Chapter 1
Introduction: Overview

The present book expands SSA methodology in many different directions and unifies different approaches and modifications within the SSA framework. This chapter is introductory; it outlines the main principles and ideas of SSA, presents a unified view on SSA, reviews its computer implementation in the form of the Rssa package, and gives references to all data sources used. The chapter contains eight sections serving different objectives.

Section 1.1 describes the generic structure of all methods from the SSA family and introduces the main concepts essential for understanding different versions of SSA and for making application of SSA in practice efficient.

Section 1.2 classifies different versions of SSA. As explained in that section, there are two complementary directions in which versions of SSA can be created: one is related to geometrical features of the object $\mathbb{X}$ and the other one is determined by the choice of the procedure of decomposition of the trajectory matrix into rank-one matrices. These two directions of variations of SSA are not related to each other so that any extension of SSA related to the geometry of $\mathbb{X}$ can be combined with any procedure of decomposition of the trajectory matrix.

Section 1.3 discusses the concept of separability, which is the most theoretically important concept of SSA. Achieving separability (for example, of a signal from noise) is the key task of SSA in most applications. Correct understanding of this concept is therefore imperative for making a particular application of SSA reliable and efficient. We will be returning to separability in many discussions within the book.

Section 1.4 briefly introduces the main underlying model used to apply SSA for common problems such as forecasting, imputation of missing data, and monitoring structural stability of time series. In the one-dimensional case, this model assumes that a part of the series can be described by a linear recurrent relation and in particular, by a sum of damped sinusoids. Estimation of parameters of this model often constitutes the main objective in signal processing.

N. Golyandina et al., *Singular Spectrum Analysis with R*, Use R!,
https://doi.org/10.1007/978-3-662-57380-8_1

In Sect. 1.5, we give information about most known implementations of SSA, describe the general structure of the RSSA package, and discuss efficiency of its implementation.

In Sect. 1.6 we briefly discuss the place of SSA among other methods of time series analysis, signal and image processing and provide a short overview of recent publications where a comparison of SSA with several traditional methods has been made.

In Sect. 1.7 we make a short historical survey of SSA, refer to recent applications of SSA and to papers which discuss combination of SSA with other methods; we also list the main papers, which a significant part of this book is based upon.

In Sect. 1.8 we make comments concerning installation of the RSSA package and describe the real-life data sets (taken from many different sources) which we have used in the book for illustrations. We provide the basic information about these data sets and specify their location. The corresponding references would help the reader to get more information about any of these data sets.

1.1 Generic Scheme of the SSA Family and the Main Concepts

We use SSA as a generic term for a family of methods based on a sequential application of the four steps schematically represented in Fig. 1.1 below and briefly described in the next section.

1.1.1 SSA Methods

We define an SSA method (or simply SSA) as any method performing the four steps depicted in Fig. 1.1 and briefly described below. The input object $\mathbb{X}$ is an ordered collection of N numbers (e.g., a time series or a digital image). We denote the set of such objects by $\mathcal{M}$. Unless stated otherwise, the entries of $\mathbb{X}$ are assumed to be real numbers although a straightforward generalization of the main SSA method to the case of complex numbers is available, see Sect. 4.1.

Input: $\mathbb{X}$, an ordered collection of N numbers.
Output: A decomposition of $\mathbb{X}$ into a sum of identifiable components: $\mathbb{X} = \widetilde{\mathbb{X}}_1 + \ldots + \widetilde{\mathbb{X}}_m$.

Step 1: Embedding The so-called *trajectory matrix* $\mathbf{X} = \mathcal{T}(\mathbb{X})$ is constructed, where $\mathcal{T}$ is a linear map transforming the object $\mathbb{X}$ into an $L \times K$ matrix of certain structure. Let us denote the set of all possible trajectory matrices by $\mathcal{M}^{(\mathrm{H})}_{L,K}$. The letter H is used to stress that these matrices have Hankel-related structure.

As an example, in 1D-SSA (that is, SSA for the analysis of one-dimensional real-valued time series), $\mathbb{X} = (x_1, \ldots, x_N)$ and $\mathcal{T} = \mathcal{T}_{\mathrm{1D\text{-}SSA}}$ maps R^N to the space

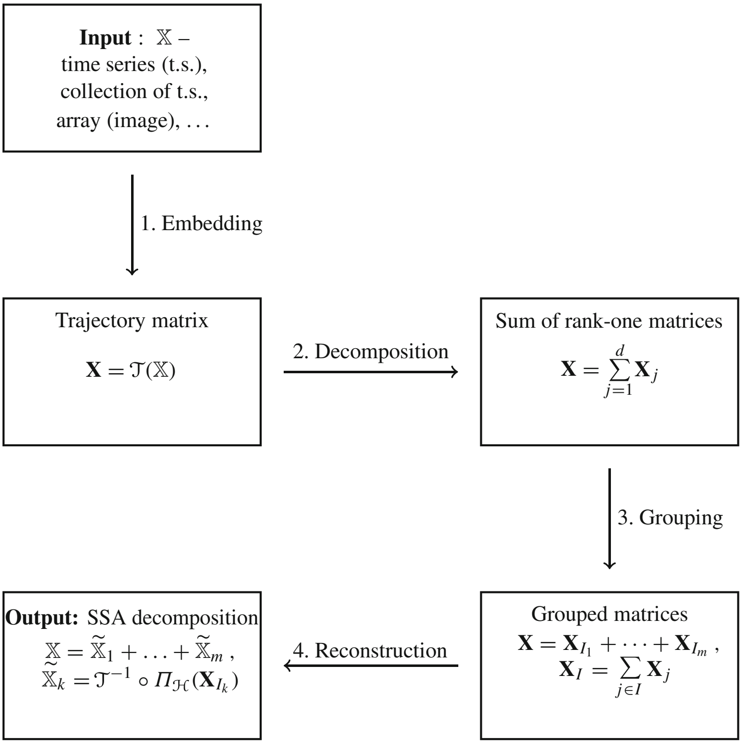
Input : $\mathbb{X}$ –
time series (t.s.),
collection of t.s.,
array (image), ...
1. Embedding
Trajectory matrix
$\mathbf{X} = \mathcal{T}(\mathbb{X})$
2. Decomposition
Sum of rank-one matrices
$\mathbf{X} = \sum_{j=1}^{d} \mathbf{X}_j$
3. Grouping
Grouped matrices
$\mathbf{X} = \mathbf{X}_{I_1} + \cdots + \mathbf{X}_{I_m}$,
$\mathbf{X}_I = \sum_{j \in I} \mathbf{X}_j$
4. Reconstruction
Output: SSA decomposition
$\mathbb{X} = \widetilde{\mathbb{X}}_1 + \ldots + \widetilde{\mathbb{X}}_m$,
$\widetilde{\mathbb{X}}_k = \mathcal{T}^{-1} \circ \Pi_{\mathcal{H}}(\mathbf{X}_{I_k})$

Fig. 1.1 SSA family: Generic scheme

of Hankel matrices $L \times K$ with equal values on the anti-diagonals, where N is the series length, L is the window length, which is a parameter, and $K = N - L + 1$:

$$\mathcal{T}_{\text{1D-SSA}}(\mathbb{X}) = \begin{pmatrix} x_1 & x_2 & x_3 & \ldots & x_K \\ x_2 & x_3 & x_4 & \ldots & x_{K+1} \\ x_3 & x_4 & x_5 & \ldots & x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & x_{L+2} & \ldots & x_N \end{pmatrix}. \tag{1.1}$$

Step 2: Decomposition of X into a Sum of Matrices of Rank 1 The result of this step is the decomposition

$$\mathbf{X} = \sum_i \mathbf{X}_i, \quad \mathbf{X}_i = \sigma_i U_i V_i^{\mathrm{T}}, \tag{1.2}$$

where $U_i \in \mathsf{R}^L$ and $V_i \in \mathsf{R}^K$ are vectors such that $\|U_i\| = 1$ and $\|V_i\| = 1$ for all i and σ_i are non-negative numbers.

The main example of this decomposition is the conventional singular value decomposition (SVD) for real-valued matrices $\mathbf{X}$. If this conventional SVD is used, then we call the corresponding SSA method "Basic SSA" (Golyandina et al. 2001; Chapter 1). Let $\mathbf{S} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$, $\lambda_1 \geq \ldots \geq \lambda_L \geq 0$ be *eigenvalues* of the matrix $\mathbf{S}$, $d = \operatorname{rank} \mathbf{X} = \max\{j : \lambda_j > 0\}$, $U_1, \ldots, U_d$ be the corresponding *eigenvectors*, and $V_j = \mathbf{X}^{\mathrm{T}} U_j / \sqrt{\lambda_j}$, $j = 1, \ldots, d$, be *factor vectors*. Denote $\mathbf{X}_j = \sqrt{\lambda_j} U_j V_j^{\mathrm{T}}$. Then the SVD of the trajectory matrix $\mathbf{X}$ can be written as

$$\mathbf{X} = \mathbf{X}_1 + \ldots + \mathbf{X}_d. \tag{1.3}$$

The triple $(\sqrt{\lambda_j}, U_j, V_j)$ consisting of the singular value $\sigma_j = \sqrt{\lambda_j}$, the left singular vector U_j and the right singular vector V_j of $\mathbf{X}$ is called *jth eigentriple*.

Step 3: Grouping The input in this step is the expansion (1.2) and a specification of how to group the components of (1.2).

Let $I = \{i_1, \ldots, i_p\} \subset \{1, \ldots, d\}$ be a set of indices. Then the resultant matrix $\mathbf{X}_I$ corresponding to the group I is defined as $\mathbf{X}_I = \mathbf{X}_{i_1} + \ldots + \mathbf{X}_{i_p}$.

Assume that a partition of the set of indices $\{1, \ldots, d\}$ into m disjoint subsets $I_1, \ldots, I_m$ is specified. Then the result of Grouping step is the *grouped matrix decomposition*

$$\mathbf{X} = \mathbf{X}_{I_1} + \ldots + \mathbf{X}_{I_m}. \tag{1.4}$$

If only one subset, I, of $\{1, \ldots, d\}$ is specified, then we still can assume that a partition of $\{1, \ldots, d\}$ is provided: this is the partition consisting of two subsets, I and $\bar{I} = \{1, \ldots, d\} \setminus I$. In this case, $\mathbf{X}_I$ is usually associated with the pattern of interest (for example, signal) and $\mathbf{X}_{\bar{I}} = \mathbf{X} - \mathbf{X}_I$ can be treated simply as a residual.

The grouping of the expansion (1.2), where $I_k = \{k\}$, is called *elementary*.

Step 4: Reconstruction At this step, each matrix $\mathbf{X}_{I_k}$ from the decomposition (1.4) is transferred back to the form of the input object $\mathbb{X}$. This transformation is performed optimally in the following sense: for a matrix $\mathbf{Y} \in \mathsf{R}^{L \times K}$, we seek for an object $\widetilde{\mathbb{Y}} \in \mathcal{M}$ that provides the minimum to $\|\mathbf{Y} - \mathcal{T}(\widetilde{\mathbb{Y}})\|_{\mathrm{F}}$, where $\|\mathbf{Z}\|_{\mathrm{F}} = \left(\sum_{ij} |z_{ij}|^2\right)^{1/2}$ is the Frobenius norm of $\mathbf{Z} = [z_{ij}] \in \mathsf{R}^{L \times K}$.

Let $\varPi_{\mathcal{H}} : \mathsf{R}^{L \times K} \to \mathcal{M}^{(\mathrm{H})}_{L,K}$ be the orthogonal projection from $\mathsf{R}^{L \times K}$ to $\mathcal{M}^{(\mathrm{H})}_{L,K}$ in the Frobenius norm. Then $\widetilde{\mathbb{Y}} = \mathcal{T}^{-1} \circ \varPi_{\mathcal{H}}(\mathbf{Y})$. The projection $\varPi_{\mathcal{H}}$ is simply the averaging of the entries corresponding to a given element of an object, see Sect. 1.1.2.6 for details. For example, in 1D-SSA the composite mapping $\mathcal{T}^{-1} \circ \varPi_{\mathcal{H}}$ uses the averaging along anti-diagonals so that $\widetilde{y}_k = \sum_{(i,j) \in \mathcal{A}_k} (\mathbf{Y})_{ij} / |\mathcal{A}_k|$, where $\mathcal{A}_k = \{(i, j) : i + j = k + 1, 1 \leq i \leq L, 1 \leq j \leq K\}$.

Let $\widehat{\mathbf{X}}_k = \mathbf{X}_{I_k}$ be the reconstructed matrices, $\widetilde{\mathbf{X}}_k = \varPi_{\mathcal{H}} \widehat{\mathbf{X}}_k$ be the corresponding trajectory matrices, and $\widetilde{\mathbb{X}}_k = \mathcal{T}^{-1}(\widetilde{\mathbf{X}}_k)$ be the reconstructed objects. Then the resulting decomposition of the initial object $\mathbb{X}$ is

$$\mathbb{X} = \widetilde{\mathbb{X}}_1 + \ldots + \widetilde{\mathbb{X}}_m. \tag{1.5}$$

If the grouping is elementary, then the reconstructed objects $\widetilde{\mathbb{X}}_k$ in (1.5) are called *elementary components*.

For convenience of referencing, Steps 1 and 2 of the generic SSA scheme are sometimes combined into the so-called "Decomposition stage" and Steps 3 and 4 are combined into "Reconstruction stage."

1.1.2 The Main Concepts

1.1.2.1 Parameters of the SSA Methods

Step 1: parameters of the linear map $\mathcal{T}$. For a given object $\mathbb{X}$, the trajectory matrix $\mathbf{X} = \mathcal{T}(\mathbb{X})$ can be computed in different ways. In 1D-SSA, there is only one parameter in Step 1, the window length L.

Step 2: no parameters if the conventional SVD is performed. Otherwise, if an alternative decomposition of $\mathbf{X}$ into a sum of rank-one matrices is used, there may be some parameters involved, see Sect. 1.2.1.

Step 3: the parameter (or parameters) that defines the grouping.

Step 4: no extra parameters.

1.1.2.2 Separability

A very important concept in the SSA methodology is separability. Let $\mathbb{X} = \mathbb{X}_1 + \mathbb{X}_2$. (Approximate) separability of $\mathbb{X}_1$ and $\mathbb{X}_2$ means that there exists a grouping such that the reconstructed object $\widetilde{\mathbb{X}}_1$ is (approximately) equal to $\mathbb{X}_1$. The representation $\mathbb{X} = \mathbb{X}_1 + \mathbb{X}_2$ can be associated with many different models such as "signal plus noise," "trend plus the rest," and "texture plus the main image."

If $\mathbb{X} = \mathbb{X}_1 + \mathbb{X}_2$ and $\mathbb{X}_1$ and $\mathbb{X}_2$ are approximately separable, then SSA can potentially separate $\mathbb{X}_1$ from $\mathbb{X}_2$; that is, it can find a decomposition $\mathbb{X} = \widetilde{\mathbb{X}}_1 + \widetilde{\mathbb{X}}_2$ so that $\widetilde{\mathbb{X}}_1 \approx \mathbb{X}_1$ and $\widetilde{\mathbb{X}}_2 \approx \mathbb{X}_2$.

Consider, as an example, Basic SSA. Properties of the SVD yield that the (approximate) orthogonality of columns and of rows of the trajectory matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ of $\mathbb{X}_1$ and $\mathbb{X}_2$ can be considered as natural separability conditions.

There is a well-elaborated theory of separability for the one-dimensional time series (Golyandina et al. 2001; Sections 1.5 and 6.1). Many important decomposition problems, from noise reduction and smoothing to trend, periodicity and signal extraction, can be solved by SSA. The success of 1D-SSA in making separation between separable objects is related to the simplicity of the Hankel structure of the trajectory matrices and the optimality features of the SVD.

We will come back to the important concept of separability in Sect. 1.3 where we define the main characteristic which is used in SSA for separability checking.

1.1.2.3 Information for Grouping

The theory of SSA exhibits the ways of helping to detect the components (σ_i, U_i, V_i) in the decomposition (1.2) related to the object component with certain properties to perform proper grouping under the condition of separability. One of the rules is that U_i and V_i (eigenvectors and factor vectors in the case of Basic SSA) produced by an object component emulate the properties of this component. For example, in Basic 1D-SSA the eigenvectors produced by slowly-varying series components are slowly-varying, the eigenvectors produced by a sine wave are sine waves with the same frequency, and so on. These properties help to perform the grouping by visual inspection of eigenvectors and also by some automatic procedures, see Sect. 2.7.

1.1.2.4 Trajectory Spaces and Signal Subspaces

Let $\mathbf{X}$ be the trajectory matrix corresponding to some object $\mathbb{X}$. The *column (row) trajectory space* of $\mathbf{X}$ is the linear subspace spanned by the columns (correspondingly, rows) of $\mathbf{X}$. The term "trajectory space" usually means "column trajectory space." The column trajectory space is a subspace of R^L, while the row trajectory space is a subspace of R^K. In general, for real-world data the trajectory spaces coincide with the corresponding Euclidean spaces, since they are produced by a noisy data. However, in the "signal plus noise" model, when the signal has rank-deficient trajectory matrix, the signal trajectory space can be called "signal subspace." Both column and row signal subspaces can be considered; note that the dimensions of the row and column subspaces coincide.

1.1.2.5 Objects of Finite Rank

The class of objects that suit SSA are the so-called objects of finite rank. We say that the object (either time series or image) has L-rank r if the rank of its trajectory matrix is $r < \min(L, K)$; that is, the trajectory matrix is rank-deficient. If the L-rank r does not depend on the choice of L for any sufficiently large object and trajectory matrix sizes, then we say that the object is of finite rank and has rank r, see Sect. 2.1.2 for rigorous definitions.

Since the trajectory matrices considered in SSA methods are either pure Hankel or consist of Hankel blocks, then the rank-deficient trajectory matrices are closely related to the objects satisfying some linear relations. These linear relations can be used for building forecasting methods. In the one-dimensional case, under some non-restrictive conditions, rank-deficient Hankel matrices are in the one-to-one correspondence with the linear recurrence relations (LRRs) of the form

$$x_n = a_1 x_{n-1} + \ldots + a_r x_{n-r}$$

and therefore are related to the time series which can be expressed as sums of products of exponentials, polynomials, and sinusoids, see Sect. 2.1.2.2.

Each specific SSA extension produces a class of specific objects of finite rank. The knowledge of ranks of objects of finite rank can help to recognize the rank-one components for the component reconstruction. For example, in order to reconstruct the exponential trend in the one-dimensional case, we need to group only one rank-one component (the exponential function has rank 1), while to reconstruct a sine wave we generally need to group two SVD components (the rank of a sine wave equals 2).

The real-life time series or images are generally not of finite rank. However, if a given object $\mathbb{X}$ is a sum of a signal of finite rank and noise, then, in view of approximate separability, SSA may be able to approximately extract the signal and subsequently use the methods that are designed for series of finite rank.

1.1.2.6 Reconstruction (Averaging)

Let us formally describe the operation of reconstruction of a matrix used in Step 4 of the generic scheme described in Sect. 1.1.1. By analogy with the one-dimensional case this operation can also be called "averaging over diagonals" even if the averaging will be performed over more complicated patterns.

Assume that the entries x_τ of the object $\mathbb{X} = \{x_\tau\}$ are indexed by the index τ which can be simply a positive integer (for the one-dimensional series) or multi-index (for digital images).

A linear map $\mathcal{T}$ is making a one-to-one transformation of $\mathcal{M}$ to $\mathcal{M}^{(\mathrm{H})}_{L,K}$, the set of $L \times K$ matrices of a specified structure. It puts elements of $\mathbb{X}$ on certain places of the matrix $\mathcal{T}(\mathbb{X}) = \mathbf{X} = [(\mathbf{X})_{ij}]$.

Let $\mathbf{e}_\tau \in \mathcal{M}$ be the object with 1 as the τth entry with all other entries zero. Define the set of indices

$$\mathcal{A}_\tau = \{(i, j) \text{ such that } (\mathbf{E}_\tau)_{ij} = 1\},$$

where $\mathbf{E}_\tau$ is the matrix

$$\mathbf{E}_\tau = [(\mathbf{E}_\tau)_{ij}] = \mathcal{T}(\mathbf{e}_\tau) \in \mathcal{M}^{(\mathrm{H})}_{L,K}.$$

If τ is the place of an element $x_\tau \in \mathbb{X}$, then $(\mathbf{X})_{ij} = x_\tau$ for all $(i, j) \in \mathcal{A}_\tau$.

Assume now that $\widehat{\mathbf{X}} \in \mathsf{R}^{L\times K}$ is an arbitrary $L \times K$ matrix and we need to compute

$$\widetilde{\mathbb{X}} = (\widetilde{x}_\tau) = \mathcal{T}^{-1} \circ \Pi_{\mathcal{H}}(\widehat{\mathbf{X}})$$

by first making the orthogonal projection of $\widehat{\mathbf{X}}$ to the set $\mathcal{M}^{(\mathrm{H})}_{L,K}$ and then writing the result in the object space $\mathcal{M}$. This operation is the extension of the "diagonal averaging" procedure applied in 1D-SSA: the elements $\widetilde{x}_\tau$ of $\widetilde{\mathbb{X}}$ are computed by the formula

$$\widetilde{x}_\tau = \sum_{(i,j)\in\mathcal{A}_\tau} (\widehat{\mathbf{X}})_{ij}/w_\tau = \langle\widehat{\mathbf{X}}, \mathbf{E}_\tau\rangle_\mathrm{F}/\|\mathbf{E}_\tau\|_\mathrm{F}^2\,,$$

where $\mathbf{E}_\tau = \mathcal{T}(\mathbf{e}_\tau)$, $w_\tau = |\mathcal{A}_\tau| = \|\mathbf{E}_\tau\|_\mathrm{F}^2$ is the number of elements in the set $\mathcal{A}_\tau$ and the Frobenius inner product $\langle\cdot,\cdot\rangle_\mathrm{F}$ is defined by

$$\langle\mathbf{Z}, \mathbf{Y}\rangle_\mathrm{F} = \sum_{i,j} z_{ij}y_{ij}\,. \tag{1.6}$$

1.2 Different Versions of SSA

Let us consider how the four steps of the generic scheme of SSA formulated in Sect. 1.1 can vary for different versions of SSA.

Step 1: the form of the object $\mathbb{X}$ and hence the specificity of the embedding operator $\mathcal{T}$ makes a big influence on how a particular version of SSA looks like.
Step 2: not only the conventional SVD but many other decompositions of $\mathbf{X}$ into rank-one matrices could be used.
Step 3: formally, this step is exactly the same for all versions of SSA although the tools used to perform the grouping may differ.
Step 4: the embedding operator $\mathcal{T}$ defined in Step 1 fully determines the operations performed at this step.

Therefore, we have two directions for creating different versions of SSA: the first direction is related to geometrical features of the object $\mathbb{X}$ and a form of the embedding operator $\mathcal{T}$, while the second direction is determined by the form of the decomposition at Step 2. Essentially, Step 1 is determined by the form of the object; therefore, its variations can be considered as extensions of 1D-SSA. If instead of SVD we use some other decomposition of $\mathbf{X}$ into rank-one matrices at Step 2, then we call the corresponding algorithm a modification of Basic SSA. These two directions of variations of SSA are not related to each other so that any extension of Step 1 can be combined with any modification of Step 2.

We start the discussion with outlining some modifications that can be offered for the use at Step 2.

1.2.1 Decomposition of X *into a Sum of Rank-One Matrices*

1.2.1.1 Variations of SSA Related to Methods of Decomposition

The conventional SVD formulated in the description of SSA in Sect. 1.1.1 is a decomposition of $\mathbf{X}$ into a sum of rank-one matrices, which has some optimality properties, see Golyandina et al. (2001; Chapter 4). Therefore, Basic SSA, which is SSA with the conventional SVD used at Step 2, can be considered as the most fundamental version of SSA among all SSA methods.

Let us enumerate several variations of SSA, which could be useful for answering different questions within the framework of SSA.

A well-known modification of Basic SSA is *Toeplitz SSA* (Sect. 2.2), which was created for dealing with stationary time series. This modification is devised for the analysis of a natural estimate of the auto-covariance matrix of the original time series and assumes that this time series is stationary. However, if the time series $\mathbb{X}$ is non-stationary, then the reconstruction obtained by Toeplitz SSA can have a considerable bias.

An important variation of SSA is *SSA with projection* (Sect. 2.3). If we have a parametric model (which should be linear in parameters and agreeable with the finite-rank assumption) for one of the components of the series, such as trend of a one-dimensional series, then a projection on a suitable subspace is performed and is followed by a decomposition of the residual, e.g., by the SVD. The known methods of SSA with centering and SSA with double centering for extraction of constant and linear trends, respectively, are special cases of SSA with projection. More generally, an arbitrary polynomial trend can be extracted by a suitable version of SSA with projection. Another use of SSA with projection is to build a subspace from a supporting series and project the main series onto this subspace.

In some versions of SSA the intention is to improve separability properties of SVD. If we use an oblique version of the SVD, then the resulting SSA method becomes Oblique SSA. The following two versions of Oblique SSA seem to be useful in practice, namely *Iterative Oblique SSA* (Sect. 2.4) and *Filter-adjusted Oblique SSA* (Sect. 2.5). The latter is useful for separation of components with equal contribution.

1.2.1.2 Nested Application of Different Versions of SSA

Since Oblique SSA does not have good approximating features, it cannot replace Basic SSA which uses the conventional SVD. Therefore, Oblique SSA should be used in a nested manner so that Basic SSA is used first to extract several components without performing careful split of these components and then one of the proposed oblique methods is used for separating the mixed components.

If we use Basic SSA for denoising and then some other version of SSA (like Independent Component Analysis or Oblique SSA) for improvement of separability, then we can interpret this as if using Basic SSA for preprocessing and using another method for a more refined analysis. There is, however, a significant difference between this and some methods mentioned in Sect. 1.7.3, where Basic SSA is used for preprocessing and then ARIMA or other methods of different nature are employed. Indeed, when we use Basic SSA for denoising and some other SSA technique like Oblique SSA for improvement of separability, then we are using the signal subspace estimated by Basic SSA rather than the estimated signal itself (recall that in the transition from the estimated signal subspace to the estimated signal we incur an additional error).

Let us schematically demonstrate the nested use of the methods as follows. Let $\mathbb{X} = \mathbb{X}^{(1)} + \mathbb{X}^{(2)} + \mathbb{X}^{(3)}$ be a decomposition of the time series and $\mathbf{X} = \mathbf{X}^{(1)} + \mathbf{X}^{(2)} + \mathbf{X}^{(3)}$ be the corresponding decomposition of the trajectory matrix of $\mathbb{X}$. Let Basic SSA return at Decomposition stage $\mathbf{X} = \widetilde{\mathbf{X}}^{(1,2)} + \widetilde{\mathbf{X}}^{(3)}$ and assume that a chosen nested method makes the decomposition $\widetilde{\mathbf{X}}^{(1,2)} = \widetilde{\mathbf{X}}^{(1)} + \widetilde{\mathbf{X}}^{(2)}$. Then the final result is $\mathbf{X} = \widetilde{\mathbf{X}}^{(1)} + \widetilde{\mathbf{X}}^{(2)} + \widetilde{\mathbf{X}}^{(3)}$ and, after the diagonal averaging, $\mathbb{X} = \widetilde{\mathbb{X}}^{(1)} + \widetilde{\mathbb{X}}^{(2)} + \widetilde{\mathbb{X}}^{(3)}$. There is no need for reconstruction of the signal by Basic SSA as only the estimated signal subspace is used for making a refined decomposition.

1.2.1.3 Features of Decompositions

The result of Decomposition step of SSA (Step 2) can be written in the form (1.2). The SVD is a particular case of (1.2) and corresponds to the orthonormal systems of $\{U_i\}$ and $\{V_i\}$. By analogy with the SVD, we will call (σ_i, U_i, V_i) eigentriples, σ_i singular values, U_i left singular vectors or eigenvectors, V_i right singular vectors or factor vectors. For most of SSA decompositions, each U_i belongs to the column space of $\mathbf{X}$ while each V_i belongs to the row space of $\mathbf{X}$. We shall call such decompositions *consistent*.

If the systems $\{U_i\}$ and $\{V_i\}$ are linearly independent, then the decomposition (1.2) is *minimal*; that is, it has smallest possible number of addends equal to $r = \operatorname{rank}\mathbf{X}$. If at least one of the systems $\{U_i\}$ or $\{V_i\}$ is not linearly independent, then the decomposition (1.2) is not minimal. If the decomposition (1.2) is not consistent, then it can be not minimal even if $\{U_i\}$ or $\{V_i\}$ are linearly independent, since their projections on the column (or row) space can be dependent.

If both vector systems $\{U_i\}$ and $\{V_i\}$ are orthogonal, then the decomposition (1.2) is called biorthogonal. If $\{U_i\}$ is orthogonal, then the decomposition is called left-orthogonal; if $\{V_i\}$ is orthogonal, then the decomposition is called right-orthogonal.

If $\mathbf{X}_i$ are F-orthogonal so that $\langle \mathbf{X}_i, \mathbf{X}_j \rangle_{\mathrm{F}} = 0$ for $i \neq j$, then we say that the corresponding decompositions are F-orthogonal. Either left or right orthogonality is sufficient for F-orthogonality. For F-orthogonal decompositions (1.2), $\|\mathbf{X}\|_{\mathrm{F}}^2 = \sum_i \|\mathbf{X}_i\|_{\mathrm{F}}^2$. In general, however, $\|\mathbf{X}\|_{\mathrm{F}}^2$ may differ from $\sum_i \|\mathbf{X}_i\|_{\mathrm{F}}^2$.

The contribution of kth matrix component $\mathbf{X}_k$ is defined as $\sigma_k^2 / \|\mathbf{X}\|_{\mathrm{F}}^2$, where $\sigma_k^2 = \|\mathbf{X}_k\|_{\mathrm{F}}^2$. For F-orthogonal decompositions, the sum of component contributions is equal to 1. Otherwise, this sum can considerably differ from 1 (e.g., the sum of component contributions can be 146%).

1.2.1.4 Decompositions in Different Versions of SSA

Let us gather several versions of 1D-SSA which are implemented in the RSSA package and based on different procedures used at Decomposition step, and indicate their features. Some of these variations are also implemented for multivariate and multidimensional versions of SSA.

Basic SSA: the conventional SVD, consistent, minimal, biorthogonal and therefore F-orthogonal decomposition. Implemented in `ssa` with `kind="1d-ssa"`.

Toeplitz SSA: generally, non-consistent, non-minimal F-orthogonal decomposition. Implemented in `ssa` with `kind="toeplitz-ssa"`.

SSA with projection: F-orthogonal but non-consistent decomposition if at least one basis vector used for the projection does not belong to the column (row) trajectory space. The components, which are obtained by projections, are located at the beginning of the decomposition and have indices $1, \ldots, n_{\text{special}}$. Implemented in `ssa` with `kind="1d-ssa"` and non-`NULL` `row.projector` or `column.projector` arguments.

Oblique SSA with filter preprocessing (Filter-adjusted O-SSA): consistent, minimal F-orthogonal decomposition. The main particular case is DerivSSA. Implemented in `fossa`.

Iterative Oblique SSA (Iterative O-SSA): consistent, minimal oblique decomposition. Implemented in `iossa`.

Oblique versions of SSA are made to perform in a nested manner.

1.2.2 Versions of SSA Dealing with Different Forms of the Object

In this section, we briefly consider different versions of SSA which operate objects $\mathbb{X}$ of different forms. The main difference between different versions of SSA of this section is the form of the embedding operator $\mathcal{T}$.

As has been mentioned above, SSA can be applied to multivariate and multidimensional objects. SSA for a system of series is called Multivariate (or Multichannel) SSA, shortly MSSA (Sect. 4.2); SSA for digital gray-scale images is called 2D-SSA (Sect. 5.1).

Complex SSA (Sect. 4.1) is a special version of SSA for the analysis of two time series of equal length or a single one-dimensional complex-valued time series.

Shaped SSA (Sects. 2.6 and 5.2) can process data of complex structure and arbitrary shape; the dimension of the object $\mathbb{X}$ is irrelevant. Shaped SSA can be applied to many different kinds of data including time series, systems of time series, digital images of rectangular and non-rectangular shapes.

For both series and images, circular versions of SSA are available. For series, circular SSA works in the metric of a circle and therefore this version is suitable for series, which are indeed defined on a circle.

For images, circular versions of SSA provide a possibility to decompose images given on a cylinder (for example, obtained as a cylindrical projection of a sphere or an ellipsoid) or given on a torus. Circular versions allow to eliminate the edge effects, which are unavoidable in the case of, e.g., planar unfolding of a cylinder.

Table 1.1 contains a list of the extensions considered in this book.

Table 1.1 Classification of different versions of SSA based on different geometrical features of the object $\mathbb{X}$

Method	Data	Notation	Trajectory matrix	Section
1D-SSA	Time series (t.s.)	$\mathbb{X} = (x_1, \ldots, x_N)$ of length N	Hankel	2.1
CSSA	Complex t.s.	$\mathbb{X}^{(1)} + \mathrm{i}\,\mathbb{X}^{(2)}$	Complex Hankel	4.1
MSSA	System of t.s.	$\mathbb{X}^{(p)}$ of length N_p, $p = 1, \ldots, s$	Stacked Hankel	4.2
2D-SSA	Rectangular image	$\mathbb{X} = (x_{ij})_{i,j=1}^{N_x,N_y}$	Hankel-Block-Hankel	5.1
ShSSA	Shaped objects	$\mathbb{X} = (x_{(i,j)})_{(i,j)\in\mathfrak{N}}$	Quasi-Hankel	5.2
Circular SSA	Circular/cylindrical	Objects on circle or cylinder	Quasi-Hankel	5.2.1.3

1.3 Separability in SSA

In this section, we discuss the SSA separability in more detail; see also Sects. 2.1.3, 2.3.3, 2.4.2, 2.5.2 for special cases.

Let us assume that we observe a sum of two objects $\mathbb{X} = \mathbb{X}^{(1)} + \mathbb{X}^{(2)}$. We say that SSA separates these two objects if a grouping at Grouping step (Step 3) can be found such that $\widetilde{\mathbb{X}}^{(k)} = \mathbb{X}^{(k)}$ for $k = 1, 2$. If these equalities hold approximately, then this defines approximate separability. Asymptotic separability can be introduced if the series length $N \to \infty$. In this case, approximate separability takes place for large enough series lengths. The separability property is very important for SSA as it means that the method potentially works; that is, it is able to extract the object components.

In Basic SSA and its multidimensional extensions, (approximate) separability means (almost) orthogonality of the object components, since the biorthogonal SVD decomposition is used. In other versions of SSA, different conditions of separability can be formulated.

If the decomposition (1.2) at Decomposition step is not unique, then two variants of separability are introduced. Weak separability means that there exists a decomposition such that after grouping we obtain $\widetilde{\mathbb{X}}^{(k)} = \mathbb{X}^{(k)}$. Strong separability means that this equality is achievable for any admissible decomposition.

Conditions of exact separability are very restrictive whereas asymptotic separability takes place for a wide range of object components. For example, slowly varying smooth components are asymptotically separable from regular oscillations and they both are asymptotically separable from noise.

In order to verify separability of the reconstructed components $\widetilde{\mathbb{X}}_1$ and $\widetilde{\mathbb{X}}_2$, we should check orthogonality of their reconstructed trajectory matrices $\widetilde{\mathbf{X}}_1$ and $\widetilde{\mathbf{X}}_2$. A convenient measure of their orthogonality is the Frobenius inner product $\langle \widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_2 \rangle_{\mathrm{F}}$ defined in (1.6).

The normalized measure of orthogonality is

$$\rho(\widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_2) = \langle \widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_2 \rangle_{\mathrm{F}} / (\|\widetilde{\mathbf{X}}_1\|_{\mathrm{F}} \|\widetilde{\mathbf{X}}_2\|_{\mathrm{F}})\,.$$

Since the trajectory matrix consists of $w_\tau = |\mathcal{A}_\tau| = \|\mathcal{T}(\mathbf{e}_\tau)\|_\mathrm{F}^2$ entries of the τth element x_τ of the initial ordered object, we can introduce the weighted inner product in the space $\mathcal{M}$: $(\mathbb{Y}, \mathbb{Z})_\mathbf{w} = \sum_\tau w_\tau y_\tau z_\tau$. Then the quantity

$$\rho_\mathbf{w}(\widetilde{\mathbb{X}}_1, \widetilde{\mathbb{X}}_2) = \rho(\widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_2) = \frac{(\widetilde{\mathbb{X}}_1, \widetilde{\mathbb{X}}_2)_\mathbf{w}}{\|\widetilde{\mathbb{X}}_1\|_\mathbf{w} \|\widetilde{\mathbb{X}}_2\|_\mathbf{w}} \tag{1.7}$$

will be called **w**-*correlation* by statistical analogy. Note however that in this definition the means are not subtracted.

Let $\widetilde{\mathbb{X}}_j$ be the elementary reconstructed components produced by the elementary grouping $I_j = \{j\}$. Then the matrix of $\rho_{ij}^{(\mathbf{w})} = \rho_\mathbf{w}(\widetilde{\mathbb{X}}_i, \widetilde{\mathbb{X}}_j)$ is called **w**-*correlation matrix*.

The norm $\|\cdot\|_\mathbf{w}$ is called the *weighted norm* and serves as a measure of contribution of the components in the decomposition (1.5): the contribution of $\widetilde{\mathbb{X}}_k$ is defined as $\|\widetilde{\mathbb{X}}_k\|_\mathbf{w}^2 / \|\mathbb{X}\|_\mathbf{w}^2$.

If the weighted correlation between a pair of elementary components is large, then this suggests that these two components are highly correlated and should perhaps be included into the same group.

1.4 Forecasting, Interpolation, Low-Rank Approximation, and Parameter Estimation in SSA

There is a class of objects, which is special for SSA. This is the class of objects satisfying linear recurrence relations. Trajectory matrices of these objects are rank-deficient and, moreover, for these objects the number of non-zero terms in the SVD (1.3) does not depend on the window length if this length is sufficiently large; we will say in such cases that the objects are of finite rank. The class of objects satisfying linear recurrence relations provides a natural model of the signal for SSA, which is a fundamentally important concept for forecasting of time series.

Linear recurrence relation (LRR) for a time series $\mathbb{S}_N = (s_i)_{i=1}^N$ is a relation of the form

$$s_{i+t} = \sum_{k=1}^{t} a_k s_{i+t-k}, \ 1 \le i \le N-t, \ a_t \ne 0, \ t < N. \tag{1.8}$$

It is well known (see, e.g., Hall (1998; Theorem 3.1.1)) that a sequence $\mathbb{S}_\infty = (s_1, \ldots, s_n, \ldots)$ satisfies the LRR (1.8) for all $i \ge 0$ if and only if for some integer p we have for all n

$$s_n = \sum_{k=1}^{p} P_k(n) \mu_k^n, \tag{1.9}$$

where $P_k(n)$ are polynomials in n and μ_k are some complex numbers. For real-valued time series, (1.9) implies that the class of time series governed by the LRRs consists of sums of products of polynomials, exponentials, and sinusoids.

A simplified model of (1.9) is $s_n = \sum_{k=1}^{p} c_k \mu_k^n$. Estimation of complex numbers $\mu_k = \rho_k e^{\mathrm{i}2\pi\omega_k}$ is equivalent to estimating the frequencies ω_k and the rates $\ln \rho_k$.

For images $\mathbb{S} = [s_{mn}]$, LRRs are two-dimensional and the common term of the signal (which can be called pattern) under the model has the form

$$s_{mn} = \sum_{k=1}^{p} P_k(m, n)\mu_k^m \nu_k^n. \tag{1.10}$$

This important fact is well known starting from Kurakin et al. (1995; §2.20).

In many real-life problems, a noisy signal (or noisy pattern for images) is observed and the problem is to forecast the signal, impute gaps in the signal, estimate signal parameters, find change-points in the signal, and so on. Note that it is not compulsory to assume that the noise is random. In a very general sense, noise is a residual which does not require further investigation.

SSA may provide estimates of the signal space, which is the subspace spanned by the chosen basis during Grouping step in SSA. The estimation of the signal subspace can be performed by iterations (so-called Cadzow iterations), which consist of iterative SSA processing, see Sect. 3.4.

In the process of estimation of the signal subspace we also obtain a parametric estimate of the signal; that is, of the set of $\{\mu_k\}$ in the 1D case and the set of $\{\mu_k, \nu_k\}$ in the 2D case. One of the most common methods is called ESPRIT for series and 2D-ESPRIT for images (see Sects. 3.1.1.2 and 5.3).

For series, sequential on-line estimation of the signal structure produces natural algorithms of subspace tracking (e.g., for monitoring structural stability of an object changing in time) and change-point detection. Also, we can forecast the found structure, e.g., by application of the constructed LRR. By performing interpolation (which is forecasting inside the series) missing data can be filled in.

1.5 The Package

1.5.1 SSA Packages

There are many implementations of SSA. They differ by potential application areas, implemented methods, interactive or non-interactive form, free or commercial use, computer system (Windows, Unix, Mac), level of reliability and support. The most known supported software packages implementing SSA are the following:

1. http://gistatgroup.com:
   general-purpose interactive "Caterpillar"-SSA software (Windows) following the methodology described in Golyandina et al. (2001), Golyandina and Zhigljavsky (2013);
2. http://www.atmos.ucla.edu/tcd/ssa:
   oriented mainly on climatic applications SSA-MTM Toolkit for spectral analysis (Ghil et al. 2002) (Unix) and its commercial extension kSpectra Toolkit (Mac), interactive;
3. The commercial statistical software, SAS, has an econometric extension called SAS/ETS®, which includes SSA in its rather basic form; this version of SSA is based on the methodology of Golyandina et al. (2001).
4. http://cran.r-project.org/web/packages/Rssa:
   R package Rssa (Korobeynikov et al. 2017), an implementation of the main SSA procedures for major platforms, extensively developed.

We consider the Rssa package as an efficient implementation of a large variety of the main SSA algorithms. This package also contains many visual tools which are useful for making proper choice of SSA parameters and examination of results.

Rssa is the only SSA package available from CRAN and we believe it is the fastest implementation of SSA. Another important feature of the package is its very close relation to the SSA methodology thoroughly described in Golyandina et al. (2001), Golyandina and Zhigljavsky (2013), Golyandina et al. (2015). As a result of this, the use of the package is well theoretically and methodologically supported.

1.5.2 Tools for Visual Control and Choice of Parameters

SSA needs tools to help choosing the method parameters and controlling the results. To a great extent, SSA is an exploratory technique and hence visual tools are vital and they are extensively used in Rssa. For example, in order to help to choose the groups in (1.4), Rssa allows plotting measures of separability of series components in the obtained decompositions.

The tools for accuracy control are divided into two groups. First, stability of results with respect to parameter changes can be checked. Second, the bootstrap procedure could be used when the model (not necessarily parametric) of either the series or other object is built on the base of signal reconstruction and then the accuracy of this model is assessed by simulation according to the estimated model. Shortly, Rssa allows the user to enjoy a large variety of graphical tools and bootstrap procedures.

1.5.3 Short Introduction to Rssa

The Rssa package implements all methods and tools mentioned above.

The main function is `ssa`, which initializes an `ssa` object and by default performs the decomposition by different methods. Together with `reconstruct`, they implement the SSA method. For nested versions, `iossa` and `fossa` serve for refined decompositions.

An `ssa` object `s` contains, among others, elements of the decomposition (1.2), which can be accessed as `s$sigma`, `s$U`, and `s$V`. Features of the decompositions differ for different versions of SSA (see Sect. 1.2.1.4). For Basic SSA, `s$sigma` are called singular values; squares of `s$sigma` are called eigenvalues; `s$U` are called eigenvectors. (We keep these names for other versions of SSA as well.) The relative contributions of components to the decomposition can be obtained as `contributions(s)`; see Sect. 1.2.1.3, where formulas for their calculation are given and explained.

A variety of functions `plot` help to visualize the results and additional information. Functionality of SSA-related methods is supplemented by the functions `forecast`, `parestimate`, and some others.

All essential versions of SSA are implemented in Rssa but not all further actions like forecasting and gap filling are consistent with all implemented versions of SSA. The user can check the `ssa` object, which is returned by the main function `ssa`, for compatibility by the function `ssa.capabilities`. This function returns a list of capabilities with information `TRUE` or `FALSE`, respectively.

A general scheme of investigation by means of Rssa is as follows:

1. perform decomposition by `ssa`;
2. visualize the result by `plot`;
3. if necessary, refine decomposition by `iossa` or `fossa`;
4. again, visualize the result by `plot`;
5. perform grouping based on the obtained visual and numerical information; in particular, choose the group of signal components;
6. then perform one of the following actions: reconstruction of series components by `reconstruct`, forecasting by `forecast`, parameter estimation by `parestimate`;
7. visualize the result by `plot`.

Note that Rssa contains more algorithms than this book formulates. However, the book has enough information to understand how to extend the algorithms, such as parameter estimation, filling-in missing data and O-SSA, to different dimensions and geometries. Many of these versions are implemented in Rssa but not described in the book explicitly: there are infinitely many dimensions and geometries and the algorithms are formulated in the book in such a manner that they can be easily generalized if needed.

1.5.4 Implementation Efficiency

The user does not need to know the specifics of the internal implementation of the Rssa functions. However, understanding of the general principles of implementation can help to use the package more effectively.

The fast implementation of SSA-related methods, which was suggested in Korobeynikov (2010), extended in Golyandina et al. (2015) and is used in the Rssa package (Korobeynikov et al. 2017), relies on the following techniques (see Shlemov and Golyandina (2014) for a more thorough discussion).

1. The truncated SVD calculated by the Lanczos methods (Golub and Van Loan 1996; Ch. 9) is used. In most SSA applications, only a number of leading SVD components correspond to the signal and therefore are used at Grouping step of the SSA algorithm. Thus, a truncated SVD rather than the full SVD is usually required by SSA.
2. Lanczos methods do not use the explicit representation of the decomposed matrix $\mathbf{A}$. They need only the results of multiplication of $\mathbf{A}$ and $\mathbf{A}^{\mathrm{T}}$ on some vectors. In view of the special Hankel-type structure of $\mathbf{A}$ in the SSA algorithms, multiplication by a vector can be implemented very efficiently with the help of the Fast Fourier Transform (FFT). Fast SVD algorithms are implemented in the R-package svd (Korobeynikov et al. 2016) in such a way that their input is the function of a vector which yields the result of fast multiplication of the vector by the trajectory matrix. Therefore, the use of svd in Rssa allows a fast and space-efficient implementation of the SSA algorithms.
3. At Reconstruction step, hankelization or quasi-hankelization of a matrix of rank 1, stored as $\sigma U V^{\mathrm{T}}$, can be written by means of the convolution operator and therefore can also be effectively implemented; this is also done in Rssa.

The overall complexity of the computations is $O(kN\log(N) + k^2N)$, where N is the number of elements in a shaped object and k is the number of considered eigentriples, see details in Korobeynikov (2010) and in Golyandina et al. (2015). This makes the computations feasible for large data sets and large window sizes. For example, the case of an image of size 299 × 299 and a window size 100 × 100 can be handled rather easily, whereas the conventional algorithms (e.g., the full SVD (Golub and Van Loan 1996)) are impractical, because the matrix that needs to be decomposed has size $10^4 \times 4 \cdot 10^4$. Using larger window sizes is often advantageous, since, for example, separability of signal from noise (in the "signal+noise" scenario) can be significantly improved.

1.6 Comparison of SSA with Other Methods

In this section, we provide some notes and a short bibliographical overview concerning comparison of SSA with several traditional methods.

1.6.1 Fourier Transform, Filtering, Noise Reduction

- Fourier Transform uses a basis given in advance, while SSA uses an adaptive basis, which is not restricted to a frequency grid with resolution $1/N$. The wavelet transform also uses fixed bases; the advantage of the wavelet transform is that a change in the frequencies can be detected by the used time-space basis. In the framework of SSA, the analysis of time series with changing frequency structure can be performed by using moving procedures, e.g., by subspace tracking.
- One of the state-of-the-art methods of frequency estimation is the high-resolution method ESPRIT, which is a subspace-based method. This method can be considered as an SSA-related method and indeed it is frequently used in the present book and in RSSA.
- Fourier Transform is very inefficient for series with modulations in amplitudes and frequencies. SSA can easily deal with amplitude modulation but cannot efficiently deal with frequency modulation.
- SSA decomposition can sometimes be viewed as an application of a set of linear filters (Bozzo et al. 2010; Harris and Yan 2010; Golyandina and Zhigljavsky 2013) with an interpretation depending on the window length L. For small L, each decomposition component on the interval $[L, K]$, where $K = N - L + 1$, can be obtained by a linear filter. Therefore, the viewpoint of filtering on the decomposition result can be adequate. For example, the reconstruction by the leading components is close to application of the triangle filter.

  If $L \simeq K$ and hence the interval $[L, K]$ is small, then it is not so. In this case, the separability approach, which is based on orthogonality of separated components, is more appropriate. Note that oblique versions of SSA can weaken the condition of orthogonality, see Sect. 2.4.
- There is a big difference between the moving averaging and SSA for noise reduction. Consider an example of a noisy sinusoid. The moving averaging will add a bias in estimation caused by the second derivative of the signal, while SSA with large L will provide an unbiased estimate of the signal.

  Note that even for small L, when the reconstruction by the leading component is a weighted moving average with positive weights and therefore has the same drawbacks as the moving averaging, the user can add additional components to remove the possible bias.
- Filtering by SSA to obtain noise reduction can be considered from the viewpoint of the low-rank approximation. The good approximation properties yield appropriate noise suppression. Empirical-mode decomposition (EMD), in turn, starts Intrinsic Mode Functions (IMF) with high frequencies, while the trend is contained in the last IMFs.

As an example of comparison of SSA, Fourier transform and wavelet transform see, e.g., Kumar et al. (2017). The authors conclusion states: “the SSA-based filtering technique is robust for regional gravity anomaly separation and could be effectively exploited for filtering other geophysical data.”

In Barrios-Muriel et al. (2016), an SSA-based de-noising technique for removal of electrocardiogram interference in Electromyography signals is compared with the high-pass Butterworth filter, wavelets, and EMD. The authors of this paper state: “the proposed SSA approach is a valid method to remove the ECG artifact from the contaminated EMG signals without using an ECG reference signal.”

In Watson (2016), many different methods for trend extraction are compared for synthetic data simulating sea level behavior; SSA is compared against moving average, wavelets, regression, EMD. The author writes: “the optimum performing analytic is most likely to be SSA whereby interactive visual inspection (VI) techniques are used by experienced practitioners to optimize window length and component separability.”

Comparison of SSA filtering and Kalman filters (KF) can be found in Chen et al. (2016), where it is shown that “both SSA and KF obtain promising results from the stations with strong seasonal signals, while for the stations dominated by the long-term variations, SSA seems to be superior.”

1.6.2 Parametric Regression

Parametric regression naturally assumes a parametric model. There is a big difference between parametric and non-parametric models: if the assumed model is true, then the related parametric methods are the most appropriate methods (if there are no outliers in the data). If the assumed parametric model is not true, then the results of parametric methods are biased and may be very misleading. Drawbacks of non-parametric methods are also clear: there are problems with forecasting, testing the model, confidence interval construction, and so on. Frequently, non-parametric methods serve as preprocessing tools for parametric methods. As discussed in Sect. 1.7.3, this is often the case for SSA.

For comparison of SSA with double centering and linear regression see, for example, Sect. 2.3 and Golyandina and Shlemov (2017). It appears that SSA with double centering as preprocessing method considerably improves the accuracy of linear trend estimation.

SSA has a very rare advantageous property: it can be a non-parametric method for preliminary analysis and can also be parametric for modeling the series governed by LRRs. Moreover, the forecast by an LRR uses the parametric model in implicit manner; therefore, it is more robust to deviations from the model than the forecast based on explicit parameter estimation.

One of the subspace-based method for constructing the model of the signal, which is governed by an LRR, is Hankel low-rank approximation (HLRA). HLRA can be considered as a method of parameter estimation in a parametric model, where only the rank of the signal is given rather than exact parametric form, see Sect. 3.4.

1.6.3 ARIMA and ETS

First, the (Seasonal) ARIMA and Exponential smoothing models (ETS, which means Error, Trend, Seasonal) totally differ from the model of SSA (for a comprehensive introduction to ARIMA and ETS, see Hyndman and Athanasopoulos (2013)). In particular, in ARIMA the noise is added at each recurrence step, while for SSA the noise is added after the signal is formed. Also, trends/seasonality in SSA are deterministic, while in ARIMA/ETS the trends/seasonality are random. As in many classical methods, ARIMA and ETS need the period values to be specified for the periodic components.

However, if one considers the analysis/forecast of real-life time series, then these time series do not exactly follow any model. Therefore, the problem of comparison of methods of different nature is not easy.

As a rule, confidence intervals for ARIMA forecasts are too large but the mean forecast can often be adequate. Advantage of Seasonal ARIMA and ETS is that the model and its parameters can be fitted automatically on the base of information criteria.

Rigorously substantiated information criteria are not constructed for SSA. One of the reasons for this is the fact that SSA is a non-parametric method. The most standard approach for the choice of parameters, when there are no given models, is the minimization of the forecasting error on the validation period. In the most frequent case, when the forecast is constructed on the base of r leading eigentriples, SSA has only two parameters (L and r), which can be estimated by the minimization of the forecasting error for several forecasts performed within the validation period, see Sect. 3.5.7.

Comparison of SSA and ARIMA/ETS was performed in many papers. Some examples are as follows.

- It is demonstrated in Hassani et al. (2015) that SSA has topped several other methods in an example involving forecasting of tourist arrivals,
- It is exhibited in Vile et al. (2012) that for predicting ambulance demands “SSA produces superior longer-term forecasts (which are especially helpful for EMS planning), and comparable shorter-term forecasts to well established methods.”
- The author of Iqelan (2017) concludes: “The forecasting results are compared with the results of exponential smoothing state space (ETS) and ARIMA models. The three techniques do similarly well in forecasting process. However, SSA outperforms the ETS and ARIMA techniques according to forecasting error accuracy measures.”
- In Hassani et al. (2009, 2013), the univariate and multivariate SSA were favorable in a comparison with ARIMA and VAR for forecasting of several series of European industrial production.

1.7 Bibliographical Notes

1.7.1 Short History

Commencement of SSA is usually associated with publication in 1986 of the papers Broomhead and King (1986) and Broomhead and King (1986b). However, some ideas, which later became parts of SSA, have been formulated very long ago (de Prony 1795). Arguably, the most influential papers on SSA published in the 1980 and 1990s, in addition to Broomhead and King (1986,b), are Fraedrich (1986), Vautard and Ghil (1989), Vautard et al. (1992), Allen and Smith (1996). In view of many successful applications of SSA, the number of publications considering SSA methodology grows exponentially and has surely reached few hundred.

A parallel development of SSA (under the name "Caterpillar") has been conducted in the former USSR, especially, in St.Petersburg (known at that time as Leningrad), see, e.g., Danilov and Zhigljavsky (1997). The authors of this book continue the traditions of the St.Petersburg school of SSA.

The monograph (Golyandina et al. 2001) contains a comprehensive description of the theoretical and methodological foundations of SSA for one-dimensional (1D) time series; the authors of that monograph tried to summarize all the knowledge about 1D-SSA available at that time. A short book (Golyandina and Zhigljavsky 2013) developed further the methodology of 1D-SSA. It reflected the authors' new understandings as well as new SSA insights including subspace-based methods, filtering and rotations in the signal space for improving separability. A substantial paper (Golyandina et al. 2015) supplements the above books by expanding SSA for processing multivariate time series and digital images.

1.7.2 Some Recent Applications of SSA

The number of publications devoted to applications of SSA is steadily increasing. In addition to the standard applications areas such as climatology, meteorology, and geophysics, there are now many papers devoted to applications in engineering, economics, finance, biomedicine, and other areas. One can find many references to recent publications in Zhigljavsky (2010) and many papers in the two special issues of *Statistics and Its Interface* (2010, v.3, No.3 and 2017, v.10, No.1), which are either fully or partly devoted to SSA. In this short section we briefly mention some recent applications of SSA. In most of these papers, only the simplest versions of SSA (that is, Basic SSA of Sect. 2.1 and Toeplitz SSA of Sect. 2.2) have been used.

Advantages of 2D-SSA (described in Sect. 5.1) over some other methods of image processing are demonstrated in Zabalza et al. (2014, 2015) in application to hyperspectral imaging. Application of 2D-SSA to gap-filling is considered in von Buttlar et al. (2014). Application of Multivariate SSA for detecting oscillator clusters in multivariate datasets is proposed in Groth and Ghil (2015).

It is not easy to find applied areas related to analysis of temporal data, where 1D-SSA was not applied. Let us give some examples. In Salgado et al. (2013) and several other papers of the same authors, SSA has been used as the main technique in the development of a tool-wear monitoring system. Security of mobile devices is considered in Genkin et al. (2016), where SSA is used for preprocessing. In Sella et al. (2016), SSA was used for extraction of economic cycles. Filho and Lima (2016) use SSA for gap filling of precipitation data. Some recent applications in climatology were considered in Mudersbach et al. (2013), Monselesan et al. (2015) and in Pepelyshev and Zhigljavsky (2017). In Karnjana et al. (2017), SSA helps to solve the problem of unauthorized modification in speech signals. In Barrios-Muriel et al. (2016), SSA is used for de-noising in the problem of removal of electrocardiogram interference in electromyography signals. The paper (Hudson and Keatley 2017) is related to the decomposition and reconstruction of long-term flowering records of eight eucalypt species. In Wang et al. (2017), SSA was used as a preprocessing tool prior to making a classification of a medical data; the authors wrote: "the results have demonstrated the robustness of the approach when testing on large scale datasets with clinically acceptable sensitivity and specificity."

1.7.3 SSA for Preprocessing/Combination of Methods

For many different methods, SSA provides improvement if it is used as a preprocessing tool. There are dozens of papers, where hybrid methods incorporating SSA are considered. In most of the applications, SSA serves for either denoising or feature extraction. Let us give some examples of papers considering hybrids of SSA and other methods.

SSA is used as a preprocessing step for ARIMA in Zhang et al. (2011). A cooperative hybrid of SSA, ARIMA, and Holt-Winters is suggested in Xin et al. (2015). In Lakshmi et al. (2016) it is shown that the hybrid SSA + ARMAX is better than ARMAX alone for detection of structural damages for problems of Structural Health Monitoring.

In machine leaning, SSA is frequently used to obtain new characteristics of time series for a subsequent use of them in other models and methods. This is called feature extraction. The paper (Sivapragasam et al. 2001) is considered as a one of the first papers, where SSA is used together with Support Vector Machines (SVM). A hybrid of SSA with Neural Networks was suggested in Lisi et al. (1995).

In Wang et al. (2016), support vector machine regression (SVR) is applied separately to the trend and fluctuations, which are extracted by SSA. The constructed method is applied to forecast a time series data of failures gathered at the maintenance stage of the Boeing 737 aircraft. It is shown that the suggested hybrid SSA+SVR outperforms Holt-Winters, autoregressive integrated moving average, multiple linear regression, group method of data handling, SSA, and SVR used separately. Similar techniques are considered in Xiao et al. (2014), where SSA is employed for extraction of the trend and seasonality and then Neural Networks and fuzzy logic are applied to them separately with consequent combination. In Wu and Chau (2011), SSA is successfully used for noise removal before Neural Networks are applied. This work contains a review of different approaches to Rainfall-runoff modeling by means of SSA used in combination with other methods.

In Zabalza et al. (2014), SSA has been applied in hyperspectral imaging for effective feature extraction (noise removal), and then SVM was used for classification. It appeared that SSA performed preprocessing better than Empirical Mode Decomposition (EMD). Note that SSA and EMD do not only compete; they can be successful as hybrids. For example, in Yang et al. (2012) EMD is used for trend extraction and then SSA is applied to forecast changes in the trend.

1.7.4 Materials Used in This Book

In writing this book we have used much material from different sources. Many sections contain the material which is entirely new but other sections are based on our previous publications. Let us briefly describe the main references we have used in writing the theoretical and methodological material of the book.

Chapter 1 (Introduction: Overview) contains an original approach to the SSA modifications from a general viewpoint. The generic scheme of SSA-family methods from Sect. 1.1 was suggested in Golyandina et al. (2015).

1D-SSA is well elaborated and therefore Chap. 2 (SSA analysis of one-dimensional time series), in addition to some new material (this especially concerns new examples and the discussions concerning RSSA), revises standard SSA techniques. Sections 2.1 (Basic SSA) and Sect. 2.2 (Toeplitz SSA) contain standard material partially taken from Golyandina et al. (2001; Chapter 1). Ideas of Sect. 2.3 (SSA with projection) were firstly suggested in Golyandina et al. (2001; Section 1.7.1) (centering) and then extended in Golyandina and Shlemov (2017). Methods described in Sect. 2.4 (Iterative Oblique SSA) and Sect. 2.5 (Filter-adjusted O-SSA and SSA with derivatives) were suggested in Golyandina and Shlemov (2015); these two sections closely follow this paper. Section 2.6 (Shaped 1D-SSA) contains a particular case of Shaped SSA, which was suggested in Golyandina et al. (2015) for multidimensional case and is described in Sect. 5.2. Section 2.7 (Automatic grouping in SSA) follows Alexandrov (2009) and Golyandina and Zhigljavsky (2013; Section 1.4.5).

Much of the theoretical material of Chap. 3 (Parameter estimation, forecasting, gap filling) is standard for the methodologies of the subspace-based methods. In writing Sect. 3.1 (Parameter estimation) we have extensively used Golyandina and Zhigljavsky (2013; Sections 2.2, 2.8). Section 3.2 (Forecasting) includes the algorithms from Golyandina et al. (2001; Chapter 2). Gap filling in Sect. 3.3 contains two methods, iterative method taken from Kondrashov and Ghil (2006) and the subspace-based method taken from Golyandina and Osipov (2007). Both methods are described in accordance with Golyandina and Zhigljavsky (2013). Section 3.4 devoted to structured low-rank approximation (briefly, SLRA) describes

Cadzow-like (Cadzow 1988) iterative algorithms for finding low-rank approximations. SLRA is a very standard approach, which was extended to weighted Cadzow iterations in Zhigljavsky et al. (2016a) and Zvonarev and Golyandina (2017).

In writing Chap. 4 (SSA for multivariate time series) we use Golyandina and Stepanov (2005) and Golyandina et al. (2015). In Chap. 5 (Image processing), we mainly follow Golyandina et al. (2015). Moreover, in Sect. 5.1 (2D-SSA) we use material from Golyandina and Usevich (2010) and in Sect. 5.2 (Shaped 2D-SSA) and Sect. 5.3 (2D ESPRIT) we incorporate the ideas developed in Golyandina et al. (2015) and Shlemov and Golyandina (2014).

Some material in the algorithmic and RSSA sections is based on the papers (Golyandina and Korobeynikov 2013; Golyandina et al. 2015).

1.8 Installation of RSSA and Description of the Data Used in the Book

1.8.1 Installation of RSSA *and Usage Comments*

The package RSSA is available from CRAN on http://CRAN.R-project.org/package=Rssa and can be installed via the standard `install.packages` routine and therefore all the dependencies are installed automatically.

There is a special library, FFTW (Frigo and Johnson 2005), which is not mandatory for the installation of RSSA; however, possibilities of RSSA would be considerably lower if FFTW is not installed. The library FFTW should be installed prior to RSSA by the standard tools of the used operating system. For example, FFTW can be installed by running `apt-get install libfftw3-bin libfftw3-dev` (Ubuntu Linux) or `brew install fftw` (MacOS, homebrew). Windows pre-built packages from CRAN already use FFTW.

Sources of RSSA can be found at https://github.com/asl/rssa, where the user can ask questions about installation and usage problems. In addition, the current development version of RSSA could be installed straight from github repository using `install_github` from DEVTOOLS (Wickham and Chang 2017).

In this book, RSSA v1.0 has been used for all illustrative examples. All SSA computations can be reproduced by the reader and should run correctly by any later version of RSSA. The sets of data used in these examples are included into the R-package SSABOOK, unless a particular set is contained in one of the following three R-packages: built-in DATASETS, FMA, and RSSA. The description of the datasets is contained in Tables 1.2 and 1.3. In order to reproduce the examples from the book, SSABOOK should be installed as well as LATTICE, LATTICEEXTRA, PLYR, and FMA. Source codes for all examples as well as the R-package SSABOOK can be downloaded from the web-site devoted to the book https://ssa-with-r-book.github.io.

In the following chapters there are quite a few sections named “Description of functions.” In these sections, we describe the main RSSA functions and their basic

Table 1.2 Description of data and R-packages

Data	Short description	Timing	R package	Dataframe name
AustralianWine	Australian wine sales: thousands of liters. By wine makers in bottles ≤ 1 l	Monthly	RSSA	AustralianWine
CO2	Atmospheric concentrations of CO2 in parts per million (ppm)	Monthly	DATASETS	co2
White dwarf	Time variation of the intensity of the variable white dwarf star PG1159-035	Each 10 s	SSABOOK	dwarfst
Production	Crude oil and natural gas plant liquids production	Monthly	SSABOOK	oilproduction
Tree rings	Tree ring indices, Douglas fir, Snake river basin, USA	Annual	SSABOOK	dftreerings
MotorVehicle	Total domestic and foreign car sales in the USA in thousands	Monthly	RSSA	MotorVehicle
US unemployment	U.S. male and female unemployment figures in thousands (16–19 years and from 20 years)	Monthly	RSSA	USUnemployment
Hotel	Hotel occupied room av.	Monthly	SSABOOK	hotel
PayNSA	All employees: Total nonfarm payrolls	Monthly	SSABOOK	paynsa
Elec	Australian electricity production	Monthly	FMA	elec
Cowtemp	Daily morning temperature of a cow, measured at 6.30am	Daily	FMA	cowtemp
Glonass	Glonass time corrections	Each 5 min	SSABOOK	g15
Sunspots	Mean total sunspot number	Annual	SSABOOK	sunspot2
Bookings	Numbers of hotel bookings through a particular web-site	Hourly	SSABOOK	bookings
EuStockMarkets	Closing prices of major European stock indices	Daily	DATASETS	EuStockMarkets
Mars	An image of Mars performed by Pierre Thierry	Image	RSSA	Mars
Brecon Beacons	The test DTM of a region in South Wales, UK	Image	SSABOOK	brecon
Kruppel	Regularized data of gene expression for the “Krüppel” gene (a drosophila embryo)	image	SSABOOK	kruppel
Monet	“A preview of the painting of Claude Monet called “Study of a Figure Outdoors: Woman with a Parasol, facing left,” 1886	Image	SSABOOK	monet

Table 1.3 Description of data and sources

Data	Length	Time range	Source
AustralianWine	176	Jan 1980–Jul 1995	Hyndman (2013)
CO2	468	Jan 1959–Jan 1997	Cleveland (1993)
White dwarf	618	during March 1989	Weigend and Gershenfeld (1993)
Production	300	Jan 1973–Dec 1997	Source: U.S. Energy Information Administration (Jan 2016) http://www.eia.gov/totalenergy/data/monthly/#summary
Tree rings	669	1282–1950	Hyndman (2013)
MotorVehicle	541	Jan 1967–Jan 2012	U.S. Bureau of Economic Analysis, 2015. Table 7.2.5s https://www.bea.gov/histdata/Releases/GDP_and_PI/2015/Q1/Third_June-24-2015/UND/Section7ALL_xls.xls
US unemployment	408	Jan 1948–Jan 1981	Andrews and Herzberg (1985)
Hotel	168	Jan 1963–Dec 1976	Hyndman (2013)
PayNSA	913	Jan 1939–Jan 2015	Hyndman (2013)
Elec	476	Jan 1956–Aug 1995	Makridakis et al. (1998)
Cowtemp	75	Unknown	Makridakis et al. (1998)
Glonass	104832	02/01/2014 to 31/12/2014	https://www.glonass-iac.ru/en/index.php
Sunspots	316	1700–2015	Hyndman (2013)
Bookings	4344	23/09/2016 to 22/03/2017	Provided by Crimtan, UK, to the authors of the R-package SSABOOK
EuStockMarkets	1860	During 1994–1998	Provided by Erste Bank AG, Vienna, Austria, to the authors of the R-package DATASETS
Mars	258 × 275	NA	http://www.astrosurf.com/buil/iris/tutorial8/doc23_us.htm
Brecon Beacons	80 × 100	NA	The data are obtained by means of the function getData of the R-package RASTER
Kruppel	200 × 200	NA	http://bdtnp.lbl.gov/Fly-Net/
Monet	400 × 263 × 3	NA	https://commons.wikimedia.org/wiki/File:Monet.012.sonnenschirm.jpg

arguments. For more information about the RSSA functions and their arguments, we refer the reader to the help information in the RSSA package.

1.8.2 Data Description

Tables 1.2 and 1.3 present the data used in the book for examples. All these sets of data can be found in the R-packages indicated in the fourth column of Table 1.2. Table 1.3 contains one of possible references; detailed descriptions and references can be found in the corresponding R-packages.

References

Alexandrov T (2009) A method of trend extraction using singular spectrum analysis. RevStat 7(1):1–22

Allen M, Smith L (1996) Monte Carlo SSA: Detecting irregular oscillations in the presence of colored noise. J Clim 9(12):3373–3404

Andrews D, Herzberg A (1985) Data. A collection of problems from many fields for the student and research worker. Springer, New York

Barrios-Muriel J, Romero F, Alonso FJ, Gianikellis K (2016) A simple SSA-based de-noising technique to remove ECG interference in EMG signals. Biomed Signal Process Control 30:117–126

Bozzo E, Carniel R, Fasino D (2010) Relationship between singular spectrum analysis and Fourier analysis: Theory and application to the monitoring of volcanic activity. Comput Math Appl 60(3):812–820

Broomhead D, King G (1986) 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: Sarkar S (ed) Nonlinear phenomena and chaos. Adam Hilger, Bristol, pp 113–144

von Buttlar J, Zscheischler J, Mahecha MD (2014) An extended approach for spatiotemporal gapfilling: dealing with large and systematic gaps in geoscientific datasets. Nonlinear Process Geoph 21:203–215

Cadzow JA (1988) Signal enhancement: a composite property mapping algorithm. IEEE Trans Acoust 36(1):49–62

Chen Q, Weigelt M, Sneeuw N, van Dam T (2016) On time-variable seasonal signals: Comparison of SSA and Kalman filtering based approach. Springer International Publishing, Cham, pp 75–80

Cleveland WS (1993) Visualizing data. Hobart Press

Danilov D, Zhigljavsky A (eds) (1997) Principal components of time series: the “Caterpillar” method. St. Petersburg Press (in Russian)

Filho ASF, Lima GAR (2016) Gap filling of precipitation data by SSA - singular spectrum analysis. J Phys Conf Ser 759(1):012,085

Fraedrich K (1986) Estimating dimensions of weather and climate attractors. J Atmos Sci 43: 419–432

Frigo M, Johnson SG (2005) The design and implementation of FFTW3. Proc IEEE 93(2):216–231

Genkin D, Pachmanov L, Pipman I, Tromer E, Yarom Y (2016) Ecdsa key extraction from mobile devices via nonintrusive physical side channels. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security. ACM, New York, NY, USA, CCS ’16, pp 1626–1638

Ghil M, Allen RM, Dettinger MD, Ide K, Kondrashov D, Mann ME, Robertson A, Saunders A, Tian Y, Varadi F, Yiou P (2002) Advanced spectral methods for climatic time series. Rev Geophys 40(1):1–41

Golub GH, Van Loan CF (1996) Matrix computations, 3rd edn. Johns Hopkins University Press, Baltimore, MD, USA

Golyandina N, Korobeynikov A (2013) Basic singular spectrum analysis and forecasting with R. Comput Stat Data Anal 71:943–954

Golyandina N, Osipov E (2007) The "Caterpillar"-SSA method for analysis of time series with missing values. J Stat Plan Inference 137(8):2642–2653

Golyandina N, Shlemov A (2015) Variations of singular spectrum analysis for separability improvement: Non-orthogonal decompositions of time series. Stat Interface 8(3):277–294

Golyandina N, Shlemov A (2017) Semi-nonparametric singular spectrum analysis with projection. Stat Interface 10(1):47–57

Golyandina N, Stepanov D (2005) SSA-based approaches to analysis and forecast of multidimensional time series. In: Proceedings of the 5th St.Petersburg workshop on simulation, June 26–July 2, 2005. St. Petersburg State University, St. Petersburg, pp 293–298

Golyandina N, Usevich K (2010) 2D-extension of singular spectrum analysis: algorithm and elements of theory. In: Olshevsky V, Tyrtyshnikov E (eds) Matrix methods: Theory, algorithms and applications. World Scientific Publishing, pp 449–473

Golyandina N, Zhigljavsky A (2013) Singular apectrum analysis for time series. Springer briefs in statistics. Springer

Golyandina N, Nekrutkin V, Zhigljavsky A (2001) Analysis of time series structure: SSA and related techniques. Chapman&Hall/CRC

Golyandina N, Korobeynikov A, Shlemov A, Usevich K (2015) Multivariate and 2D extensions of singular spectrum analysis with the Rssa package. J Stat Softw 67(2):1–78

Groth A, Ghil M (2015) Monte Carlo singular spectrum analysis (SSA) revisited: Detecting oscillator clusters in multivariate datasets. J Climate 28(19):7873–7893

Hall MJ (1998) Combinatorial theory. Wiley, New York

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. Int J Forecast 25(1):103–118

Hassani H, Heravi S, Zhigljavsky A (2013) Forecasting UK industrial production with multivariate singular spectrum analysis. J Forecast 32(5):395–408

Hassani H, Webster A, Silva ES, Heravi S (2015) Forecasting U.S. tourist arrivals using optimal singular spectrum analysis. Tourism Manag 46:322–335

Hudson IL, Keatley MR (2017) Singular spectrum analytic (ssa) decomposition and reconstruction of flowering: Signatures of climatic impacts. Environ Model Assess 22(1):37–52

Hyndman R, Athanasopoulos G (2013) Forecasting: principles and practice. OTexts, URL http://otexts.org/fpp/, accessed on 28.07.2017

Hyndman RJ (2013) Time Series Data Library. URL http://data.is/TSDLdemo, accessed on 10/08/2013

Iqelan BM (2017) A singular spectrum analysis technique to electricity consumption forecasting. Int J Eng Res Appl 7(3):92–100

Karnjana J, Unoki M, Aimmanee P, Wutiwiwatchai C (2017) Tampering detection in speech signals by semi-fragile watermarking based on singular-spectrum analysis. Springer International Publishing, Cham, pp 131–140

Kondrashov D, Ghil M (2006) Spatio-temporal filling of missing points in geophysical data sets. Nonlinear Process Geophys 13(2):151–159

Korobeynikov A (2010) Computation- and space-efficient implementation of SSA. Stat Interface 3(3):357–368

Korobeynikov A, Larsen RM, Wu KJ, Yamazaki I (2016) SVD: Interfaces to various state-of-art SVD and eigensolvers. URL http://CRAN.R-project.org/package=svd, R package version 0.4

Korobeynikov A, Shlemov A, Usevich K, Golyandina N (2017) Rssa: A collection of methods for singular spectrum analysis. URL http://CRAN.R-project.org/package=Rssa, R package version 1.0

Kumar KS, Rajesh R, Tiwari RK (2017) Regional and residual gravity anomaly separation using the singular spectrum analysis-based low pass filtering: a case study from Nagpur, Maharashtra, India. Explor Geophys.

Kurakin V, Kuzmin A, Mikhalev A, Nechaev A (1995) Linear recurring sequences over rings and modules. J Math Sci 76(6):2793–2915

Lakshmi K, Rao ARM, Gopalakrishnan N (2016) Singular spectrum analysis combined with ARMAX model for structural damage detection. Struct Control Health Monit. https://doi.org/10.1002/stc.1960

Lisi F, Nicolis O, Sandri M (1995) Combining singular-spectrum analysis and neural networks for time series forecasting. Neural Process Lett 2(4):6–10

Makridakis S, Wheelwright S, Hyndman R (1998) Forecasting: Methods and applications, 3rd edn. Wiley, New York

Monselesan DP, O'Kane TJ, Risbey JS, Church J (2015) Internal climate memory in observations and models. Geophys Res Lett 42(4):1232–1242

Mudersbach C, Wahl T, Haigh ID, Jensen J (2013) Trends in high sea levels of German North Sea gauges compared to regional mean sea level changes. Cont Shelf Res 65:111–120

Pepelyshev A, Zhigljavsky A (2017) Ssa analysis and forecasting of records for earth temperature and ice extents. Stat Interface 10(1):151–163

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'alkool à différentes températures. J de l'Ecole Polytechnique 1(2):24–76

Salgado D, Cambero I, Olivenza JH, Sanz-Calcedo JG, López PN, Plaza EG (2013) Tool wear estimation for different workpiece materials using the same monitoring system. Procedia Eng 63:608–615

Sella L, Vivaldo G, Groth A, Ghil M (2016) Economic cycles and their synchronization: A comparison of cyclic modes in three European countries. J Bus Cycle Res 12(1):25–48

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

Sivapragasam C, Liong SY, Pasha M (2001) Rainfall and runoff forecasting with SSA–SVM approach. J Hydroinform 3(3):141–152

Vautard M, 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

Vile JW J L nnd Gillard, Harper PR, Knight VA (2012) Predicting ambulance demand using singular spectrum analysis. J Oper Res Soc 63(11):1556–1565

Wang S, Tang HL, turk LIA, Hu Y, Sanei S, Saleh GM, Peto T (2017) Localizing microaneurysms in fundus images through singular spectrum analysis. IEEE Trans Biomed Eng 64(5):990–1002

Wang X, Wu J, Liu C, Wang S, Niu W (2016) A hybrid model based on singular spectrum analysis and support vector machines regression for failure time series prediction. Qual Reliab Eng Int 32(8):2717–2738, qRE-16-0186.R2

Watson PJ (2016) Identifying the best performing time series analytics for sea level research. Springer International Publishing, Cham, pp 261–278

Weigend A, Gershenfeld N (eds) (1993) Time series prediction: Forecasting the future and understanding the past. Addison-Wesley, Reading

Wickham H, Chang W (2017) DEVTOOLS: Tools to Make Developing R Packages Easier. URL http://CRAN.R-project.org/package=devtools, R package version 1.13.2

Wu C, Chau K (2011) Rainfall–runoff modeling using artificial neural network coupled with singular spectrum analysis. J Hydrol 399(3):394–409

Xiao Y, Liu JJ, Hu Y, Wang Y, Lai KK, Wang S (2014) A neuro-fuzzy combination model based on singular spectrum analysis for air transport demand forecasting. J Air Transp Manag 39:1–11

Xin W, Chao L, Weiren X, Ying L (2015) A failure time series prediction method based on UML model. In: 2015 4th international conference on computer science and network technology (ICCSNT), vol 01, pp 62–70

Yang Z, Bingham C, Ling WK, Zhang Y, Gallimore M, Stewart J (2012) Unit operational pattern analysis and forecasting using EMD and SSA for industrial systems. Springer, Berlin, Heidelberg, pp 416–423

Zabalza J, Ren J, Wang Z, Marshall S, Wang J (2014) Singular spectrum analysis for effective feature extraction in hyperspectral imaging. IEEE Geosci Remote Sens Lett 11(11):1886–1890

Zabalza J, Ren J, Zheng J, Han J, Zhao H, Li S, Marshall S (2015) Novel two-dimensional singular spectrum analysis for effective feature extraction and data classification in hyperspectral imaging. IEEE Trans Geosci Remote Sens 53(8):4418–4433

Zhang Q, Wang BD, He B, Peng Y, Ren ML (2011) Singular spectrum analysis and ARIMA hybrid model for annual runoff forecasting. Water Resour Manag 25(11):2683–2703

Zhigljavsky A (2010) Singular spectrum analysis for time series: Introduction to this special issue. Stat Interface 3(3):255–258

Zhigljavsky A, Golyandina N, Gillard J (2016a) Analysis and design in the problem of vector deconvolution. In: Kunert J, Müller HC, Atkinson CA (eds) mODa 11 - Advances in model-oriented design and analysis. Springer International Publishing, pp 243–251

Zvonarev N, Golyandina N (2017) Iterative algorithms for weighted and unweighted finite-rank time-series approximations. Stat Interface 10(1):5–18