR, SSA allows one to obtain the explicit form of the signal and to estimate the parameters; that is, SSA is able to perform the parametric modeling.
3. If the signal satisfies an LRR only approximately (or locally), the forecasting and missing data imputation can be performed in the framework of SSA without constructing the explicit parametric model; that is, SSA is an adaptable method. This is one of the key advantages of SSA, which considerably extends the range of applications.

3.10 | SSA: forecasting and gap filling

In this section, we put the forecasting and gap filling (missing data imputation) together, since the forecasting can be considered as a particular case of the gap filling with artificial gaps at the place of the predicted data. On the other hand, the gap filling can be considered as forecasting internal data.

Parameters finding via cross-validation

In general, if an algorithm has parameters, their choice can be performed with the help of the cross-validation procedure, which consists of constructing the prediction on the training data and then calculation of errors on the test data; the parameters are chosen to minimize the cross-validation error.

For forecasting, a moving prediction is performed for the cross-validation, where the test sets step after the training sets. For gap filling, artificial gaps, which are located at arbitrary positions, are considered as the test data. In general, errors of imputations of the test data can be considered for the choice of forecasting parameters; however, the forecasting accuracy can significantly differ from the imputation accuracy, since for the imputation we have data from both sides, while the prediction uses data from one side only and thereby is less stable. It is an important distinction, since we are looking for a tradeoff between accuracy and stability, which can be different for prediction and interpolation.

SSA forecasting

Typically, the model of time series, which is a sum of a signal of finite rank and noise, is considered for forecasting. However, the prediction of signal components (e.g., a trend) is possible as well. Also, the signal or its components should not be exactly of finite rank. Several subspace-based forecasting methods are suggested in Golyandina et al. (2001) and Golyandina and Zhitljavsky (2013): recurrent, vector and simultaneous ones. The first two versions of the SSA forecasting are more frequently used. If the signal is exactly of finite rank, all three methods give the same result.

After the decomposition step of the SSA algorithm has been performed, the elementary time series components can be identified and grouped. Each group consisting of signal components induces a subspace spanned by the eigenvectors from the corresponding group; then the subspace approach (see Section 2.5.1) is applied to each group separately. The subspace-based methods are fast and work well for forecasting in many cases, when SSA is appropriate. It is convenient that one should not choose the number of elementary components in advance.

Let us comment the SSA recurrent prediction. The recurrent forecasting is performed by a special LRR (the min-norm LRR, whose vector of coefficients has a minimal norm), which approximately governs the signal (or the chosen signal component). This algorithm is known as the linear prediction algorithm (Kumaresan & Tufts, 1982), where the min-norm LRR is used for forecasting.

Let us explain the speciality of the min-norm LRR. There are a lot of LRRs, which govern the same time series, that is, $s_n = \sum_{i=1}^{\tau} b_i s_{n-i}$, for various vectors of coefficients $B = (b_1, \dots, b_{\tau})^T$; however, these LRRs have different suppressing properties: $D\hat{s}_n = D\sum_{i=1}^{\tau} b_i (s_{n-i} + \delta_{n-i}) = \|B\|^2 \sigma^2$, where δ_k , $k = 1, \dots, N$, are white noise, $\sigma^2 = D\delta_k$. If $\tau = L - 1$ is fixed, the SSA forecasting LRR (5) has the minimum norm of the vector of coefficients and therefore has the best suppressing properties.

Examples of forecasting are shown in Fragment 3.2.1.²⁰

Subspace-based missing-data imputation

The subspace-based approach, which is used for forecasting, can be considered for imputation of missing values as well (Golyandina & Osipov, 2007). Generally, if a prediction algorithm works well for the studied time series, the same algorithm can be applied to fill a gap by forward prediction from the left, by backward prediction from the right and then by a combination of the results.

Subspace-based methods work well if the signal subspace can be estimated with sufficient accuracy. In terms of SSA, this means the necessity of approximate separability of the signal, which we want to analyze, from the residual.

For gap filling, the shaped version of SSA (see Section 2.9.3) is used to obtain the SSA decomposition. Shaped SSA which is applied to time series with gaps limits the choice of the window length, since a sufficient number of windows of the chosen length should be located at the places with no gaps and thereby the separability which needs a large window length can be worsen. Therefore, the subspace-based methods for gap filling are suitable only for a small number of gaps; generally, for several compactly located sets of missing values to allow one to estimate the subspace (Fragment 3.3.1²¹).

Iterative gap filling

In Kondrashov and Ghil (2006), the approach from Beckers and Rixen (2003) was applied to time series. The approach suggested in Beckers and Rixen (2003) is very general and can be applied to data in different forms. The algorithm is iterative and has two parameters (for SSA), the window length L and the rank r . In the first step, the missing data are filled in by some numbers; e.g., by the average value. Then $\text{SSA}(L, r)$ is applied to the obtained time series, which has no missing data, for calculating the reconstructed series. Next, the values at the positions, which initially were with no missing entries, are changed to the original values; $\text{SSA}(L, r)$ is applied to the obtained time series, and so on. Thus, we have repeated iterations. Cross-validation, where artificial gaps serve as the test set, can be used to choose L and r .

The iterative method is time consuming; however, this is a universal approach for missing data imputation, since it is applicable for arbitrary gap locations (see Fragments 3.3.2 and 3.3.3²²). For forecasting, the iterative subspace-based method is not stable, since it uses the original data, which “hold” the imputation only from the left.

AR and SSA forecasting

We refer to Section 3.2 for discussion of common and unique features of AR and SSA forecasting.

For the signal model consisting of a trend and seasonality, seasonal ARIMA can be competitive with SSA. There are real-world examples where ARIMA provides better accuracy; and vice versa (see, e.g., De Klerk, 2015a; Hassani, Heravi, & Zghljavsky, 2009). One of the advantages of ARIMA is the ability to automatically select the order of the ARIMA model using information criteria like the Akaike information criterion (AIC) or the Bayesian information criterion (BIC). The number r of signal components can be chosen in SSA also on the basis of the AIC/BIC approach; however, as it was discussed in the paragraph devoted to the signal extraction in Section 3.14, this approach has many limitations. Note that the Seasonal ARIMA model requires to know the period of the periodic time series component, whereas SSA does not. Moreover, observations of a few periods can be sufficient for SSA and definitely is insufficient for seasonal ARIMA. Fragment 3.5.18²³ contains an example of comparison of SSA, seasonal ARIMA and ETS.

3.11 | SSA and signal detection (Monte Carlo SSA)

The problem of signal detection is very important in practice. If noise is strong, it is easy to find spurious signals, since noise (being considering as a stationary process) contains all frequencies. The mean contribution of each frequency is determined by the spectral density of noise. In particular, this implies that if the spectral density is larger for low frequencies, the probability of spurious trends or spurious sine waves with low frequencies increases. This is exactly the case of the so-called red noise (AR(1) with a positive coefficient).

Since we observe one realization of a time series, the contribution of each frequency from the grid $\{k/N, k = 0, \dots, [N/2]\}$ (we consider the periodogram values, which correspond to these frequencies, as their contributions) is random with variance, which does not tend to zero as N tends to infinity. Moreover, it has an exponential distribution, that is, large values are likely.

The question of existence of a signal in noise can be reduced to the construction of a criterion for testing the null hypothesis that the time series is a pure noise; the criterion should be powerful against the alternative that a signal is present. There are a lot of such criteria for different models of noise. Most of them are related to the white noise model.

In the framework of SSA, red noise is the other focus of attention. One of the reasons is that SSA was primarily popular in climatology, where climatic time series are conventionally modeled as red noise. In addition, the properties of red noise suit SSA, since red noise has a monotonic spectral function (see Section 3.8 for a brief discussion).

The method for detection of a signal in red noise was called Monte Carlo SSA (Allen & Robertson, 1996; Allen & Smith, 1996; Garnot, Groth, & Ghil, 2018; Greco et al., 2011; Groth & Ghil, 2015; Jemwa & Aldrich, 2006; Palus & Novotná, 2004), since it uses simulations. It seems that the method's name does not reflect its purpose as a method for hypothesis testing; however, that is the name by which this approach is known.

The approach used in Monte Carlo SSA is straightforward. First, a characteristic of data that reflects the difference between the null and alternative hypotheses should be chosen; then surrogate data are simulated according to the null-hypothesis to construct the distribution of the chosen characteristic. Finally, one checks if the chosen characteristic of the real-world data, which can be called test statistic, lies outside $(1 - \gamma)/2$ -and $(1 + \gamma)/2$ -quantiles of the constructed distribution. If this is true, the null hypothesis is rejected at the significance level $1 - \gamma$. In the case of Monte Carlo SSA, this characteristic is the squared norm of the projection of the trajectory matrix on a chosen vector which refers to a given frequency. The relation with SSA is in the choice of the projection vector as one of the eigenvectors of the trajectory matrix; then the test statistic is equal to the corresponding eigenvalue.

Certainly, there are additional problems, which should be solved applying the described scheme. For example, the parameters of the AR(1) process satisfying the null-hypothesis are unknown and should be estimated. For signal detection, it is not enough to choose only one characteristic (one projection vector corresponding to one frequency if we speak about Monte Carlo SSA), since a probable signal contain frequencies unknown in advance. Thus, the problem of multiple testing arises. We refer to (Golyandina, 2019) for the description of a more strict statistical approach to constructing the Monte Carlo SSA test, where the problem of multiple testing is solved and an approach for controlling the type I error and estimating the test power is suggested.

3.12 | SSA and outliers

The problem of robustness to outliers is essential for any method. Let us consider how this problem can be solved in SSA. Recall that for signal extraction, Basic SSA can be expressed through two projections: $\tilde{S} = T^{-1}\Pi_{\mathcal{H}}\Pi_r TX$ (Section 2.5.2). In Basic SSA, both projections are performed in the Frobenius norm (which can be called the norm in L_2). The squared Frobenius norm is equal to the sum of squared matrix/vector entries.

There are two modifications of SSA, in which the projections are performed in another norm.

Weighted projections

The first approach is to use a weighted norm with different weights of the time series points, where the points which are suspected to be outliers have smaller weights. In this approach, the projections are performed in the weighted Frobenius norm. The weights are chosen by an iterative procedure like that used in the LOWESS nonparametric smoothing (Cleveland, 1979) or in the iteratively reweighted least-squares method (IRLS) (Holland & Welsch, 1977), where the weights are chosen depending upon the residual values in a specific way.

In this approach, the algorithm of SSA with weights should be implemented. SSA with special weights, where the ordinary SVD is changed to the oblique SSA, can be implemented with approximately the same computational cost as Basic SSA (Zvonarev & Golyandina, 2017). However, arbitrary weights of different time series points are not the case. The SVD with arbitrary weights has no closed-form solution and therefore needs an iterative numerical solution. Thus, the algorithm of SSA with weighted projections, which helps to remove outliers, contains two loops: the inner loop (to calculate the weighted SVD) and the outer loop (to recalculate weights basing on the residuals); therefore, the algorithm is very time-consuming. This approach is described in Trickett, Burroughs, and Milton (2012) and Chen and Sacchi (2014), where the authors consider the weighted projection to the set of low-rank matrices; however, it seems they consider the unweighted projection to the set of Hankel matrices (compare with Zvonarev & Golyandina, 2017, Prop.2).

L_1 projections

The second approach is also frequently used in approximation problems. To improve the robustness, the projections are constructed in the L_1 -norm. The idea to use the L_1 -norm matrix approximation instead of the L_2 -norm one (i.e., instead of the ordinary SVD if we talk about SSA) is very popular in data analysis. Again, L_1 -SVD has no closed-form solution and therefore time-consuming iterative algorithms should be applied. There are many papers devoted to L_1 low-rank approximations. It is

shown in Markopoulos, Karystinos, and Pados (2014) that the optimal solution in the real-valued case has the computational cost of order $O(N^{dk-k+1})$, where d is the rank of the data matrix and k is the number of desirable components. Therefore, sub-optimal solutions are considered to decrease the cost (see, e.g., Kundu, Markopoulos, & Pados, 2014).

The L_1 -projection on the set of Hankel matrices is performed by the change of diagonal averaging to taking medians instead of averages. The algorithm with the use of L_1 -norm is considered in Kalantari, Yarmohammadi, and Hassani (2016). However, the problem of its implementation with a reasonable computational cost is still not solved.

The previous considerations were related to modifications, which would be robust to outliers. Another problem is to detect outliers. A common approach to outlier detection is to use a change-point detection method; then outliers are removed and the data can be analyzed by a nonmodified standard method. Detection of outliers can be performed by subspace-based methods with the help of SSA, see (Golyandina et al., 2001, chapter 3.6.1), or by standard statistical methods in the SVD step (which is similar to PCA; see, e.g., De Klerk, 2015b).

3.13 | SSA and a priori/a posteriori information

Let us consider what information about the time series can help to modify the SSA algorithm for more accurate estimates or to analyze the algorithm results. Note that the general rule is valid: if the used a priori information is wrong, the modified algorithm can yield totally wrong results.

The most frequently used a priori assumption is the stationarity of the time series; then Toeplitz SSA is used (Section 2.2.6). In Fragment 2.2.2²⁴ the comparison of Toeplitz SSA with Basic SSA in dependence on the exponential rate is demonstrated (the rate equal to 0 corresponds to the stationarity).

Another possible a-priori assumption is that the trend is polynomial. Especially for the linear trend, it is theoretically proved and empirically confirmed that SSA with projection can considerably improve the trend extraction, see Section 3.5 for details.

The second approach can be called posterior (Holmström & Launonen, 2013; Launonen & Holmström, 2017) or bootstrap. On the basis of the bootstrap approach, bootstrap confidence intervals can be constructed for any characteristic, which is estimated by SSA; for example, for the signal itself or for the signal's parameters. The bootstrap approach includes estimation of the signal and noise parameters based on the SSA decomposition. Then simulation of a sample consisting of the estimated signal plus simulated noise allows one to construct confidence and prediction intervals. Note that the same approach is used in Monte Carlo SSA (which is actually bootstrap SSA) for testing hypotheses, see Section 3.11, and in the SSA-forecasting for constructing the bootstrap confidence intervals (Golyandina et al., 2018, section 3.2.1.5).

The posterior approach used in Holmström and Launonen (2013) and Launonen and Holmström (2017) for the detection of trend/periodic components tests the stability of the decomposition components to distinguish between the signal and noise.

3.14 | SSA: Automatic identification and batch processing

Let us describe approaches to the automatic identification of ETs in SSA for their grouping and then extracting the trend and periodic time series components.

Trend identification

The methods proposed for the automatic identification of the trend components are quite natural, since the trend can be described in a nonparametric way as a low-frequency component of the time series.

In Vautard et al. (1992), different methods for the detection of trend components in the SVD of the trajectory matrix were suggested; in particular, the number of zeros or the Kendall's tau correlation coefficient were considered for the detection of trend eigenvectors in the grouping step of the SSA algorithm. The number of zeros shows (in an indirect way) if a component is low-frequency. The Kendall's tau correlation reflects if a component is increasing or decreasing.

In Alexandrov (2009) and Golyandina and Zhitljavskiy (2013, section 2.4.5.2), low-frequency components are extracted in a direct way by analysis of component's periodograms. More precisely, eigenvectors or factor vectors (or elementary reconstructed time series) taken from the SSA decomposition are considered. Then a frequency range $[0, \omega_0]$ and a threshold are chosen. If the contribution of frequencies from the given frequency range is larger than the threshold, the component is referred to as the trend one. This simple algorithm works very well if the trend components are separated from the residual. If the trend does not have a complex form, the trend is usually well separated, see Fragment 2.8.9.²⁵ A slightly different approach is described in Watson (2016).

Periodicity identification

The approach based on component's periodograms can be extended to detecting harmonics (sine waves). A specific feature of the harmonics extraction is that a sine wave produces two components in the SSA decomposition for any frequency in $(0, 0.5)$ and only one component for the frequency 0.5. The algorithm of the recognition of paired sine-wave components based on the component's periodograms was suggested in Vautard et al. (1992) and studied for the application to exponentially-modulated harmonics in Alexandrov and Golyandina (2005).

Whereas the trend is as a rule well separated from the residual, pairs of components produced by different harmonics can mix if the harmonic's amplitudes are close. Two SSA modifications, iterative oblique SSA and SSA with derivatives (see Section 2.10.2) can be applied for improving the separability before the use of identification algorithms.

Grouping

Above, we described different approaches to the trend identification and the identification of periodic components. A common approach to the automatic grouping is to apply a clustering algorithm to the matrix of weighted correlations between the elementary reconstructed time series. If the time series components are well separated, this approach works well, see Fragment 2.7.1.²⁶ However, this way of grouping fails if the groups are poorly separated.

Use of automatic identification

Automatic identification and batch processing have their own parameters, which can be chosen according to the assumed structure of the time series components of interest. Therefore, these techniques work in the case of analysis of a collection of similar time series. Generally, to choose parameters of the identification procedure, a preliminary analysis of several time series should be performed in an interactive manner.

Note that a method of automatic identification, which calculates some characteristics of the decomposition components and then compares them with a threshold, can provide a helpful guess for the interactive grouping based on the values of the considered characteristics.

Signal identification

A completely different way of identification of the signal components is based on the approach that uses a parametric model. If the signal is assumed to be of finite rank r and is dominated, that is, the r leading components correspond to the signal in the SSA decomposition, then the model (which is determined by the signal rank) can be chosen by information criteria like AIC or BIC (BIC is recommended). The information criteria need finding the MLE of the model parameters, whereas, as a rule, an LS estimate of the signal is constructed within low-rank approximation methods (Section 3.4). The WLS estimate with appropriate weights, which coincides with the MLE if the residuals are Gaussian, is considered in Zvonarev and Golyandina (2018), where a fast algorithm for the WLS estimation is proposed; in particular, the difference with the approach from Usevich and Markovsky (2014) is discussed.

Consider the simple case of white Gaussian noise, where the ordinary LS estimate and the MLE are the same. Denote $\tilde{S}(d)$ the LS estimate of the signal of length N assuming the parametric model of time series of rank d . The number of parameters is $2d$. Define $RSS(d) = \|\tilde{S}(d) - S\|^2$. Then, by definition,

$$AIC(d) = N \ln(RSS(d)/N) + 4d, \quad BIC(d) = N \ln(RSS(d)/N) + 2d \ln N.$$

The values of AIC/BIC can be used for the choice of rank r in a conventional way: the estimate of r is the minimum point of the considered information criterion.

Information criteria "as is" can be used in the SLRA statement of the problem, see Section 3.4, if the signal is of finite rank and we are able to construct its MLE. In practice, signals are only approximated by time series of finite rank. Even if the signal is of finite rank, the signal estimate given by SSA is generally not of finite rank and is not the LS estimate. Thus, the use of information criteria for the rank estimation is questionable in their application to real-world problems.

The other general approach applied to choosing the rank r automatically is the cross-validation, which is briefly discussed in Section 3.10. This approach is time-consuming and can be applied to only long time series; moreover, the aim of this approach is to find r for better prediction/gap filling, not for signal rank estimation. However, this technique is appropriate under much weaker assumptions about the signal and noise; therefore, it is applicable in practice. The R code for the choice of the signal rank r by cross-validation can be found in Fragments 3.5.13–3.5.15.²⁷

3.15 | SSA and machine learning

As we have mentioned, SSA can be called PCA for time series. Therefore, the use of SSA in many cases is similar to the use of PCA (SVD) for multivariate data. In Golyandina et al. (2018, section 1.7.3) one can find a brief review of papers, where SSA together with some other methods (SVM, SVR, and NN among others) are used in machine learning.

4 | IMPLEMENTATION OF SSA

4.1 | Software and fast implementation

At the moment, there are a lot of different implementations of SSA. Let us enumerate several of them:

1. the general-purpose interactive “Caterpillar”-SSA software (<http://gistatgroup.com>, Windows);
2. the interactive software oriented mainly on climatic applications, SSA-MTM Toolkit for spectral analysis and its commercial extension kSpectra Toolkit (<http://www.atmos.ucla.edu/tcd/ssa>, Unix, Mac);
3. the commercial statistical software SAS, which includes SSA to its econometric extension SAS/ETS;
4. the R-package **Rssa**, a cross-platform implementation of a lot of SSA-related methods (<http://cran.r-project.org/web/packages/Rssa>).

Fast effective algorithms are implemented in the **Rssa** package, where the computational cost (in flops) is dropped from $O(N^3)$ down to $O(kN\log N + k^2N)$ and the memory consumption is reduced from $O(N^2)$ to $O(N)$; here N is the time series length, k is the number of calculated ETs, the window length L is considered proportional to N . Briefly, the approach is based on the Lanczos algorithm and on computing the vector multiplication through fast Fourier transform applied for the calculation of convolutions (Golyandina et al., 2015; Korobeynikov, 2010).

Note that the **Rssa**-implementation of the SSA decomposition does not take into consideration if the data were updated (if new data were appended to the time series). Therefore, the application of SSA to the updated data doubles the computational cost. There are different approaches to updating the SVD. However, it is still an unanswered question whether an algorithm for updating SSA can be faster than the current implementation of SSA.

Let us finally remark that reasoning about the computational cost of SSA-related methods, which are reported in the literature, can be irrelevant; e.g., the sources can use only the information known on the publication date. Let us give some examples. In many papers, SSA is considered as a very time-consuming method because of the used SVD expansion; in particular, in some past papers, the method ESPRIT is called time-consuming (with computational cost $O(N^3)$ as the time series length tends to infinity). However, the method implementation in **Rssa** is much faster.

Another example is the SSA vector forecasting. In Golyandina et al. (2001), this method is called very time-consuming in comparison with the SSA recurrent forecasting. In **Rssa**, the implementation of the SSA vector forecasting (Golyandina et al., 2015) is even faster than that of the recurrent one. On the other hand, this does not mean that the recurrent forecasting cannot be done faster in next implementations.

One more example is related to mathematical issues. In SSA and especially in MSSA, the understanding that for the original trajectory matrix and for the transposed one the SVD decompositions are in fact the same can help to considerably decrease the computational cost by choosing the case, which is less time-consuming for the used numerical algorithm.

4.2 | Example of calculations in **Rssa**

Let us demonstrate how fast are the computations in **Rssa**. For the time series length $N = 1,000,000$ and the window length $L = 500,000$, the reconstruction of a sine wave signal based on two leading components is executed in a few seconds:

```
> library("Rssa")
> N <- 1000000
> signal <- sin((1:N)*2*pi/10)
> ts <- signal + 10*rnorm(1:N)
> system.time(s <- ssa(ts, L = N/2, svd.method = "auto", neig = 2))
      user     system    elapsed
      1.19      0.16     1.34
```

```
> system.time(rec <- reconstruct(s, groups = list(sig = 1:2)))
      user      system     elapsed
      0.55       0.13      0.67
> max(abs(signal - rec$sig))
[1] 0.0515102
```

5 | CONCLUSION

As the readers can see, even a brief description of SSA-related themes composes a very large paper. It is therefore difficult to complete this review paper with a concise conclusion. Summing up, we want to express the hope that the paper can help researchers from various scientific fields to gain new insights and successfully apply SSA to their studies together with other standard methods.

ACKNOWLEDGMENTS

I am grateful to my co-authors of the papers and monographs devoted to SSA for the joint work and fruitful discussions, which gave me the possibility to look at SSA from different points of view. My special thanks to Vladimir Nekrutkin for the provided review of the SSA literature of the past years.

CONFLICT OF INTEREST

The author has declared no conflicts of interest for this article.

ORCID

Nina Golyandina  <https://orcid.org/0000-0003-1400-8209>

ENDNOTES

¹ <https://ssa-with-r-book.github.io/01-chapter2-part1.html#fragment-241-noisy-sum-of-three-sinusoids-iterative-o-ssa>

² <https://ssa-with-r-book.github.io/01-chapter2-part1.html#fragment-251-separation-of-two-sine-waves-with-equal-amplitudes>

³ <https://ssa-with-r-book.github.io/01-chapter2-part1.html#fragments-211-australian-wines-input-and-212-fort-reconstruction>

⁴ <https://ssa-with-r-book.github.io/02-chapter2-part2.html#fragment-281-tree-rings-frequency-decomposition>

⁵ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-359-fort-estimation-of-parameters-by-basic-ssa>

⁶ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-3511-fort-estimation-of-parametric-real-valued-form>

⁷ <https://ssa-with-r-book.github.io/04-chapter4.html>

⁸ <https://ssa-with-r-book.github.io/05-chapter5.html#fragment-531-mars-parameter-estimation-with-2d-esprit>

⁹ <https://ssa-with-r-book.github.io/05-chapter5.html#fragment-532-mars-parameter-estimation-with-shaped-2d-esprit>

¹⁰ <https://ssa-with-r-book.github.io/05-chapter5.html#fragment-542-brecon-beacons-decomposition>

¹¹ <https://ssa-with-r-book.github.io/01-chapter2-part1.html#fragment-262-incomplete-decomposition-for-a-series-with-a-gap>

¹² <https://ssa-with-r-book.github.io/05-chapter5.html#fragments-521-auxiliary-plot-of-2d-image-and-522-mars-mask-specification-and-decomposition>

¹³ <https://ssa-with-r-book.github.io/05-chapter5.html#fragments-546-kruppel-analysis-of-data-given-on-a-cylinder>

¹⁴ <https://ssa-with-r-book.github.io/02-chapter2-part2.html#fragment-287-hotel-ssa-with-projection-linear-trend-detection>

¹⁵ <https://ssa-with-r-book.github.io/02-chapter2-part2.html#fragment-286-us-unemployment-improvement-by-derivssa>

¹⁶ <https://ssa-with-r-book.github.io/05-chapter5.html#fragments-524-mars-identification-and-525-mars-improvement-by-derivssa>

¹⁷ <https://ssa-with-r-book.github.io/02-chapter2-part2.html#fragment-282-fort-basic-ssa-and-iterative-o-ssa-trends>

¹⁸ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-3512-sunspots-subspace-tracking>

¹⁹ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-358-fort-cadzow-iterations>

²⁰ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-321-forecasting-of-co2>

- ²¹ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-331-subspace-based-gap-filling>
- ²² <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-332-iterative-gap-filling-one-gap>
- ²³ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-3518-sweetwhite-comparison-of-ssa-arima-and-ets>
- ²⁴ <https://ssa-with-r-book.github.io/01-chapter2-part1.html#fragment-222-simulation-comparison-of-toeplitz-and-basic-ssa>
- ²⁵ <https://ssa-with-r-book.github.io/02-chapter2-part2.html#fragment-289-paynsa-automatically-identified-trend>
- ²⁶ <https://ssa-with-r-book.github.io/01-chapter2-part1.html#fragment-271-white-dwarf-auto-grouping-by-clustering>
- ²⁷ <https://ssa-with-r-book.github.io/03-chapter3.html#fragment-3513-functions-for-the-search-of-optimal-parameters>

ORCID

Nina Golyandina  <https://orcid.org/0000-0003-1400-8209>

RELATED WIRES ARTICLES

[Principal component analysis](#)

REFERENCES

- Alexandrov, T. (2009). A method of trend extraction using singular spectrum analysis. *RevStat*, 7(1), 1–22.
- Alexandrov, T., & Golyandina, N. (2005). Automatic extraction and forecast of time series cyclic components within the framework of SSA. In *Proceedings of the 5th St. Petersburg Workshop on Simulation* (pp. 45–50). St. Petersburg: St. Petersburg State University.
- Allen, M. R., & Robertson, A. W. (1996). Distinguishing modulated oscillations from coloured noise in multivariate datasets. *Climate Dynamics*, 12 (11), 775–784.
- Allen, M. R., & Smith, L. A. (1996). Monte Carlo SSA: Detecting irregular oscillations in the presence of colored noise. *Journal of Climate*, 9(12), 3373–3404.
- Badeau, R., David, B., & Richard, G. (2008). Cramér–Rao bounds for multiple poles and coefficients of quasi-polynomials in colored noise. *IEEE Transactions on Signal Processing*, 56(8–1), 3458–3467. <https://doi.org/10.1109/TSP.2008.921719>
- Badeau, R., Richard, G., & David, B. (2003). Adaptive ESPRIT algorithm based on the PAST subspace tracker. In *In 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003 (ICASSP '03)* (Vol. 6, p. VI-229). New York, NY: IEEE. <https://doi.org/10.1109/ICASSP.2003.1201660>
- Badeau, R., Richard, G., & David, B. (2008). Performance of ESPRIT for estimating mixtures of complex exponentials modulated by polynomials. *IEEE Transactions on Signal Process.*, 56(2), 492–504.
- Barkhuijsen, H., de Beer, R., & van Ormondt, D. (1987). Improved algorithm for noniterative time-domain model fitting to exponentially damped magnetic resonance signals. *Journal of Magnetic Resonance*, 73, 553–557.
- Basilevsky, A., & Hum, D. P. J. (1979). Karhunen–Loéve analysis of historical time series with an application to plantation births in Jamaica. *Journal of the American Statistical Association*, 74, 284–290.
- Beckers, J., & Rixen, M. (2003). EOF calculations and data filling from incomplete oceanographic data sets. *Journal of Atmospheric and Oceanic Technology*, 20, 1839–1856.
- Belonin, M. D., Golubeva, V. A., & Skublov, G. T. (1982). *Faktornij analiz v geologii* [Factor analysis in geology]. Moscow: Nedra Retrieved from <http://www.geokniga.org/books/4531> (in Russian)
- Belonin, M. D., Tatarinov, I. V., Kalinin, O. M., Shimanskij, V. K., Beskrovnaya, O. V., & Pohitonova, T. E. (1971). In A. Margolin (Ed.), *Faktornij analiz v neftyanoj geologii: Obzor* [Factor analysis in petroleum geology: Review]. Moskow: VIEMS (in Russian).
- Bozzo, E., Carniel, R., & Fasino, D. (2010). Relationship between singular spectrum analysis and Fourier analysis: Theory and application to the monitoring of volcanic activity. *Computers & Mathematics with Applications*, 60(3), 812–820.
- Broomhead, D., & King, G. (1986a). Extracting qualitative dynamics from experimental data. *Physica D*, 20, 217–236.
- Broomhead, D., & King, G. (1986b). On the qualitative analysis of experimental dynamical systems. In S. Sarkar (Ed.), *Nonlinear phenomena and chaos* (pp. 113–144). Bristol: Adam Hilger.
- Buchstaber, V. (1994). Time series analysis and grassmannians. In S. Gindikin (Ed.), *Applied problems of radon transform* (Vol. 162, pp. 1–17). Providence, RI: AMS.
- Cadzow, J. A. (1988). Signal enhancement: a composite property mapping algorithm. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(1), 49–62.
- Chen, K., & Sacchi, M. D. (2014). Robust reduced-rank filtering for erratic seismic noise attenuationrobust reduced-rank filtering. *Geophysics*, 80 (1), V1–V11. <https://doi.org/10.1190/geo2014-0116.1>
- Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368), 829–836. <https://doi.org/10.1080/01621459.1979.10481038>

- Colebrook, J. M. (1978). Continuous plankton records—Zooplankton and environment, northeast Atlantic and North Sea, 1948–1975. *Oceanologica Acta*, 1, 9–23.
- Danilov, D., & Zhigljavsky, A. (Eds.). (1997). *Glavnye komponenty vremennyyh ryadov: metod ‘Gusenica’ [principal components of time series: the ‘Caterpillar’ method]*. St. Petersburg: St. Petersburg University Press Retrieved from <http://www.gistatgroup.com/gus/book1/index.html> (in Russian)
- De Klerk, J. (2015a). A comparison of singular spectrum analysis forecasting methods to forecast South African tourism arrivals data. *Studies in Economics and Econometrics*, 39(2), 21–40.
- De Klerk, J. (2015b). Time series outlier detection using the trajectory matrix in singular spectrum analysis with outlier maps and ROBPCA. *South African Statistical Journal*, 49(1), 61–76.
- de Prony, G. (1795). Essai expérimental et analytique sur les lois de la dilatabilité des fluides élastiques et sur celles de la force expansive de la vapeur de l'eau et la vapeur de l'alcool à différentes températures. *Journal de l'École polytechnique*, 1(2), 24–76.
- Eckart, C., & Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3), 211–218. <https://doi.org/10.1007/BF02288367>
- Efimov, V., & Galaktionov, Y. (1983). About possibility of forecasting of cyclic variability for mammal abundance. *Journal of General Biology*, 44 (3), 343–352 (in Russian).
- Elsner, J. B., & Tsonis, A. A. (1996). *Singular spectrum analysis: A new tool in time series analysis*. New York, NY: Plenum.
- Fraedrich, K. (1986). Estimating dimensions of weather and climate attractors. *Journal of the Atmospheric Sciences*, 43, 419–432.
- Gantmacher, F. R. (1959). *The theory of matrices* (Vol. 68). New York, NY: Chelsea Publishing Company.
- Garnot, V. S. F., Groth, A., & Ghil, M. (2018). Coupled climate-economic modes in the Sahel's interannual variability. *Ecological Economics*, 153, 111–123. <https://doi.org/10.1016/j.ecolecon.2018.07.006>
- Ghil, M., & Vautard, R. (1991). Interdecadal oscillations and the warming trend in global temperature time series. *Nature*, 350, 324–327.
- Golyandina, N. (2010). On the choice of parameters in singular spectrum analysis and related subspace-based methods. *Statistics and Its Interface*, 3 (3), 259–279.
- Golyandina, N. (2019). *Statistical approach to detection of signals by Monte Carlo singular spectrum analysis: Multiple testing*. arXiv:1903.01485. Retrieved from <https://arxiv.org/abs/1903.01485>
- Golyandina, N., Korobeynikov, A., Shlemov, A., & Usevich, K. (2015). Multivariate and 2D extensions of singular spectrum analysis with the Rssa package. *Journal of Statistical Software*, 67(2), 1–78.
- Golyandina, N., Korobeynikov, A., & Zhigljavsky, A. (2018). *Singular spectrum analysis with R*. Berlin, Heidelberg: Springer-Verlag.
- Golyandina, N., & Lomtev, M. (2016). Improvement of separability of time series in singular spectrum analysis using the method of independent component analysis. *Vestnik St. Petersburg University. Mathematics*, 49(1), 9–17.
- Golyandina, N., Nekrutkin, V., & Zhigljavsky, A. (2001). *Analysis of time series structure: SSA and related techniques*. New York, NY: Chapman & Hall/CRC Press.
- Golyandina, N., & Osipov, E. (2007). The “Caterpillar”-SSA method for analysis of time series with missing values. *Journal of Statistical Planning and Inference*, 137(8), 2642–2653.
- Golyandina, N., & Shlemov, A. (2015). Variations of singular spectrum analysis for separability improvement: Non-orthogonal decompositions of time series. *Statistics and Its Interface*, 8(3), 277–294.
- Golyandina, N., & Shlemov, A. (2017). Semi-nonparametric singular spectrum analysis with projection. *Statistics and Its Interface*, 10(1), 47–57.
- Golyandina, N., & Usevich, K. (2010). 2D-extension of singular spectrum analysis: algorithm and elements of theory. In V. Olshevsky & E. Tyrtyshnikov (Eds.), *Matrix methods: Theory, algorithms and applications* (pp. 449–473). Singapore: World Scientific Publishing.
- Golyandina, N., & Zhigljavsky, A. (2013). *Singular spectrum analysis for time series*. Berlin: Springer.
- Greco, G., Rosa, R., Beskin, G., Karpov, S., Romano, L., Guarneri, A., ... Bedogni, R. (2011). Evidence of deterministic components in the apparent randomness of GRBs: clues of a chaotic dynamic. *Scientific Reports*, 1, 91.
- Grenander, U., & Szegő, G. (1984). *Toeplitz forms and their applications*. New York, NY: Chelsea.
- Groth, A., & Ghil, M. (2011, Sep). Multivariate singular spectrum analysis and the road to phase synchronization. *Physical Review E*, 84, 036206. <https://doi.org/10.1103/PhysRevE.84.036206>
- Groth, A., & Ghil, M. (2015). Monte Carlo singular spectrum analysis (SSA) revisited: Detecting oscillator clusters in multivariate datasets. *Journal of Climate*, 28(19), 7873–7893. <https://doi.org/10.1175/JCLI-D-15-0100.1>
- Hall, M. J. (1998). *Combinatorial theory*. New York, NY: Wiley.
- Hansen, P. C., & Jensen, S. H. (1998, June). FIR filter representations of reduced-rank noise reduction. *IEEE Transactions on Signal Processing*, 46, 1737–1741.
- Harris, T., & Yan, H. (2010). Filtering and frequency interpretations of singular spectrum analysis. *Physica D*, 239, 1958–1967.
- Hassani, H., Heravi, S., & Zhigljavsky, A. (2009). Forecasting European industrial production with singular spectrum analysis. *International Journal of Forecasting*, 25(1), 103–118.
- Hassani, H., & Thomakos, D. (2010). A review on singular spectrum analysis for economic and financial time series. *Statistics and Its Interface*, 3 (3), 377–397.
- Hassani, H., Xu, Z., & Zhigljavsky, A. (2011). Singular spectrum analysis based on the perturbation theory. *Nonlinear Analysis: Real World Applications*, 12(5), 2752–2766. <https://doi.org/10.1016/j.nonrwa.2011.03.020>
- Holland, P. W., & Welsch, R. E. (1977). Robust regression using iteratively reweighted least-squares. *Communications in Statistics—Theory and Methods*, 6(9), 813–827. <https://doi.org/10.1080/03610927708827533>

- Holmström, L., & Launonen, I. (2013, October). Posterior singular spectrum analysis. *Statistical Analysis and Data Mining*, 6(5), 387–402. <https://doi.org/10.1002/sam.11195>
- Hua, Y. (1992). Estimating two-dimensional frequencies by matrix enhancement and matrix pencil. *IEEE Transactions on Signal Processing*, 40(9), 2267–2280.
- Huang, N. E., & Wu, Z. (2008). A review on Hilbert–Huang transform: Method and its applications to geophysical studies. *Reviews of Geophysics*, 46, RG2006. <https://doi.org/10.1029/2007RG000228>
- Ivanova, E., & Nekrutkin, V. (2019). Two asymptotic approaches for the exponential signal and harmonic noise in singular spectrum analysis. *Statistics and Its Interface*, 12(1), 49–59.
- Jemwa, G. T., & Aldrich, C. (2006). Classification of process dynamics with Monte Carlo singular spectrum analysis. *Computers & Chemical Engineering*, 30(5), 816–831. <https://doi.org/10.1016/j.compchemeng.2005.12.005>
- Kalantari, M., Yarmohammadi, M., & Hassani, H. (2016). Singular spectrum analysis based on L1-norm. *Fluctuation and Noise Letters*, 15(01), 1650009.
- Kato, M., Yen-Wei Chen, & Gang Xu. (2006). Articulated hand tracking by PCA-ICA approach. In *7th International Conference on Automatic Face and Gesture Recognition (FGFR06)* (pp. 329–334). doi: <https://doi.org/10.1109/FGR.2006.21>
- Keppenne, C., & Ghil, M. (1992). Adaptive filtering and prediction of the southern oscillation index. *Journal of Geophysical Research: Atmospheres*, 97(D18), 20449–20454.
- Keppenne, C., & Lall, U. (1996). Complex singular spectrum analysis and multivariate adaptive regression splines applied to forecasting the southern oscillation. *Experimental Long-Lead Forecast Bulletin*. <https://www.cpc.ncep.noaa.gov/products/predictions/experimental/bulletin/Mar96/article13.html>
- Khan, M. A. R., & Poskitt, D. S. (2013). A note on window length selection in singular spectrum analysis. *Australian & New Zealand Journal of Statistics*, 55(2), 87–108. <https://doi.org/10.1111/anzs.12027>
- Kondrashov, D., & Ghil, M. (2006). Spatio-temporal filling of missing points in geophysical data sets. *Nonlinear Processes in Geophysics*, 13(2), 151–159.
- Korobeynikov, A. (2010). Computation- and space-efficient implementation of SSA. *Statistics and Its Interface*, 3(3), 357–368.
- Korobeynikov, A., Shlemov, A., Usevich, K., & Golyandina, N. (2017). *Rssa: A collection of methods for singular spectrum analysis. R package version 1.0* [Computer software manual]. Retrieved from <http://CRAN.R-project.org/package=Rssa>
- Kouchaki, S., Sanei, S., Arbon, E. L., & Dijk, D. (2015, Jan). Tensor based singular spectrum analysis for automatic scoring of sleep EEG. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(1), 1–9. <https://doi.org/10.1109/TNSRE.2014.2329557>
- Kumaresan, R., & Tufts, D. (1982). Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 30(6), 833–840.
- Kundu, S., Markopoulos, P. P., & Pados, D. A. (2014). Fast computation of the L1-principal component of real-valued data. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 8028–8032). New York, NY: IEEE. <https://doi.org/10.1109/ICASSP.2014.6855164>
- Launonen, I., & Holmström, L. (2017). Multivariate posterior singular spectrum analysis. *Statistical Methods & Applications*, 26(3), 361–382. <https://doi.org/10.1007/s10260-016-0371-x>
- Leles, M. C. R., Sansao, J. P. H., Mozelli, L. A., & Guimaraes, H. N. (2018). A new algorithm in singular spectrum analysis framework: The Overlap-SSA (ov-SSA). *SoftwareX*, 8, 26–32. <https://doi.org/10.1016/j.softx.2017.11.001>
- Markopoulos, P. P., Karystinos, G. N., & Pados, D. A. (2014, Oct). Optimal algorithms for L_1 -subspace signal processing. *IEEE Transactions on Signal Processing*, 62(19), 5046–5058. <https://doi.org/10.1109/TSP.2014.2338077>
- Markovsky, I. (2019). *Low rank approximation: Algorithms, implementation, applications (communications and control engineering)* (2nd ed.). Berlin: Springer.
- Mees, A. I., Rapp, P. E., & Jennings, L. S. (1987, Jul). Singular-value decomposition and embedding dimension. *Physical Review A*, 36, 340–346. <https://doi.org/10.1103/PhysRevA.36.340>
- Moskvina, V., & Zhigljavsky, A. (2003). An algorithm based on singular spectrum analysis for change-point detection. *Communications in Statistics: Simulation and Computation*, 32(2), 319–352.
- Nekrutkin, V. (2010). Perturbation expansions of signal subspaces for long signals. *Statistics and Its Interface*, 3, 297–319.
- Nekrutkin, V., & Vasilinetc, I. (2017). Asymptotic extraction of common signal subspaces from perturbed signals. *Statistics and Its Interface*, 10(1), 27–32.
- Oropeza, V. (2010). *The singular spectrum analysis method and its application to seismic data denoising and reconstruction* (master thesis). University of Alberta.
- Palus, M., & Novotná, D. (2004, December). Enhanced Monte Carlo singular system analysis and detection of period 7.8 years oscillatory modes in the monthly NAO index and temperature records. *Nonlinear Processes in Geophysics*, 11(5/6), 721–729.
- Papy, J. M., De Lathauwer, L., & Van Huffel, S. (2005). Exponential data fitting using multilinear algebra: the single-channel and multi-channel case. *Numerical Linear Algebra with Applications*, 12(8), 809–826.
- Pietilä, A., El-Segagier, M., Vigário, R., & Pesonen, E. (2006). Blind source separation of cardiac murmurs from heart recordings. In J. Rosca, D. Erdogmus, J. C. Príncipe, & S. Haykin (Eds.), *Independent component analysis and blind signal separation* (Vol. 3889, pp. 470–477). Berlin/Heidelberg: Springer.
- Plaut, G., & Vautard, R. (1994). Spells of low-frequency oscillations and weather regimes in the northern hemisphere. *Journal of the Atmospheric Sciences*, 51, 210–236.

- Roy, R., & Kailath, T. (1989). ESPRIT: estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37, 984–995.
- Sahnoun, S., Usevich, K., & Comon, P. (2017). Multidimensional ESPRIT for damped and undamped signals: Algorithm, computations, and perturbation analysis. *IEEE Transactions on Signal Processing*, 65(22), 5897–5910.
- Saniei, S., & Hassani, H. (2015). *Singular spectrum analysis of biomedical signals*. New York, NY: CRC Press.
- Santamaría, I., Pantaleón, C., & Ibanez, J. (2000). A comparative study of high-accuracy frequency estimation methods. *Mechanical Systems and Signal Processing*, 14(5), 819–834. <https://doi.org/10.1006/mssp.2000.1321>
- Sauer, Y., Yorke, J., & Casdagli, M. (1991). Embedology. *Journal of Statistical Physics*, 65, 579–616.
- Shlemov, A., & Golyandina, N. (2014). Shaped extensions of singular spectrum analysis. In *21st International Symposium on Mathematical Theory of Networks and Systems, July 7–11, 2014. Groningen, The Netherlands* (pp. 1813–1820).
- Sidiropoulos, N. D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E. E., & Faloutsos, C. (2017, July). Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13), 3551–3582. <https://doi.org/10.1109/TSP.2017.2690524>
- Stoica, P., & Moses, R. L. (2005). *Spectral analysis of signals*. Upper Saddle River, NJ: Pearson/Prentice Hall.
- Stoica, P., & Soderstrom, T. (1991, Aug). Statistical analysis of MUSIC and subspace rotation estimates of sinusoidal frequencies. *IEEE Transactions on Signal Processing*, 39(8), 1836–1847. <https://doi.org/10.1109/78.91154>
- Trickett, S. (2003). F-xy eigenimage noise suppression. *Geophysics*, 68(2), 751–759.
- Trickett, S., Burroughs, L., & Milton, A. (2012). *Robust rank-reduction filtering for erratic noise* (Tech. Rep.). doi: <https://doi.org/10.1190/segam2012-0129.1>
- Tufts, D. W., Kumaresan, R., & Kirssteins, I. (1982, June). Data adaptive signal estimation by singular value decomposition of a data matrix. *Proceedings of the IEEE*, 70(6), 684–685.
- Usevich, K., & Markovsky, I. (2014). Variable projection for affinely structured low-rank approximation in weighted 2-norms. *Journal of Computational and Applied Mathematics*, 272, 430–448.
- Van Huffel, S. (1993). Enhanced resolution based on minimum variance estimation and exponential data modeling. *Signal Processing*, 33(3), 333–355. [https://doi.org/10.1016/0165-1684\(93\)90130-3](https://doi.org/10.1016/0165-1684(93)90130-3)
- Vautard, R., & Ghil, M. (1989). Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series. *Physica D*, 35, 395–424.
- Vautard, R., Yiou, P., & Ghil, M. (1992). Singular-spectrum analysis: A toolkit for short, noisy chaotic signals. *Physica D*, 58, 95–126.
- Vlassieva, E., & Golyandina, N. (2009). First-order SSA-errors for long time series: model examples of simple noisy signals. In *Proceedings of the 6th St. Petersburg Workshop on Simulation, June 28–July 4, 2009, St. Petersburg* (Vol. 1, pp. 314–319). St.-Petersburg: St.-Petersburg State University.
- Watson, P. J. (2016). Identifying the best performing time series analytics for sea level research. In I. Rojas & H. Pomares (Eds.), *Time series analysis and forecasting: Selected contributions from the itise conference* (pp. 261–278). Cham: Springer International Publishing.
- Weare, B. C., & Nasstrom, J. S. (1982). Examples of extended empirical orthogonal function analyses. *Monthly Weather Review*, 110(6), 481–485.
- Yang, D., Yi, C., Xu, Z., Zhang, Y., Ge, M., & Liu, C. (2017). Improved tensor-based singular spectrum analysis based on single channel blind source separation algorithm and its application to fault diagnosis. *Applied Sciences*, 7(4), 1–15.
- Yiou, P., Baert, E., & Loutre, M. (1996). Spectral analysis of climate data. *Surveys in Geophysics*, 17, 619–663.
- Yiou, P., Sornette, D., & Ghil, M. (2000). Data-adaptive wavelets and multi-scale singular-spectrum analysis. *Physica D: Nonlinear Phenomena*, 142(3), 254–290. [https://doi.org/10.1016/S0167-2789\(00\)00045-2](https://doi.org/10.1016/S0167-2789(00)00045-2)
- Zvonarev, N., & Golyandina, N. (2017). Iterative algorithms for weighted and unweighted finite-rank time-series approximations. *Statistics and Its Interface*, 10(1), 5–18.
- Zvonarev, N., & Golyandina, N. (2018). Modified Gauss–Newton method in low-rank signal estimation. arXiv:1803.01419. Retrieved from <https://arxiv.org/abs/1803.01419>

How to cite this article: Golyandina N. Particularities and commonalities of singular spectrum analysis as a method of time series analysis and signal processing. *WIREs Comput Stat*. 2020;12:e1487. <https://doi.org/10.1002/wics.1487>