Chapter 3
SSA for Forecasting, Interpolation, Filtering and Estimation

3.1 SSA Forecasting Algorithms

3.1.1 Main Ideas and Notation

A reasonable forecast of a time series can be performed only if the series has a structure and there are tools to identify and use this structure. Also, we should assume that the structure of the time series is preserved for the future time period over which we are going to forecast (continue) the series. The last assumption cannot be validated using the data to be forecasted. Moreover, the structure of the series can rarely be identified uniquely. Therefore, the situation of different and even contradictory forecasts is not impossible. Hence it is important not only to understand and express the structure but also to assess its stability.

A forecast can be made only if a model is built. The model should be either derived from the data or at least checked against the data. In SSA forecasting, these models can be described through the linear recurrence relations (LRRs). The class of series governed by LRRs is rather wide and important for practical applications. This class contains the series that are linear combinations of products of exponential, polynomial and harmonic series.

Assume that $\mathbb{X}_N = \mathbb{X}_N^{(1)} + \mathbb{X}_N^{(2)}$, where the series $\mathbb{X}_N^{(1)}$ satisfies an LRR of relatively small order and we are interested in forecasting of $\mathbb{X}_N^{(1)}$. For example, $\mathbb{X}_N^{(1)}$ can be signal, trend or seasonality. The idea of recurrent forecasting is to estimate the underlying LRR and then to perform forecasting by applying the estimated LRR to the last points of the SSA approximation of the series $\mathbb{X}_N^{(1)}$. The main assumption allowing SSA forecasting is that for a certain window length L the series components $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ are approximately strongly separable. In this case, we can reconstruct the series $\mathbb{X}_N^{(1)}$ with the help of a selected set of the eigentriples and obtain approximations to both the series $\mathbb{X}_N^{(1)}$, its trajectory space and the true LRR.

N. Golyandina and A. Zhigljavsky, *Singular Spectrum Analysis for Time Series*, SpringerBriefs in Statistics,
https://doi.org/10.1007/978-3-662-62436-4_3

Let $\mathbb{X}_N = \mathbb{X}_N^{(1)} + \mathbb{X}_N^{(2)}$ and we intend to forecast $\mathbb{X}_N^{(1)}$. If $\mathbb{X}_N^{(1)}$ is a time series of finite rank $r < L$, then it generates an L-trajectory subspace of dimension r. This subspace reflects the structure of $\mathbb{X}_N^{(1)}$ and hence it can be taken as a base for forecasting.

Let us formally describe the forecasting algorithms in a chosen subspace. As we assume that estimates of this subspace are constructed by SSA, we shall refer to the algorithms as the algorithms of SSA forecasting.

Forecasting within a subspace means a continuation of the L-lagged vectors of the time series in such a way that they lie in or very close to the chosen subspace of R^L. We consider the following three forecasting algorithms: recurrent, vector and simultaneous.

Inputs in the forecasting algorithms:

(a) Time series $\mathbb{X}_N = (x_1, \ldots, x_N)$, $N > 2$.

(b) Window length L, $1 < L < N$.

(c) Linear space $\mathcal{L}_r \subset \mathsf{R}^L$ of dimension $r < L$. We assume that $\mathbf{e}_L \notin \mathcal{L}_r$, where $\mathbf{e}_L = (0, 0, \ldots, 0, 1)^{\mathrm{T}} \in \mathsf{R}^L$; in other terms, $\mathcal{L}_r$ is not a 'vertical' space.

(d) Number M of points to forecast for.

Notation:

(a) $\mathbf{X} = [X_1 : \ldots : X_K]$ (with $K = N - L + 1$) is the trajectory matrix of $\mathbb{X}_N$.

(b) $P_1, \ldots, P_r$ is an orthonormal basis in $\mathcal{L}_r$.

(c) $\widehat{\mathbf{X}} \overset{\text{def}}{=} [\widehat{X}_1 : \ldots : \widehat{X}_K] = \sum\limits_{i=1}^{r} P_i P_i^{\mathrm{T}} \mathbf{X}$. The vector $\widehat{X}_i$ is the orthogonal projection of X_i onto the space $\mathcal{L}_r$.

(d) $\widetilde{\mathbf{X}} = \boldsymbol{\Pi}_{\mathcal{H}} \widehat{\mathbf{X}} = [\widetilde{X}_1 : \ldots : \widetilde{X}_K]$ is the result of hankelization of the matrix $\widehat{\mathbf{X}}$. The matrix $\widetilde{\mathbf{X}}$ is the trajectory matrix of some time series $\widetilde{\mathbb{X}}_N = (\widetilde{x}_1, \ldots, \widetilde{x}_N)$.

(e) For any vector $Y \in \mathsf{R}^L$, we denote by $\overline{Y} \in \mathsf{R}^{L-1}$ the vector consisting of the last $L - 1$ components of the vector Y and by $\underline{Y} \in \mathsf{R}^{L-1}$ the vector consisting of the first $L - 1$ components of Y.

(f) We set $\nu^2 = \pi_1^2 + \ldots + \pi_r^2$, where π_i is the last component of the vector P_i ($i = 1, \ldots, r$). As ν^2 is the squared cosine of the angle between the vector $\mathbf{e}_L$ and the linear space $\mathcal{L}_r$, it is called the *verticality coefficient* of $\mathcal{L}_r$. Since $\mathbf{e}_L \notin \mathcal{L}_r$, $\nu^2 < 1$.

The following statement is fundamental.

Proposition 3.1 *In the notation above, the last component y_L of any vector $Y = (y_1, \ldots, y_L)^{\mathrm{T}} \in \mathcal{L}_r$ is a linear combination of the first components $y_1, \ldots, y_{L-1}$:*

$$y_L = a_1 y_{L-1} + a_2 y_{L-2} + \ldots + a_{L-1} y_1,$$

where the vector $R = (a_{L-1}, \ldots, a_1)^{\mathrm{T}}$ can be expressed as

$$R = \frac{1}{1 - \nu^2} \sum_{i=1}^{r} \pi_i \underline{P_i} \tag{3.1}$$

and does not depend on the choice of the basis $P_1, \ldots, P_r$ in the linear space $\mathcal{L}_r$.

Proof follows from the fact that the formula (3.1) is a particular case of (3.10) below with $n = L, m = r$ and $\mathfrak{Q} = \{L\}$. Another proof of Proposition 3.1 is contained in the proof of [21, Theorem 5.2].

3.1.2 Formal Description of the Algorithms

Below we write down the algorithms of SSA forecasting; see [24, Sect. 3.2.2] and [25] containing the R codes for the calls of the corresponding functions of the Rssa package.

3.1.2.1 Recurrent Forecasting

In the above notation, the *recurrent forecasting algorithm* (briefly, *R-forecasting*) can be formulated as follows.

Algorithm of R-forecasting.

1. The time series $\mathbb{Y}_{N+M} = (y_1, \ldots, y_{N+M})$ is defined by

$$y_i = \begin{cases} \widetilde{x}_i & \text{for } i = 1, \ldots, N, \\ \sum\limits_{j=1}^{L-1} a_j y_{i-j} & \text{for } i = N+1, \ldots, N+M. \end{cases} \tag{3.2}$$

2. The numbers $y_{N+1}, \ldots, y_{N+M}$ form the M terms of the recurrent forecast.

This yields that R-forecasting is performed by the direct use of the LRR with coefficients $\{a_j, j = 1, \ldots, L-1\}$ derived in Proposition 3.1.

Remark 3.1 Let us define the linear operator $\mathcal{P}_{\text{Rec}} : \mathsf{R}^L \mapsto \mathsf{R}^L$ by the formula

$$\mathcal{P}_{\text{Rec}} Y = \begin{pmatrix} \overline{Y} \\ R^{\mathrm{T}} \overline{Y} \end{pmatrix}. \tag{3.3}$$

Set

$$Z_i = \begin{cases} \widetilde{X}_i & \text{for } i = 1, \ldots, K, \\ \mathcal{P}_{\text{Rec}} Z_{i-1} & \text{for } i = K+1, \ldots, K+M. \end{cases} \tag{3.4}$$

It is easily seen that the matrix $\mathbf{Z} = [Z_1 : \ldots : Z_{K+M}]$ is the trajectory matrix of the series $\mathbb{Y}_{N+M}$. Therefore, (3.4) can be regarded as the vector form of (3.2).

3.1.2.2 Vector Forecasting

The idea of *vector forecasting* (briefly, *V-forecasting*) is as follows. Let us assume that we can continue the sequence of vectors $\widehat{X}_1, \ldots, \widehat{X}_K$ (which belong to the subspace $\mathcal{L}_r$) for M steps so that:

(a) the continuation vectors Z_m $(K < m \leq K + M)$ belong to the same subspace $\mathcal{L}_r$;
(b) the matrix $\mathbf{X}_M = [\widehat{X}_1 : \ldots : \widehat{X}_K : Z_{K+1} : \ldots : Z_{K+M}]$ is approximately Hankel.

Then, after obtaining the matrix $\mathbf{X}_M$, we can obtain the forecasted series $\mathbb{Y}_{N+M}$ by means of the diagonal averaging of this matrix.

In addition to the notation introduced above let us bring in some more notation. Consider the matrix

$$\boldsymbol{\Pi} = \underline{\mathbf{V}}\,\underline{\mathbf{V}}^{\mathrm{T}} + (1 - \nu^2) R R^{\mathrm{T}}, \tag{3.5}$$

where $\underline{\mathbf{V}} = [\underline{P_1} : \ldots : \underline{P_r}]$. The matrix $\boldsymbol{\Pi}$ is the matrix of the linear operator that performs the orthogonal projection $\mathsf{R}^{L-1} \mapsto \underline{\mathcal{L}_r}$, where $\underline{\mathcal{L}_r} = \mathrm{span}(\underline{P_1}, \ldots, \underline{P_r})$; note that this matrix $\boldsymbol{\Pi}$ is a particular case of the matrix defined in (3.11) with $m = r$, $n = L$ and $\mathcal{Q} = \{L\}$. Finally, we define the linear operator $\mathcal{P}_{\mathrm{Vec}} : \mathsf{R}^L \mapsto \mathcal{L}_r$ by the formula

$$\mathcal{P}_{\mathrm{Vec}} Y = \begin{pmatrix} \boldsymbol{\Pi}\overline{Y} \\ R^{\mathrm{T}}\overline{Y} \end{pmatrix}. \tag{3.6}$$

Algorithm of V-forecasting.

1. In the notation above, define the vectors Z_i as follows:

$$Z_i = \begin{cases} \widehat{X}_i & \text{for } \ i = 1, \ldots, K, \\ \mathcal{P}_{\mathrm{Vec}} Z_{i-1} & \text{for } \ i = K+1, \ldots, K+M+L-1. \end{cases} \tag{3.7}$$

2. By constructing the matrix $\mathbf{Z} = [Z_1 : \ldots : Z_{K+M+L-1}]$ and making its diagonal averaging we obtain the series $y_1, \ldots, y_{N+M+L-1}$.
3. The numbers $y_{N+1}, \ldots, y_{N+M}$ form the M terms of the vector forecast.

Remark 3.2 Note that in order to get M forecast terms, the vector forecasting procedure performs $M+L-1$ steps. The aim is the permanence of the forecast under variations in M: the M-step forecast ought to coincide with the first M values of the forecast for $M+1$ or more steps. In view of the definition of the diagonal averaging, to achieve this we have to make $L-1$ extra steps.

3.1.2.3 Simultaneous Forecasting

R-forecasting is based on the fact that the last coordinate of any vector in the subspace $\mathcal{L}_r$ is determined by its first $L - 1$ coordinates. The idea of the *simultaneous forecast-*

ing algorithm is based on the following relation: under some additional conditions, the last M coordinates of any vector in $\mathfrak{L}_r$ can be expressed through its first $L - M$ coordinates. Certainly, $L - M$ should be larger than r and therefore $M < L - r$.

Let $\mathrm{span}(\mathbf{e}_i, i = L - M + 1, \ldots, L) \cap \mathfrak{L}_r = \{\mathbf{0}\}$. For a vector $Y \in \mathfrak{L}_r$, denote $Y_1 = (y_1, \ldots, y_{L-M})^{\mathrm{T}}$ and $Y_2 = (y_{L-M+1}, \ldots, y_L)^{\mathrm{T}}$. Then $Y_2 = \mathbf{R}Y_1$, where the matrix $\mathbf{R}$ is defined by (3.10) below with $n = L$, $m = r$ and $\mathcal{Q} = \{L - M + 1, \ldots, L\}$.

Algorithm of simultaneous forecasting.

1. In the notation above, define the time series $\mathbb{Y}_{N+M} = (y_1, \ldots, y_{N+M})$ by

$$\begin{aligned} &y_i = \widetilde{x}_i \quad \text{for} \quad i = 1, \ldots, N, \\ &(y_{N+1}, \ldots, y_{N+M})^{\mathrm{T}} = \mathbf{R}(y_{N-(L-M)+1}, \ldots, y_N)^{\mathrm{T}}. \end{aligned} \tag{3.8}$$

2. The numbers $y_{N+1}, \ldots, y_{N+M}$ form the M terms of the simultaneous forecast.

Remark 3.3 The algorithm formulated above is an analogue of the algorithm of R-forecasting, since $\mathbf{R}$ in (3.8) is applied to the reconstructed series. An analogue of V-forecasting can also be considered.

3.1.3 SSA Forecasting Algorithms: Similarities and Dissimilarities

If $\mathfrak{L}_r$ is spanned by certain eigenvectors obtained from the SVD of the trajectory matrix of the series $\mathbb{X}_N$, then the corresponding forecasting algorithm will be called *Basic SSA forecasting algorithm.*

Let us return to Basic SSA and assume that our aim is to extract an additive component $\mathbb{X}_N^{(1)}$ from a series $\mathbb{X}_N$. For an appropriate window length L, we obtain the SVD of the trajectory matrix of the series $\mathbb{X}_N$ and select the eigentriples $(\sqrt{\lambda_i}, U_i, V_i)$, $i \in I$, corresponding to $\mathbb{X}_N^{(1)}$. Then we obtain the resultant matrix

$$\mathbf{X}_I = \sum_{i \in I} \sqrt{\lambda_i} U_i V_i^{\mathrm{T}}$$

and, after the diagonal averaging, we obtain the reconstructed series $\widetilde{\mathbb{X}}_N^{(1)}$ that estimates $\mathbb{X}_N^{(1)}$.

The columns $\widehat{X}_1, \ldots, \widehat{X}_K$ of the resultant matrix $\mathbf{X}_I$ belong to the linear space $\mathfrak{L}_r = \mathrm{span}(U_i, i \in I)$. If $\mathbb{X}_N^{(1)}$ is strongly separable from $\mathbb{X}_N^{(2)} \stackrel{\mathrm{def}}{=} \mathbb{X}_N - \mathbb{X}_N^{(1)}$, then $\mathfrak{L}_r$ coincides with $\mathcal{X}^{(L,1)}$ (the trajectory space of the series $\mathbb{X}_N^{(1)}$) and $\mathbf{X}_I$ is a Hankel matrix (in this case, $\mathbf{X}_I$ is the trajectory matrix of the series $\mathbb{X}_N^{(1)}$). Then the recurrent, vector and simultaneous forecasts coincide and the resulting procedure could be called the *exact continuation* of $\mathbb{X}_N^{(1)}$. More precisely, in this situation the matrix $\boldsymbol{\Pi}$

is the identity matrix, and (3.6) coincides with (3.3). Furthermore, the matrix $\mathbf{Z}$ has Hankel structure and the diagonal averaging does not change the matrix elements.

If $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ are approximately strongly separable, then $\mathcal{L}_r$ is close to $\mathcal{X}^{(L,1)}$ and $\mathbf{X}_I$ is approximately a Hankel matrix.

If there is no exact separability, then different forecasting algorithms usually give different results. Let us describe the difference between them. Since the recurrent and vector forecasting algorithms are more conventional and have less limitations, we shall concentrate on the recurrent and vector forecasting algorithms only.

- In a typical situation, there is no time series such that the linear space $\mathcal{L}_r$ (for $r < L - 1$) is its trajectory space [21, Proposition 5.6], and therefore this space cannot be the trajectory space of the series to be forecasted. The R-forecasting method uses $\mathcal{L}_r$ to obtain the LRR of the forecasting series. The V-forecasting procedure tries to perform the L-continuation of the series in $\mathcal{L}_r$: any vector $Z_{i+1} = \mathcal{P}_{\mathrm{Vec}} Z_i$ belongs to $\mathcal{L}_r$, and $\underline{Z_{i+1}}$ is as close as possible to $\overline{Z_i}$. The last component of Z_{i+1} is obtained from $\underline{Z_{i+1}}$ by the same LRR as used in R-forecasting.
- Both forecasting methods have two general stages: the diagonal averaging and continuation. For R-forecasting, the diagonal averaging is used to obtain the reconstructed series, and continuation is performed by applying the LRR. In V-forecasting, these two stages are used in the reverse order; first, vector continuation in $\mathcal{L}_r$ is performed and then the diagonal averaging gives the forecast.
- If there is no exact separability it is hard to compare the recurrent and vector forecasting methods theoretically. Closeness of two forecasts obtained by two different algorithms can be used as an argument in favour of forecasting stability.
- R-forecasting is simpler to interpret in view of the link between LRRs and their characteristic polynomials, see Sect. 3.2. On the other hand, numerical study demonstrates that V-forecasting is typically more ‘conservative’ (or less ‘radical’) when R-forecasting may exhibit either rapid increase or decrease.
- Only very few years ago, V-forecasting was considered as having a larger computational cost than R-forecasting; however, due to the current efficient implementation in Rssa [23], V-forecasting has become even slightly faster than R-forecasting.

Remark 3.4 Forecasting algorithms described in Sect. 3.1 are based on the estimation of the trajectory subspace of the forecasted component. In addition to Basic SSA, there are other methods of estimation of the trajectory subspace. For example, if the subspace is estimated by Toeplitz SSA, we obtain Toeplitz SSA forecasting algorithms. We may wish to use SSA with centering for estimating the subspace; in this case, we arrive at corresponding modifications of SSA forecasting with centering, see [21, Sect. 2.3.3].

3.1.4 Appendix: Vectors in a Subspace

In this section, we formulate two technical results that provide the theoretical ground for both forecasting and filling in methods. For proofs and details, we refer to [16].

Consider the Euclidean space R^n. Define $J_n = \{1, \ldots, n\}$ and denote by $\mathfrak{Q} = \{i_1, \ldots, i_s\} \subset J_n$ an ordered set, $|\mathfrak{Q}| = s$. Let $\mathbf{I}_s$ denote the unit $s \times s$ matrix. We define a *restriction of a vector* $X = (x_1, \ldots, x_n)^{\mathrm{T}} \in \mathsf{R}^n$ onto a set of indices $\mathfrak{Q} = \{i_1, \ldots, i_s\}$ as the vector $X\big|_{\mathfrak{Q}} = (x_{i_1}, \ldots, x_{i_s})^{\mathrm{T}} \in \mathsf{R}^s$. *The restriction of a matrix* onto a set of indices is the matrix consisting of restrictions of its column vectors onto this set.

The restriction of a q-dimensional subspace $\mathfrak{L}_q$ onto a set of indices $\mathfrak{Q}$ is the space spanned by restrictions of all vectors of $\mathfrak{L}_q$ onto this set; the restricted space will be denoted by $\mathfrak{L}_q\big|_{\mathfrak{Q}}$. It is easy to prove that for any basis $\{H_i\}_{i=1}^q$ of the subspace $\mathfrak{L}_q$, the equality $\mathfrak{L}_q\big|_{\mathfrak{Q}} = \mathrm{span}\big(H_1\big|_{\mathfrak{Q}}, \ldots, H_q\big|_{\mathfrak{Q}}\big)$ holds.

3.1.4.1 Filling in Vector Coordinates in the Subspace

Consider an m-dimensional subspace $\mathfrak{L}_m \subset \mathsf{R}^n$ with $m < n$. Denote by $\{P_k\}_{k=1}^m$ an orthonormal basis in $\mathfrak{L}_m$ and define the matrix $\mathbf{P} = [P_1 : \ldots : P_m]$. Fix an ordered set of indices $\mathfrak{Q} = \{i_1, \ldots, i_s\}$ with $s = |\mathfrak{Q}| \le n - m$.

First, note that the following conditions are equivalent (it follows from [16, Lemma 2.1]): (1) for any $Y \in \mathfrak{L}_m\big|_{J_n \setminus \mathfrak{Q}}$ there exists an unique vector $X \in \mathfrak{L}_m$ such that $X\big|_{J_n \setminus \mathfrak{Q}} = Y$, (2) the matrix $\mathbf{I}_s - \mathbf{P}\big|_{\mathfrak{Q}}\big(\mathbf{P}\big|_{\mathfrak{Q}}\big)^{\mathrm{T}}$ is non-singular, and (3) $\mathrm{span}(\mathbf{e}_i, i \in \mathfrak{Q}) \cap \mathfrak{L}_m = \{\mathbf{0}_n\}$. Either of these conditions can be considered as a condition of unique filling in of the missing vector components with indices from $\mathfrak{Q}$.

Proposition 3.2 *Let the matrix* $\mathbf{I}_s - \mathbf{P}\big|_{\mathfrak{Q}}\big(\mathbf{P}\big|_{\mathfrak{Q}}\big)^{\mathrm{T}}$ *be non-singular. Then for any vector* $X \in \mathfrak{L}_m$ *we have*

$$X\big|_{\mathfrak{Q}} = \mathbf{R}\, X\big|_{J_n \setminus \mathfrak{Q}}, \tag{3.9}$$

where

$$\mathbf{R} = \Big(\mathbf{I}_s - \mathbf{P}\big|_{\mathfrak{Q}}\big(\mathbf{P}\big|_{\mathfrak{Q}}\big)^{\mathrm{T}}\Big)^{-1} \mathbf{P}\big|_{\mathfrak{Q}}\big(\mathbf{P}\big|_{J_n \setminus \mathfrak{Q}}\big)^{\mathrm{T}}. \tag{3.10}$$

3.1.4.2 Projection Operator

Let $Y \in \mathsf{R}^n$ and $Z = Y\big|_{J_n \setminus \mathfrak{Q}} \in \mathsf{R}^{n-s}$. Generally, $Z \notin \mathfrak{L}_m\big|_{J_n \setminus \mathfrak{Q}}$. However, for applying formula (3.9) to obtain the vector from $\mathfrak{L}_m$, it is necessary that $Z \in \mathfrak{L}_m\big|_{J_n \setminus \mathfrak{Q}}$. The orthogonal projector $\mathsf{R}^{n-s} \to \mathfrak{L}_m\big|_{J_n \setminus \mathfrak{Q}}$ transfers Z to $\mathfrak{L}_m\big|_{J_n \setminus \mathfrak{Q}}$.

Set $\mathbf{V} = \mathbf{P}\big|_{J_n \setminus \mathfrak{Q}}$ and $\mathbf{W} = \mathbf{P}\big|_{\mathfrak{Q}}$ for the convenience of notation. The matrix of the projection operator $\boldsymbol{\Pi}_{J_n \setminus \mathfrak{Q}}$ can be derived as follows.

Proposition 3.3 *Assume that the matrix* $\mathbf{I}_s - \mathbf{W}\mathbf{W}^{\mathrm{T}}$ *is nonsingular. Then the matrix of the orthogonal projection operator* $\boldsymbol{\Pi}_{J_n \setminus \mathfrak{Q}}$ *has the form*

$$\boldsymbol{\Pi}_{J_n\setminus\mathfrak{Q}} = \mathbf{V}\mathbf{V}^{\mathrm{T}} + \mathbf{V}\mathbf{W}^{\mathrm{T}}(\mathbf{I}_s - \mathbf{W}\mathbf{W}^{\mathrm{T}})^{-1}\mathbf{W}\mathbf{V}^{\mathrm{T}}. \tag{3.11}$$

3.2 LRR and Associated Characteristic Polynomials

The theory of the linear recurrence relations and associated characteristic polynomials is well known (for example, see [11, Chap. V, Sect. 4]). Here we provide a short survey of the results which are most essential for understanding SSA forecasting.

Definition 3.1 A time series $\mathbb{S}_N = \{s_i\}_{i=1}^N$ is *governed by an LRR*, if there exist $a_1, \ldots, a_t$ such that

$$s_{i+t} = \sum_{k=1}^{t} a_k s_{i+t-k},\ 1 \le i \le N-t,\ a_t \ne 0,\ t < N. \tag{3.12}$$

The number t is called the order of the LRR, $a_1, \ldots, a_t$ are the coefficients of the LRR. If $t = r$ is the minimal order of an LRR that governs the time series S_N, then the corresponding LRR is called *minimal* and we say that the time series $\mathbb{S}_N$ has *finite-difference dimension* r.

Note that if the minimal LRR governing the signal $\mathbb{S}_N$ has order r with $r < N/2$, then $\mathbb{S}_N$ has rank r (see Sect. 2.3.1.2 for the definition of the series of finite rank).

Definition 3.2 A polynomial $P_t(\mu) = \mu^t - \sum_{k=1}^t a_k \mu^{t-k}$ is called a *characteristic polynomial* of the LRR (3.12).

Let the time series $\mathbb{S}_\infty = (s_1, \ldots, s_n, \ldots)$ satisfy the LRR (3.12) with $a_t \ne 0$ and $i \ge 1$. Consider the characteristic polynomial of the LRR (3.12) and denote its different (complex) roots by $\mu_1, \ldots, \mu_p$ with $1 \le p \le t$. All these roots are non-zero as $a_t \ne 0$. Let the multiplicity of the root μ_m be k_m, where $1 \le m \le p$ and $k_1 + \ldots + k_p = t$. The following well-known result (see e.g. [21, Theorem 5.3] or [27]) provides an explicit form for the series which satisfies the LRR.

Theorem 3.1 *The time series $\mathbb{S}_\infty = (s_1, \ldots, s_n, \ldots)$ satisfies the LRR (3.12) for all $i \ge 0$ if and only if*

$$s_n = \sum_{m=1}^{p} \left(\sum_{j=0}^{k_m-1} c_{mj} n^j \right) \mu_m^n, \tag{3.13}$$

where the complex coefficients c_{mj} depend on the first t points $s_1, \ldots, s_t$.

For the real-valued time series, Theorem 3.1 implies that the class of time series governed by the LRRs consists of sums of products of polynomials, exponentials and sinusoids.

3.2.1 Roots of the Characteristic Polynomials

Let the series $\mathbb{S}_N = (s_1, \ldots, s_N)$ be governed by an LRR (3.12) of order t. Let $\mu_1, \ldots, \mu_p$ be pair-wise different (complex) roots of the characteristic polynomial $P_t(\mu)$. As $a_t \neq 0$, all these roots are not equal to zero. We also have $k_1 + \ldots + k_p = t$, where k_m are the multiplicities of the roots μ_m $(m = 1, \ldots, p)$.

Denote $s_n(m, j) = n^j \mu_m^n$ for $1 \le m \le p$ and $0 \le j \le k_m - 1$. Theorem 3.1 tells us that the general solution of the equation (3.12) is

$$s_n = \sum_{m=1}^{p} \sum_{j=0}^{k_m-1} c_{mj} s_n(m, j) \tag{3.14}$$

with certain complex c_{mj}. The coefficients c_{mj} are defined by $s_1, \ldots, s_t$, the first t elements of the series $\mathbb{S}_N$.

Thus, each root μ_m produces a component

$$s_n^{(m)} = \sum_{j=0}^{k_m-1} c_{mj} s_n(m, j) \tag{3.15}$$

of the series $\mathbb{S}_N$. Let us fix m and consider the m-th component in the case $k_m = 1$, which is the main case in practice. Set $\mu_m = \rho e^{\mathrm{i}2\pi\omega}$, $\omega \in (-1/2, 1/2]$, where $\rho > 0$ is the modulus (absolute value) of the root and $2\pi\omega$ is its polar angle.

If ω is either 0 or 1/2, then μ_m is a real root of the polynomial $P_t(\mu)$ and the series component $s_n^{(m)}$ is real and is equal to $c_{m0}\mu_m^n$. This means that $s_n^{(m)} = A\rho^n$ for positive μ_m and $s_n^{(m)} = A\,(-1)^n \rho^n = A\rho^n \cos(\pi n)$ for negative μ_m. This last case corresponds to the exponentially modulated saw-tooth sequence.

All other values of ω lead to complex μ_m. In this case, P_t has a complex conjugate root $\mu_l = \rho e^{-\mathrm{i}2\pi\omega}$ of the same multiplicity $k_l = 1$. We thus can assume $0 < \omega < 1/2$ and describe a pair of conjugate roots by the pair of real numbers (ρ, ω) with $\rho > 0$ and $\omega \in (0, 1/2)$.

By adding up the components $s_n^{(m)}$ and $s_n^{(l)}$ corresponding to these conjugate roots we obtain the real series $A\rho^n \cos(2\pi\omega n + \varphi)$, with A and φ expressed in terms of c_{m0} and c_{l0}. The frequency ω can be expessed in the form of the period $T = 1/\omega$ and vice versa.

The asymptotic behaviour of $s_n^{(m)}$ mainly depends on $\rho = |\mu_m|$. Let us consider the simplest case $k_m = 1$ as above. If $\rho < 1$, then $s_n^{(m)}$ rapidly tends to zero and asymptotically has no influence on the whole series (3.14). Alternatively, the root with $\rho > 1$ and $|c_{m0}| \neq 0$ leads to a rapid increase of $|s_n|$ (at least, of a certain subsequence of $\{|s_n|\}$).

Let r be the finite-difference dimension of a series $\mathbb{S}_N$. Then the characteristic polynomial of the minimal LRR of $\mathbb{S}_N$ has order r and it has r roots. The same series satisfies many other LRRs of dimensions $t > r$. Consider any such LRR (3.12) with $t > r$. The characteristic polynomial $P_t(\mu)$ of the LRR (3.12) has t roots with r

roots (we call them the *main roots*) coinciding with the roots of the minimal LRR. The other $t-r$ roots are *extraneous*: in view of the uniqueness of the representation (3.15), the coefficients c_{mj} corresponding to these roots are equal to zero. However, the LRR (3.12) governs a wider class of series than the minimal LRR.

Since the roots of the characteristic polynomial specify its coefficients uniquely, they also determine the corresponding LRR. Consequently, by removing the extraneous roots of the characteristic polynomial $P_t(\mu)$, corresponding to the LRR (3.12), we can obtain the polynomial describing the minimal LRR of the series.

Example 3.1 (*Annual periodicity*) Assume that the series $\mathbb{S}_N$ has zero mean and period 12. Then it can be expressed as a sum of six harmonics:

$$s_n = \sum_{k=1}^{5} c_k \cos(2\pi nk/12 + \varphi_k) + c_6 \cos(\pi n). \tag{3.16}$$

Under the condition $c_k \neq 0$ for $k = 1, \ldots, 6$, the series has finite-difference dimension 11. In other words, the characteristic polynomial of the minimal LRR governing the series (3.16) has 11 roots. All these roots have modulus 1. The real root -1 corresponds to the last term in (3.16). The harmonic term with frequency $\omega_k = k/12$ ($k = 1, \ldots, 5$) generates two complex conjugate roots $\exp(\pm i2\pi k/12)$, which have polar angles $\pm 2\pi k/12$. □

3.2.2 Min-Norm LRR

Consider a time series $\mathbb{S}_N$ of rank r governed by an LRR. Let L be the window length ($r < \min(L, K)$, $K = N - L + 1$), $\mathbf{S}$ be the trajectory matrix of $\mathbb{S}_N$, $\mathcal{S}$ be its trajectory space, $P_1, \ldots, P_r$ form an orthonormal basis of $\mathcal{S}$ and $\mathcal{S}^\perp$ be the orthogonal complement to $\mathcal{S}$. Denote $A = (a_{L-1}, \ldots, a_1, -1)^\mathrm{T} \in \mathcal{S}^\perp$, $a_{L-1} \neq 0$. Then the time series $\mathcal{S}$ satisfies the LRR

$$s_{i+(L-1)} = \sum_{k=1}^{L-1} a_k s_{i+(L-1)-k}, \quad 1 \leq i \leq K. \tag{3.17}$$

Conversely, if a time series is governed by an LRR (3.17), then the LRR coefficients $B = (a_{L-1}, \ldots, a_1)^\mathrm{T}$ complemented with -1 yield the vector $\begin{pmatrix} B \\ -1 \end{pmatrix} \in \mathcal{S}^\perp$. Note that any LRR that governs a time series can be treated as a forward linear prediction. In addition, if we consider a vector in $\mathcal{S}^\perp$ with -1 as the first coordinate, then we obtain the so-called backward linear prediction [46].

For any matrix $\mathbf{A}$, we denote by $\underline{\mathbf{A}}$ the matrix $\mathbf{A}$ with the last row removed and by $\overline{\mathbf{A}}$ the matrix $\mathbf{A}$ without the first row.

From the viewpoint of prediction, the LRR governing a time series of rank r has coefficients derived from the condition $\underline{\mathbf{S}}^{\mathrm{T}} B = (s_L, \ldots, s_N)^{\mathrm{T}}$. This system of linear equations may have several solutions, since the vector $(s_L, \ldots, s_N)^{\mathrm{T}}$ belongs to the column space of the matrix $\underline{\mathbf{S}}^{\mathrm{T}}$. It is well-known that the least-squares solution expressed by the Moore–Penrose pseudo-inverse to $\underline{\mathbf{S}}^{\mathrm{T}}$ yields the vector B with minimum norm (the solution for the method of total least squares coincides with it). It can be shown that this minimum-norm solution B_{LS} can be expressed as

$$B_{\mathrm{LS}} = (a_{L-1}, \ldots, a_1)^{\mathrm{T}} = \frac{1}{1-\nu^2} \sum_{i=1}^{r} \pi_i \underline{P_i}, \tag{3.18}$$

where π_i are the last coordinates of P_i and $\nu^2 = \sum_{i=1}^{r} \pi_i^2$.

Thus, one of the vectors from $\mathbb{S}^{\perp}$, which equals $A_{\mathrm{LS}} = \begin{pmatrix} B_{\mathrm{LS}} \\ -1 \end{pmatrix}$, has a special significance and the corresponding LRR is called the *min-norm LRR*; it provides the min-norm (forward) prediction. Similarly, we can derive a relation for the min-norm backward prediction.

It is shown in [21, Proposition 5.5] and [31] that the forward min-norm prediction vector A_{LS} is the normalized (so that its last coordinate is equal to -1) projection of the L-th coordinate vector $\mathbf{e}_L$ on $\mathbb{S}^{\perp}$, the orthogonal complement to the signal subspace. Therefore, the min-norm prediction vector depends on the signal subspace only.

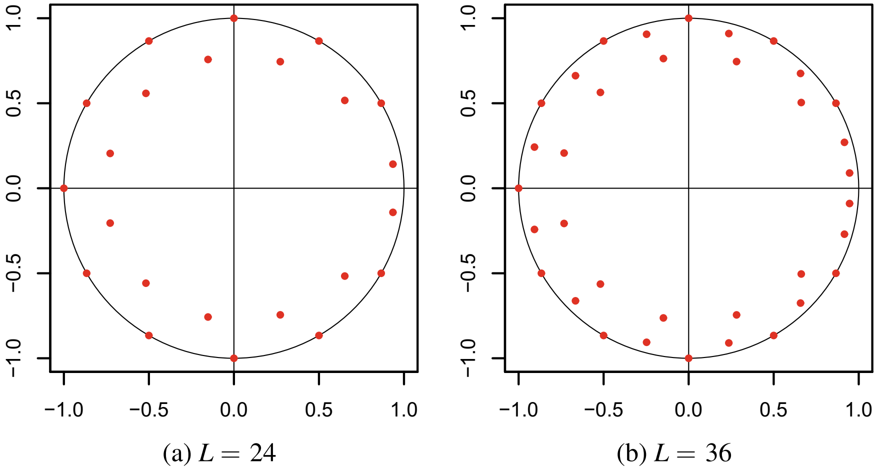
The following property demonstrates the importance of the minimum norm of the LRR coefficients for noise reduction.

Proposition 3.4 *Let $\mathbb{X}_N = \mathbb{S}_N + \mathbb{P}_N$, where $\mathbb{P}_N$ is a stationary white noise with zero mean and variance σ^2, X and S be L-lagged vectors of $\mathbb{X}_N$ and $\mathbb{S}_N$ correspondingly and $C \in \mathsf{R}^{L-1}$. Then for $x = C^{\mathrm{T}}\overline{S}$ and $\widetilde{x} = C^{\mathrm{T}}\overline{X}$, we have $\mathsf{E}\widetilde{x} = x$ and $\mathsf{D}\widetilde{x} = \|C\|^2\sigma^2$.*

The proof directly follows from the equality $\mathsf{D}\sum_{i=1}^{L-1} c_i(y_i + \varepsilon_i) = \mathsf{D}\sum_{i=1}^{L-1} c_i\varepsilon_i = \|C\|^2\sigma^2$, where $C = (c_1, \ldots, c_{L-1})^{\mathrm{T}}$ and ε_i, $i = 1, \ldots, L-1$ are i.i.d. random variables with zero mean and variance σ^2.

If $X = X_K$ is the last lagged vector of $\mathbb{S}_N$, then $\widetilde{x} = C^{\mathrm{T}}\overline{X}_K$ can be considered as a forecasting formula applied to a noisy signal and $\|C\|^2$ regulates the variance of this forecast.

The following property of the min-norm LRR, which was derived in [32], is extremely important for forecasting: all extraneous roots of the min-norm LRR lie inside the unit circle of the complex plane. Example 3.2, where the min-norm LRR is used, illustrates this property giving us hope that in the case of real-life series (when both the min-norm LRR and the related initial data are perturbed) the terms related to the extraneous roots in (3.13) only slightly influence the forecast. Moreover, bearing in mind the results concerning the distribution of the extraneous roots (see [38, 47]), we can expect that the extraneous summands cancel each other out.



(a) $L = 24$ (b) $L = 36$

Fig. 3.1 Annual periodicity: main and extraneous roots

Example 3.2 (*Annual periodicity and extraneous roots*) Let us consider the series (3.16) from Example 3.1 and the min-norm LRR, which is not minimal. Let N be large enough. If we select certain $L \geq 12$ and take $r = 11$ and $\mathfrak{L}_r = \mathcal{S}(\mathbb{S}_N)$, then the vector $R = (a_{L-1}, \ldots, a_1)^{\mathrm{T}}$ defined in (3.18) produces the LRR (3.17), which is not minimal but governs the series (3.16).

Let us take $c_i = i - 1$, $\varphi_1 = \ldots = \varphi_5 = 0$ and $L = 24,\ 36$. The roots of the characteristic polynomials of the LRR (3.17) are depicted in Fig. 3.1. We can see that the main 11 roots of the polynomial compose 11 of 12 vertices of a regular dodecagon and lie on the unit circle in the complex plane. Twelve ($L = 24$) and twenty four ($L = 36$) extraneous roots have smaller moduli. □

Remark 3.5 Note that the min-norm LRR forms the basis for the SSA forecasting methods introduced in Sect. 3.1 (see [21, Sect. 2.1]). In particular, R-forecasting uses the estimated min-norm LRR for forecasting: compare the formulas (3.1) and (3.18) for the coefficients of the LRRs.

3.3 Recurrent Forecasting as Approximate Continuation

Exact continuation is hardly practically significant. Indeed, it seems unwise to assume that a real-life time series is governed by some LRR of relatively small dimension. Therefore, we need to consider approximate continuation; it is of much greater importance in practice than exact continuation. In this section we consider approximate continuation with the help of recurrent forecasting. Most discussions are also relevant for other SSA forecasting algorithms.

3.3.1 Approximate Separability and Forecasting Errors

Let $\mathbb{X}_N = \mathbb{X}_N^{(1)} + \mathbb{X}_N^{(2)}$ and suppose that the time series $\mathbb{X}_N^{(1)}$ admits a recurrent continuation. Denote by d the dimension of the minimal recurrence relation governing $\mathbb{X}_N^{(1)}$. If $d < \min(L, N-L+1)$, then $d = \operatorname{rank}_L(\mathbb{X}_N^{(1)})$.

If $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ are strongly separable for some window length L, then the trajectory space of $\mathbb{X}_N^{(1)}$ can be found and we can perform recurrent continuation of the series $\mathbb{X}_N^{(1)}$ by the method described in Sect. 3.1.2.1. We now assume that $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ are approximately strongly separable and discuss the problem of approximate continuation (forecasting) of the series $\mathbb{X}_N^{(1)}$ in the subspace $\mathfrak{L}_r$. The choice of $\mathfrak{L}_r$ is described in Sect. 3.1.3. If the choice is proper, $r = d$.

The series of forecasts y_n $(n > N)$ defined by (3.2) generally does not coincide with the recurrent continuation of the series $\mathbb{X}_N^{(1)}$. The deviation between these two series makes the forecasting error. This error has two origins. The main origin is the difference between the linear space $\mathfrak{L}_r$ and $\mathfrak{X}^{(L,1)}$, the trajectory space of the series $\mathbb{X}_N^{(1)}$ (some inequalities connecting the perturbation of the LRR (3.2) with that of $\mathfrak{X}^{(L,1)}$ are derived in [36], see also the end of Sect. 2.3.3 for a related discussion). Since the LRR (3.2) is produced by the vector R and the latter is strongly related to the space $\mathfrak{L}_r$, the discrepancy between $\mathfrak{L}_r$ and $\mathfrak{X}^{(L,1)}$ produces an error in the LRR governing the series of forecasts. In particular, the finite-difference dimension of the series of forecasts y_n $(n > N)$ is generally larger than r.

The other origin of the forecasting error lies in the initial data used to build the forecast. In the case of recurrent continuation, the initial data is $x_{N-L+2}^{(1)}, \ldots, x_N^{(1)}$, where $x_n^{(1)}$ is the n-th term of the series $\mathbb{X}_N^{(1)}$. In the Basic SSA R-forecasting algorithm, the initial data consists of the last $L-1$ terms $y_{N-L+2}, \ldots, y_N$ of the reconstructed series. Since generally $x_n^{(1)} \neq y_n$, the initial data used in LRR is a source of forecasting errors. The splitting of the whole error into two parts is investigated in [14] by simulations. For L close to $N/2$, these parts are comparable while for small L the contribution of the error caused by the wrong reconstruction is larger.

On the other hand, if the quality of approximate separability of $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ is rather good and we select the proper eigentriples associated with $\mathbb{X}^{(1)}$, then we can expect that the linear spaces $\mathfrak{L}_r$ and $\mathfrak{X}^{(L,1)}$ are close. Therefore, the coefficients in the LRR (3.2) are expected to be close to those of the LRR governing the recurrent continuation of the series $\mathbb{X}_N^{(1)}$. Similarly, approximate separability implies that the reconstructed series y_n is close to $x_n^{(1)}$ and therefore the error due to the imprecision of the initial data used for forecasting is also small. As a result, in this case we can expect that the Basic SSA R-forecasting procedure provides a reasonably accurate approximation to the recurrent continuation of $\mathbb{X}_N^{(1)}$, at least in the first few steps.

Remark 3.6 Since the forecasting procedure contains two generally unrelated parts, namely, estimation of the LRR and estimation of the reconstruction, we can modify these two parts of the algorithm separately. For example, for forecasting a signal, the LRR can be applied to the initial time series if the last points of the reconstruction are expected to be biased. Another modification of the forecasting procedure is considered in [14]; it is based on the use of different window lengths for estimation of the LRR and for reconstruction of the time series.

3.3.2 Approximate Continuation and Characteristic Polynomials

In this section, we continue discussing the errors of separability and forecasting. The discrepancy between $\mathcal{L}_r$ and $\mathcal{X}^{(L,1)}$ will be assessed in terms of the characteristic polynomials.

We have three LRRs: (i) the minimal LRR of order r governing the time series $\mathbb{X}_N^{(1)}$, (ii) the continuation LRR of order $L-1$, which governs $\mathbb{X}_N^{(1)}$ but also produces $L-r-1$ extraneous roots in its characteristic polynomial P_{L-1}, and (iii) the forecasting min-norm LRR governing the series of forecasts y_n $(n > N)$.

The characteristic polynomial $P_{L-1}^{(x)}$ of the forecasting LRR and continuation polynomial P_{L-1} have $L-1$ roots. If $\mathcal{L}_r$ and $\mathcal{X}^{(L,1)}$ are close, then the coefficients of the continuation and forecasting recurrence relations must be close too. Therefore, all simple roots of the forecasting characteristic polynomial $P_{L-1}^{(x)}$ must be close to the roots of the continuation polynomial P_{L-1}. The roots μ_m with multiplicities $k_m > 1$ could be perturbed in a more complex manner.

Example 3.3 (*Perturbation of the multiple roots*) Let us consider the series $\mathbb{X}_N$ with

$$x_n = (A + 0.1\,n) + \sin(2\pi n/10), \quad n = 0, \ldots, 199.$$

Evidently, $\mathbb{X}_N = \mathbb{X}_N^{(1)} + \mathbb{X}_N^{(2)}$ with the linear series $\mathbb{X}_N^{(1)}$ defined by $x_{n+1}^{(1)} = A + 0.1\,n$ and the harmonic series $\mathbb{X}_N^{(2)}$ corresponding to $x_{n+1}^{(2)} = \sin(2\pi n/10)$.

The series $\mathbb{X}_N$ has rank 4 and is governed by the minimal LRR of order 4. Therefore, any LRR governing $\mathbb{X}_N$ produces a characteristic polynomial with four main roots. These main roots do not depend on A; the linear part of the series generates one real root $\mu = 1$ of multiplicity 2, while the harmonic series corresponds to two complex conjugate roots $\rho e^{\pm \mathrm{i}2\pi\omega}$ with modulus $\rho = 1$ and frequency $\omega = 0.1$.

Our aim is to forecast the series $\mathbb{X}_N^{(1)}$ for $A = 0$ and $A = 50$ with the help of the Basic SSA R-forecasting algorithm. In both cases, we take the window length $L = 100$ and choose the eigentriples that correspond to the linear part of the initial time series $\mathbb{X}_N$. (For $A = 0$ we take the two leading eigentriples, while for $A = 50$ the appropriate eigentriples have the ordinal numbers 1 and 4.) Since the series $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ are not exactly separable for any A and any choice of L, we deal with approximate separability. The forecasting polynomials $P_{L-1}^{(x)}$ with $A = 0$ and $A = 50$ demonstrate different splitting of the double root $\mu = 1$ into two simple ones. For $A = 0$ there appear two complex conjugate roots with $\rho = 1.002$ and $\omega = 0.0008$, while in the case $A = 50$ we obtain two real roots equal to 1.001 and 0.997. All extraneous roots are smaller than 0.986. This means that for $A = 0$ the linear series $\mathbb{X}_N^{(1)}$ is approximated by a low-frequency harmonic with slightly increasing exponential amplitude. In the case $A = 50$, the approximating series is the sum of two exponentials, one of them is slightly increasing and another one is slightly decreasing. Therefore, we have different long-term forecasting formulas: oscillating for $A = 0$ and exponentially increasing for $A = 50$. For short-term forecasting, this difference is not important. □

Let us consider the part of the forecasting error caused by errors in the initial data, that is, in the reconstruction of the forecasted series component. If the LRR is not minimal ($L > r + 1$), then the corresponding characteristic polynomial P_{L-1} has $L - 1 - r$ extraneous roots. If there is no reconstruction error, then the extraneous roots do not affect the forecast since the coefficients c_{mj} in (3.13) for the corresponding summands are equal to zero. However, if one applies the LRR to the perturbed initial terms, then the extraneous roots start to affect the forecasting results. The extraneous roots of the min-norm LRR lie within the unit circle and their effect on the forecasting decreases for long-term forecasting. Unfortunately, the minimal LRR is not appropriate for forecasting as it is very sensitive to errors in the initial data. Hence the presence of extraneous roots should be taken into account.

In the case of approximate separability, the min-norm LRR is found approximately. As a consequence, the extraneous roots can have absolute values larger than 1. The extraneous roots with moduli greater than 1 are very hazardous, since the extraneous summand μ^n in (3.13), caused by an extraneous root μ with $|\mu| > 1$, grows indefinitely. Therefore, it is important to look at the extraneous roots of the LRR used for forecasting.

If the forecasted series component $\mathbb{X}_N^{(1)}$ is the signal, then the main roots can be called signal roots. Note that the knowledge of extraneous roots should be used both for finding the parametric form (3.13) of the signal (then we should identify the signal roots and remove the extraneous roots) and also for forecasting the signal (then we do not need to know the values of the roots but we would like to have no extraneous roots outside the unit circle).

Since the forecasting LRR is fully determined by the roots of its characteristic polynomial, certain manipulations with the polynomial roots can be performed to modify the R-forecasting procedure.

- Let the main roots of the min-norm LRR of order $L - 1$ be identified or estimated (e.g. by ESPRIT, see Sect. 3.8.2). For example, for a time series with the signal components which are not decreasing, the estimated main roots typically have maximal moduli (since the extraneous roots lie inside the unit circle). Thereby, we obtain the estimated minimal LRR, which is also the min-norm LRR of order r. However, it follows from the definition of the minimum norm that the norm of the coefficients of the minimal LRR is larger than that of the min-norm LRR of order $L - 1$ for $L > r + 1$. Therefore, the forecast by the minimal LRR is more sensitive to errors in the initial data. Simulations demonstrate that in most cases the use of the minimal LRR does not give the most accurate forecast and, moreover, such forecast is often rather unstable.

- A safe way of correcting the LRR is by adjusting the identified main roots when an additional information about the time series is available. For example, if we know that the forecasted oscillations have stationary periodicities with constant amplitudes, then we know that the root moduli are equal to one and therefore the corresponding roots can be substituted with $\mu' = \mu/\|\mu\|$. If there is a periodicity with known period in the time series, then we can correct the arguments of the corresponding roots (for example, to 1/12, 1/6 and so on for a monthly data with seasonality).
- If the main roots have been estimated, then an explicit formula for the time series values in the form (3.13) can be obtained (using estimates of c_{mj} obtained by the least-squares method) and the forecast can be produced by this explicit formula. However, an explicit forecast needs root estimation, whereas R-forecasting does not need root estimation and therefore it is more robust.

3.4 Confidence Bounds for the Forecasts

There are several conventional ways of estimating the accuracy of a forecast. Most of them can be applied for forecasting of the signal in the signal plus noise model.

1. Theoretical confidence intervals can be constructed if the model of time series is known and there are theoretical results about the distribution of the forecast.
2. Bootstrap confidence intervals can be constructed if the model of the signal is estimated in the course of analysis.
3. The accuracy of forecasting can be tested by removal of the last few points and then forecasting their values (so-called *retrospective forecast*). This can be repeated with the cut made at different points.
4. If we are not interested in the retrospective forecast (we really need to forecast the future) and cannot reliably build an SSA model (as well as any other model) then we can use the following approach: we build a large number of SSA forecasts (e.g. using a variety of L and different but reasonable grouping schemes) and compare the forecast values at the horizon we are interested in. If the forecasts are going all over the place then we cannot trust any of them. If however the variability of the constructed forecasts is small then we (at least partly) may trust them; see [39] for details and examples.

If there is a set of possible models, then the model can be chosen by minimizing the forecasting errors. An adjustment taking into account the number of parameters in the models should be made similar to the methods based on characteristics like the Akaike information criterion or by using the degrees-of-freedom adjustments, see discussion in Sect. 2.4.4.3.

There are not enough theoretical results which would help in estimating the accuracy of SSA forecasts theoretically. Below in this section we consider bootstrap confidence intervals in some detail. Since construction of bootstrap confidence intervals is very similar to that of the Monte Carlo confidence intervals, we also consider

Monte Carlo techniques for the investigation of the accuracy of reconstruction and forecasting. Note that by constructing bootstrap confidence intervals for forecasting values we also obtain confidence limits for the reconstructed values.

3.4.1 Monte Carlo and Bootstrap Confidence Intervals

According to the main SSA forecasting assumptions, the component $\mathbb{X}_N^{(1)}$ of the time series $\mathbb{X}_N$ ought to be governed by an LRR of relatively small dimension, and the residual series $\mathbb{X}_N^{(2)} = \mathbb{X}_N - \mathbb{X}_N^{(1)}$ ought to be approximately strongly separable from $\mathbb{X}_N^{(1)}$ for some window length L. In particular, $\mathbb{X}_N^{(1)}$ is assumed to be a finite subseries of an infinite series, which is a recurrent continuation of $\mathbb{X}_N^{(1)}$. These assumptions hold for a wide class of practical series.

To establish confidence bounds for the forecast, we have to apply even stronger assumptions, related not only to $\mathbb{X}_N^{(1)}$, but to $\mathbb{X}_N^{(2)}$ as well. We assume that $\mathbb{X}_N^{(2)}$ is a finite subseries of an infinite random noise series $\mathbb{X}^{(2)}$ that perturbs the signal $\mathbb{X}^{(1)}$.

We only consider Basic SSA R-forecasting method. All other SSA forecasting procedures can be treated analogously.

Let us consider a method of constructing confidence bounds for the signal $\mathbb{X}^{(1)}$ at the moment of time $N + M$. In the unrealistic situation, when we know both the signal $\mathbb{X}^{(1)}$ and the true model of the noise $\mathbb{X}_N^{(2)}$, a direct Monte Carlo simulation can be used to check statistical properties of the forecast value $\widetilde{x}_{N+M}^{(1)}$ relative to the actual value $x_{N+M}^{(1)}$. Indeed, assuming that the rules for the eigentriple selection are fixed, we can simulate Q independent copies $\mathbb{X}_{N,i}^{(2)}$ of the process $\mathbb{X}_N^{(2)}$ and apply the forecasting procedure to Q independent time series $\mathbb{X}_{N,i} \overset{\text{def}}{=} \mathbb{X}_N^{(1)} + \mathbb{X}_{N,i}^{(2)}$. Then the forecasting results will form a sample $\widetilde{x}_{N+M,i}^{(1)}$ $(1 \le i \le Q)$, which should be compared against $x_{N+M}^{(1)}$. In this way, *Monte Carlo confidence bounds* for the forecast can be build.

Since in practice we do not know the signal $\mathbb{X}_N^{(1)}$, we cannot apply this procedure. Let us describe the bootstrap procedure for constructing the confidence bounds for the forecast (for a general methodology of bootstrap, see, for example, [10, Sect. 5]).

For a suitable window length L and the grouping of eigentriples, we have the representation $\mathbb{X}_N = \widetilde{\mathbb{X}}_N^{(1)} + \widetilde{\mathbb{X}}_N^{(2)}$, where $\widetilde{\mathbb{X}}_N^{(1)}$ (the reconstructed series) approximates $\mathbb{X}_N^{(1)}$, and $\widetilde{\mathbb{X}}_N^{(2)}$ is the residual series. Suppose now that we have a (stochastic) model of the residuals $\widetilde{\mathbb{X}}_N^{(2)}$. For instance, we can postulate some model for $\mathbb{X}_N^{(2)}$ and, since $\widetilde{\mathbb{X}}_N^{(1)} \approx \mathbb{X}_N^{(1)}$, apply the same model for $\widetilde{\mathbb{X}}_N^{(2)}$ with the estimated parameters. Then, simulating Q independent copies $\widetilde{\mathbb{X}}_{N,i}^{(2)}$ of the series $\mathbb{X}_N^{(2)}$, we obtain Q series $\mathbb{X}_{N,i} \overset{\text{def}}{=} \widetilde{\mathbb{X}}_N^{(1)} + \widetilde{\mathbb{X}}_{N,i}^{(2)}$ and produce Q forecasting results $\widetilde{x}_{N+M,i}^{(1)}$ in the same manner as in the straightforward Monte Carlo simulation.

More precisely, any time series $\mathbb{X}_{N,i}$ produces its own reconstructed series $\widetilde{\mathbb{X}}_{N,i}^{(1)}$ and its own forecasting linear recurrence relation LRR_i for the same window length

L and the same set of the eigentriples. Starting at the last $L-1$ terms of the series $\widetilde{\mathbb{X}}^{(1)}_{N,i}$, we perform M steps of forecasting with the help of its LRR$_i$ to obtain $\widetilde{x}^{(1)}_{N+M,i}$.

As soon as the sample $\widetilde{x}^{(1)}_{N+M,i}$ $(1 \le i \le Q)$ of the forecasting results is obtained, we can calculate its (empirical) lower and upper quantiles of some fixed level γ and obtain the corresponding confidence interval for the forecast. This interval will be called the *bootstrap confidence interval.* Simultaneously with the bootstrap confidence intervals for the signal forecasting values, we obtain the bootstrap confidence intervals for the reconstructed values. The average of the bootstrap forecast sample (*bootstrap average forecast*) estimates the mean value of the forecast, while the mean square deviation of the sample shows the accuracy of the estimate.

The simplest model for $\widetilde{\mathbb{X}}^{(2)}_N$ is the model of Gaussian white noise. The corresponding hypothesis can be checked with the help of the standard tests for randomness and normality. Another natural approach for noise generation uses the empirical distribution of the residual; this version is implemented in the RSSA package (see [24, Sect. 3.2.1.5]).

3.4.2 Confidence Intervals: Comparison of Forecasting Methods

The aim of this section is to compare different SSA forecasting procedures using several artificial series and the Monte Carlo confidence intervals.

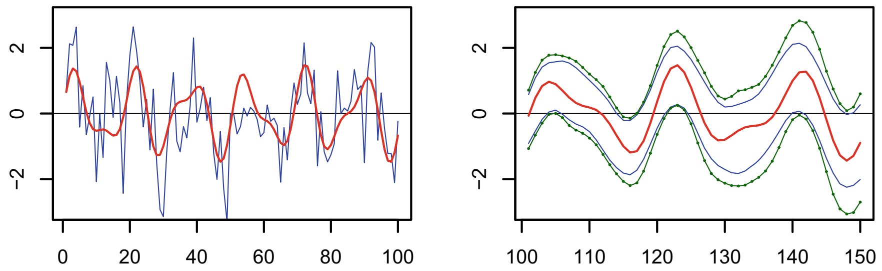
Let $\mathbb{X}_N = \mathbb{X}^{(1)}_N + \mathbb{X}^{(2)}_N$, where $\mathbb{X}^{(2)}_N$ is Gaussian white noise with standard deviation σ. Assume that the signal $\mathbb{X}^{(1)}_N$ admits a recurrent continuation. We shall perform a forecast of the series $\mathbb{X}^{(1)}_N$ for M steps using different versions of SSA forecasting and appropriate eigentriples associated with $\mathbb{X}^{(1)}_N$. Several effects will be illustrated in the proposed simulation study. First, we shall compare some forecasting methods from the viewpoint of their accuracy. Second, we shall demonstrate the role of the proper choice of the window length.

We will consider two examples. In both of them, $N = 100$, $M = 50$ and the standard deviation of the Gaussian white noise $\mathbb{X}^{(2)}_N$ is $\sigma = 1$. The confidence intervals are obtained in terms of the 2.5% upper and lower quantiles of the corresponding empirical c.d.f. using the sample size $Q = 10000$.

Periodic signal: recurrent and vector forecasting. Let us consider a periodic signal $\mathbb{X}^{(1)}_N$ of the form

$$x^{(1)}_n = \sin(2\pi n/17) + 0.5\sin(2\pi n/10).$$

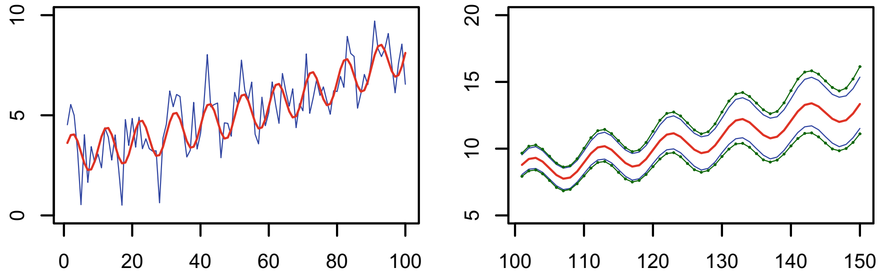
The series $\mathbb{X}^{(1)}_N$ has difference dimension 4, and we use four leading eigentriples for its forecasting under the choice $L = 50$. The initial series $\mathbb{X}_N = \mathbb{X}^{(1)}_N + \mathbb{X}^{(2)}_N$ and the signal $\mathbb{X}^{(1)}_N$ (red thick line) are depicted in Fig. 3.2a.



(a) Periodic signal and the initial series

(b) Confidence intervals

Fig. 3.2 Comparison of recurrent and vector forecasts



(a) The signal and the initial series

(b) Two confidence intervals

Fig. 3.3 Separability and forecasting

Let us apply the Monte Carlo simulation for Basic SSA recurrent and vector forecasting algorithms. Figure 3.2b shows the confidence Monte Carlo intervals for both methods and the true continuation of the signal $\mathbb{X}_N^{(1)}$ (red thick line). Confidence intervals for R-forecasting are marked by dots, while blue thin lines correspond to V-forecasting. We can see that these intervals practically coincide for relatively small numbers of forecasting steps, while V-forecasting has some advantage in the long-term forecasting.

Separability and forecasting. Consider the series $\mathbb{X}_N^{(1)}$ with

$$x_n^{(1)} = 3a^n + \sin(2\pi n/10), \quad a = 1.01.$$

This series is governed by an LRR of dimension 3. Consider Basic SSA R-forecasting for up to 50 points of the signal values $x_{N+j}^{(1)}$ using the series $\mathbb{X}_N = \mathbb{X}_N^{(1)} + \mathbb{X}_N^{(2)}$. We compare two window lengths, $L = 15$ and $L = 50$. The first three eigentriples are chosen for the reconstruction in both choices of L. The series $\mathbb{X}_N$ and the signal $\mathbb{X}_N^{(1)}$ (red thick line) are depicted in Fig. 3.3a.

Figure 3.3b shows that the Monte Carlo forecasting confidence intervals for $L = 15$ (green thin line with dots) are apparently wider than that for $L = 50$. This is not surprising since the choice $L = 50$ corresponds to better separability. This is confirmed by comparing the values of the separability characteristics. In particular, the **w**-correlation (2.18) between the extracted signal and the residual is equal to 0.0083 for $L = 15$ and it equals 0.0016 for $L = 50$. Recall that the exact separability gives zero value for the **w**-correlation.

3.5 Summary and Recommendations on Forecasting Parameters

Let us summarize the material of the previous sections taking as an example Basic SSA R-forecasting method. Other versions of SSA forecasting can be described and commented on similarly.

1. *Statement of the problem*
   We have a time series $\mathbb{X}_N = \mathbb{X}_N^{(1)} + \mathbb{X}_N^{(2)}$ and need to forecast its component $\mathbb{X}_N^{(1)}$.
2. *The main assumptions*

   - The series $\mathbb{X}_N^{(1)}$ admits a recurrent continuation with the help of an LRR of a relatively small dimension r.
   - There exists L such that the series $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ are approximately strongly separable for the window length L.

3. *Proper choice of parameters*
   Since we have to select the window length L providing a sufficient quality of separability and to find the eigentriples corresponding to $\mathbb{X}_N^{(1)}$, all the major rules and recommendations for the use of Basic SSA are applicable here. Note that in this case we must separate $\mathbb{X}_N^{(1)}$ from $\mathbb{X}_N^{(2)}$, but we do not need to obtain a detailed decomposition of the series $\mathbb{X}_N$.
4. *Specifics and potential pitfalls*

   - Since SSA forecasting procedure needs an estimation of the LRR, some recommendations concerning the window length for reconstruction and forecasting can differ. SSA modifications that use different window lengths for reconstruction and for building the forecasting formula can be used.
   - In Basic SSA, if we enlarge the set of proper eigentriples by some extra eigentriples with small singular values, then the result of reconstruction will essentially be the same. When dealing with forecasting, such an operation can produce large perturbations since the trajectory space $\mathcal{X}^{(L,1)}$ will be perturbed a lot; its dimension will be enlarged, and therefore the LRR governing the forecast will be modified. In this case, the magnitude of the extra singular values is not important but the location of the extraneous roots of the characteristic polynomials is important.

5. *Characteristics of forecasting*
   The following characteristics may be helpful in judging the forecasting quality.

- *Separability characteristics.* All separability characteristics considered in Sect. 2.3.3 are of importance for forecasting.
- *Polynomial roots.* The roots of the characteristic polynomial of the forecasting LRR can give an insight into the behaviour of the forecast. These polynomial roots can be useful in answering the following two questions:

  (a) We expect that the forecast has some particular form (for example, we expect it to be increasing). Do the polynomial roots describe such a possibility? For instance, an exponential growth has to be indicated by a single real root (slightly) greater than 1 but if we try to forecast the annual periodicity, then pairs of complex roots with frequencies $\approx k/12$ have to exist.
  (b) Although extraneous roots of the true min-norm LRR have moduli smaller than 1, the extraneous roots of the estimated LRR can be larger than 1. Since the polynomial roots with moduli greater than 1 correspond to the series components with increasing envelopes (see Sect. 3.2), large extraneous roots may cause problems even in the short-term forecasting. This is a serious pitfall that always has to be closely monitored.

- *Verticality coefficient.* The verticality coefficient ν^2 is the squared cosine of the angle between the space $\mathcal{L}_r$ and the vector $\mathbf{e}_L$. The condition $\nu^2 < 1$ is necessary for forecasting. The norm of the min-norm LRR (3.18) coefficients is equal to $\nu^2/(1-\nu^2)$. This characteristic reflects the ability of the LRR to decrease the noise level, see Proposition 3.4. If ν^2 is close to 1, then the norm is very large. This often means that too many extra eigentriples are taken for the reconstruction of $\mathbb{X}_N^{(1)}$ (alternatively, the whole approach is inadequate).

6. *The role of the initial data*

   Apart from the number M of forecast steps, the formal parameters of Basic SSA R-forecasting algorithm are the window length L and the set I of eigentriples describing $\mathbb{X}_N^{(1)}$. These parameters determine both the forecasting LRR (3.1) and the initial data used in the forecasting formula. Evidently, the forecasting result significantly depends on this data, especially when the forecasting LRR has extraneous roots.

   The SSA R-forecasting method uses the last $L-1$ terms $\widetilde{x}_{N-L+2}^{(1)}, \ldots, \widetilde{x}_N^{(1)}$ of the reconstructed series $\widetilde{\mathbb{X}}_N^{(1)}$ as the initial data for forecasting. In view of the properties of the diagonal averaging, the last (and the first) terms of the series $\mathbb{X}_N^{(1)}$ are usually reconstructed with poorer precision than the middle ones. This effect may cause substantial forecasting errors. For example, any linear (and nonconstant) series $x_n = an + b$ is governed by the minimal LRR $x_n = 2x_{n-1} - x_{n-2}$, which does not depend on a and b. The parameters a and b used in the forecast are completely determined by the initial data x_1 and x_2. Evidently, errors in this data may considerably modify the forecast.

   Thus, it is important to check the last points of the reconstructed series (for example, to compare them with the expected future behaviour of the series $\mathbb{X}_N^{(1)}$). Even the use of the last points of the initial series as the initial data in the forecasting formula may improve the forecast.

7. *Reconstructed series and LRRs*
   In the situation of strong separability between $\mathbb{X}_N^{(1)}$ and $\mathbb{X}_N^{(2)}$ and proper eigentriple selection, the reconstructed series is governed by the LRR which exactly corresponds to the series $\mathbb{X}_N^{(1)}$. Discrepancies in this correspondence indicate on possible errors: insufficient separability (which can be caused by a bad choice of the forecasting parameters) or general inadequacy of the model. We can suggest the following ways of testing for the presence of these errors and reducing them.

   - *Global discrepancies.* Rather than using an LRR for forecasting, we can use it for approximation of either the whole reconstructed series or its subseries. For instance, if we take the first terms of the reconstructed series as the initial data (instead of the last ones) and make $N - L + 1$ steps of the procedure, we can check whether the reconstructed series can be globally approximated with the help of the LRR.
   - *Local discrepancies.* The procedure above corresponds to the long-term forecasting. To check the short-term correspondence of the reconstructed series and the forecasting LRR, one can apply a slightly different method which is called the multistart recurrent continuation. In it, for a relatively small M we perform M steps of the multistart recurrent continuation procedure, modifying the initial data from $(\widetilde{x}_1^{(1)}, \ldots, \widetilde{x}_{L-1}^{(1)})$ to $(\widetilde{x}_{K-M+1}^{(1)}, \ldots, \widetilde{x}_{N-M}^{(1)})$, $K = N - L + 1$. The M-step continuation is computed with the help of the forecasting LRR. The results should be compared with $\widetilde{x}_{L+M-1}^{(1)}, \ldots, \widetilde{x}_N^{(1)}$. Since both the LRR and the initial data have errors, the local discrepancies for small M are usually more informative than the global ones. Moreover, by using different M we can estimate the maximal number of steps for a reasonable forecast.

   Note that if the discrepancies are small then this does not necessarily imply that the forecasting is accurate. This is because the forecasting LRR is tested on the same points that were used for the calculation of the forecasting LRR.

8. *Forecasting stability and reliability*
   While the correctness of the forecast cannot be checked using the data only, the reliability of the forecast can be examined. Let us mention several methods for carrying out such an examination.

   - *Different algorithms.* We can try different forecasting algorithms (for example, recurrent and vector) with the same parameters. If their results approximately coincide, we have an argument in favour of forecasting stability.
   - *Different window lengths.* If the separability characteristics are stable under small variation in the window length L, we can compare the forecasts for different L.
   - *Forecasting of truncated series.* We can truncate the initial series $\mathbb{X}_N$ by removing the last few terms from it. If the separability conditions are stable under this operation, then we can forecast the truncated terms and compare the result with the initial series $\mathbb{X}_N$ and the reconstructed series $\widetilde{\mathbb{X}}_N^{(1)}$ obtained without truncation. If the forecast is regarded as adequate, then its continuation by the same LRR can be regarded as reliable.

9. *Confidence intervals*
   Confidence intervals discussed in Sect. 3.4 give important additional information about the accuracy and stability of forecasts.

3.6 Case Study: 'Fortified Wine'

To illustrate the SSA forecasting techniques, we consider the time series 'Fortified wine' (monthly volumes of fortified wine sales in Australia from January 1984 till June 1994, Fig. 2.16). Naturally, time series forecasting should be based on the preliminary time series investigation. We examine both the initial time series of length 174 and its subseries consisting of the first 120 points. We name the former FORT174 and the latter FORT120.

SSA forecasting should only be applied to a time series governed (may be approximately) by some LRR. Therefore, we start with the study of the series from this point of view.

Linear Recurrence Relation Governing the Time Series

Preliminary analysis shows that the 'FORT174' time series (see Sects. 2.3.1.2 and 2.4.2.2) can be decomposed into a sum of a signal and a noise. For window length $L = 84$, the signal can be reconstructed by means of ET1–11 and the **w**-correlation between the signal component and the noise component is 0.004 which is small enough. Thus, the estimated signal subspace of R^L has dimension 11, the min-norm LRR has dimension $L - 1$ and the reconstructed time series (the signal) can be approximated by a time series governed by this LRR. For the series FORT120 and $L = 60$ the signal also corresponds to ET1–11, the **w**-correlation with the residual is slightly larger (it equals 0.005).

Table 3.1 presents the information for 19 leading roots of the characteristic polynomials corresponding to two estimated min-norm LRRs. The roots (recall that they are complex numbers) are ordered by decreasing their moduli. The label 'compl.' for the 'Type' column of Table 3.1 notes that this line relates to two conjugate complex roots $\rho_j\, e^{\pm \mathrm{i} 2\pi \omega_j}$, $0 < \omega_j < 0.5$. In this case, the period $1/\omega_j$ is listed in the table. The first six rows can be interpreted easily: the rows 1–3 and 5–6 correspond to conjugate complex roots, which produce harmonics with periods 6, 4, 2.4, 12, and 3. Moduli larger than one correspond to harmonics with increasing amplitudes, a modulus smaller than one yield a decreasing amplitude. The forth row of the table corresponds to the real-valued root with modulus 0.997. There are no more signal roots and all other roots are extraneous. All moduli of the extraneous roots are less than one. The columns marked 'ET' indicate the correspondence between the eigentriples and the polynomial roots.

The series is decreasing and therefore the roots with modulus larger than 1 are most probably inadequate. Especially, the leading root (ET6–7) has modulus 1.013 for FORT120 which is a possible reason for an unstable forecast. Also, for FORT120

Table 3.1 Time series FORT174 and FORT120: the leading roots of the characteristic polynomials for the min-norm LRRs

FORT174, $L = 84$					FORT120, $L = 60$				
N	ET	Modulus	Period	Type	N	ET	Modulus	Period	Type
1	6–7	1.003	5.969	Compl.	1	6–7	1.013	5.990	Compl.
2	8–9	1.000	3.994	Compl.	2	8–11	1.007	2.376	Compl.
3	4–5	0.998	2.389	Compl.	3	4–5	1.000	4.001	Compl.
4	1	0.997	No	Real	4	1	0.997	No	Real
5	2–3	0.994	12.002	Compl.	5	2–3	0.994	12.033	Compl.
6	10–11	0.989	3.028	Compl.	6	8–11	0.982	3.002	Compl.
7		0.976	3.768	Compl.	7		0.968	5.311	Compl.
8		0.975	3.168	Compl.	8		0.966	9.635	Compl.
9		0.975	10.212	Compl.	9		0.966	3.688	Compl.
10		0.975	5.480	Compl.	10		0.965	2.268	Compl.

two harmonics are mixed; therefore, two pairs of conjugated roots put into correspondence with four eigentriples ET8–11.

Let us check whether the time series FORT174 is well fitted by the estimated min-norm LRR. The maximum value of the global discrepancy between the reconstructed signal and its approximation by a time series governed by the used LRR (that is, the error of global approximation) is equal to 132 and it is smaller than 10% of the time series values. Note that we use the first 83 points as the initial data for the LRR; so the approximation error is calculated starting from the 84-th point.

Let us consider the minimal LRR of order 11 generated by the estimated signal roots presented in Table 3.1 above the horizontal line. (Recall that there is a one-to-one correspondence between LRRs and the roots of the associated characteristic polynomials.) If we take the points 73–83 as the initial data for this LRR, the series governed by the minimal LRR better approximates the time series (the maximum discrepancy is equal to 94). Thus we conclude that the time series is well approximated by the time series governed by the minimal LRR of order 11. Note that since the long-term forecast by the minimal LRR is very sensitive to the initial data, the choice of points 73–83 as the initial data was rather fortunate. The results for local approximation (discrepancy) are similar (magnitudes of errors are smaller while using the minimal LRR).

Since we know the exact period of the time series periodical component (due to its seasonal behavior), we can adjust the LRR by changing the roots so that they correspond to the periods 6, 4, 2.4, 12 and 3. This LRR of order 11 is called an adjusted minimal LRR. The local approximation errors, corresponding to the adjusted minimal LRR, are slightly smaller than for the minimal LRR.

The analytic form of the time series governed by the adjusted minimal LRR is

Table 3.2 Time series FORT120: relative MSD errors of the reconstruction and forecasts

ET	rec (%)	vec12 (%)	rec12 (%)	rec_init12 (%)	vec54 (%)	rec54 (%)	rec_init54 (%)
1	23.11	23.34	23.46	23.49	23.84	23.73	24.02
3	14.79	15.82	16.19	16.41	17.60	17.78	18.17
5	11.63	15.49	15.58	15.44	15.23	15.23	15.57
7	9.70	14.13	15.65	14.41	15.12	24.98	23.26
11	7.45	16.76	17.48	15.59	21.34	23.30	20.57

$$
\begin{aligned}
y_n = {} & C_1 0.997^n + C_2 0.994^n \sin(2\pi n/12 + \varphi_2) + \\
& + C_3 \sin(2\pi n/4 + \varphi_3) + C_4 1.003^n \sin(2\pi n/6 + \varphi_4) + \\
& + C_5 0.998^n \sin(2\pi n/2.4 + \varphi_5) + C_6 0.989^n \sin(2\pi n/3 + \varphi_6).
\end{aligned}
$$

The coefficients C_i and φ_i can be estimated by the least-squares method, see [15, Sect. 3.3] and Fragment 3.5.9 in [24]

The terms are ordered by their eigenvalue shares (in the order of their decreasing). Recall that ordering by roots moduli is generally different from ordering by eigenvalues, since roots moduli are related to the rates of increase/decrease of the time series components and thereby influence a future behavior of the time series governed by the corresponding LRR.

Thus, a preliminary investigation implies that the time series FORT174 and FORT120 well fit to the respective models of the form required, so we can start their forecasting.

Choice of Forecasting Methods and Parameters

Let us demonstrate the approach to forecasting on the 'Fortified wine' example, investigating the accuracy of forecasting the values at the points 121–174 (the test period) on the base of the reconstruction of the points 1–120 (the base period 'FORT120'). The 12-point ahead and 54-point ahead forecasts are considered. Table 3.2 summarizes the errors of forecasts for different forecasting methods. The relative MSD errors of estimation of $\mathbb{Y}$ by $\widetilde{\mathbb{Y}}$ are calculated as

$$\|\widetilde{\mathbb{Y}} - \mathbb{Y}\|_{\mathrm{F}} / \|\mathbb{Y}\|_{\mathrm{F}} \cdot 100\%. \tag{3.19}$$

In Table 3.2, the column 'ET' shows the chosen numbers of the leading eigentriples, the column 'rec' gives the reconstruction errors, the columns 'vec12', 'rec12', 'vec54', 'rec54' correspond to vector and recurrent forecasting for the horizons 12 and 54 terms respectively. The suffix '_init' means that the forecasting formula was applied to the initial series rather than to the reconstructed one.

Let us now discuss the main points of the forecasting logic.

1. Note first that only a set of components separated from the residual may be chosen. For the 'FORT120' series the admissible numbers of components are 1, 3, 5, 7, or 11.
2. There is a conflict between the accuracy of reconstruction and stability of forecasting. In Table 3.2 the errors of reconstruction decrease (the column 'rec') while the errors of forecasts decrease in the beginning and increase later. Note that all considered components are related to the signal and therefore the increase of errors is related to instability.
3. The observed behaviour of the forecasting errors means that the optimal number of the components for forecasting is 7 for 12-term ahead and 5 for 54-term ahead forecasts. This is a natural result since the stability of forecasting is much more important for the long-term forecasting.
4. The vector forecasting method provides more stable forecast of 'FORT120' which becomes apparent for long horizons.
5. Comparison of the forecasting methods can be performed by means of the confidence intervals: smaller size of the confidence intervals indicates better stability of forecasting. This approach does not help for choosing the optimal number of components, since the rough forecast can be the most stable. However, this is a good tool to compare forecasting modifications for a fixed number of components. In particular, the size of the bootstrap confidence interval for ET1–7 is one and half times smaller for the vector forecast than that for the recurrent forecast.
6. Table 3.2 shows that generally the forecasting formula can be applied to the initial time series (the columns with the suffix '_init') instead of the reconstructed one. However, there is no noticeable improvement.
7. The LRR can be adjusted by two ways. The first modification is to remove extraneous roots and to adjust the signal roots using prior information. For the 'FORT120' time series we know the periods of the seasonal component. The modified LRR looks more appropriate as it adequately incorporates the periodicity. However, the forecast is very unstable and gives the forecasting error several times larger than for the min-norm LRR. The reason is that the initial data with error can significantly change the amplitudes of the true harmonics. Certainly, the minimal LRR should be applied to the reconstructed series.
8. A specific feature of this dataset is that the behaviour of the series is close to multiplicative. However, this time series is not purely multiplicative since the form of the seasonal period differ from year to year (Fig. 2.17). The last conclusion is confirmed by different moduli of the roots. For the initial time series the leading harmonic with period 12 is decreasing and the estimated modulus of the corresponding root is equal to 0.994. Therefore, the decreasing exponential has stable behaviour, regardless of the estimation errors. After the log-transformation of a multiplicative series, the root modulus becomes close to 1 and the estimation error can give the modulus of the estimated root larger than 1; that is, the forecast (especially, long-term) could be unstable. The 'FORT120' series demonstrates this effect, since the forecasting error for the log-transformed data is larger than that for the original data.

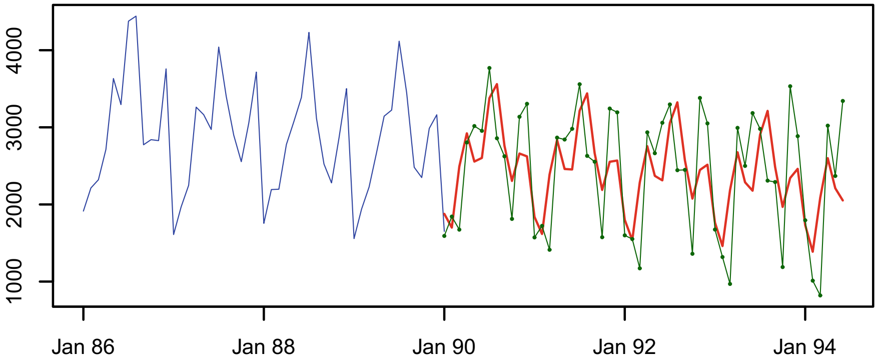


Fig. 3.4 FORT120: two forecasts

Figure 3.4 shows the last 4 years of the series ‘FORT120’ (blue line) and two vector forecasts for 54 points ahead: the stable and accurate forecast based on ET1–5 (red thick line) and the forecast with unstable and less accurate behaviour based on ET1–11 (green line with dots). Note that the accuracy of forecasting for 12-points ahead is approximately the same for both forecasts.

Summarizing we make the following conclusions concerning our experience with forecasting the ‘Fortified wine’ series: (1) it is better to use the original time series rather than its log-transformed version; (2) the best eigentriple group used for forecasting is either ET1–5 for long-term forecast or ET1–7 for short-term forecast; (3) V-forecasting is more accurate that R-forecasting.

The conclusion about the accuracy of the forecasts is made on the base of comparison of the forecasts with series values in the forecasted points. Stability of the forecasts can be checked by means of the confidence intervals and does not require the knowledge of the series values. For two considered forecasts, the size of confidence intervals for ET1–11 is more than twice larger than that for ET1–5, if we take the last forecasted year. Thus, this example demonstrates that the more accurate long-term forecast corresponds to the more stable one.

3.7 Imputation of Missing Values

This section is devoted to the extension of SSA forecasting algorithms for the analysis of time series with missing data.

The following three approaches for solving this problem are known. The first approach was suggested in [42]. This approach is suitable for stationary time series only and uses the following simple idea: in the process of the calculation of inner products of vectors with missing components, we use only pairs of valid vector components and omit the others.

The second '*Iterative*' approach uses an iterative interpolation. Initially, the places of missing values are filled with arbitrary numbers. Then these numbers are iteratively refined by the successive application of SSA. After each iteration, the values at the places of missing values are taken from the previous iteration but the other values are taken from the initial time series. This approach can be formally applied for almost any location of missing values. Therefore, several artificial gaps can be added and then be used to justify the choice of SSA parameters, namely, the window length and the number of chosen components. This idea was suggested in [4] for the imputation of missing values in matrices and then was extended to time series in [30]. The iterative approach has a semi-empirical reasoning for convergence. However, even for noiseless signals the gaps cannot be filled in one iteration. Therefore, this method has large computational cost. Also, it does not provide exact imputation and it needs an additional information about the subspace dimension.

The third approach of filling in missing data is an extension of SSA forecasting algorithms. This approach is considered below in this section and is called '*the subspace approach*'. According to this approach we continue the structure of the extracted component to the gaps caused by the missing data [16]. The theory of SSA assumes that the forecasted component is (or is approximated by) a time series of finite rank. Theoretical results concerning the exact reconstruction of missing values are also based on this assumption. Nevertheless, the constructed algorithms are applicable to real-life time series with missing values where they give approximate results.

Note that in a particular case, when the missing values are located at the end of the series, the problem of their filling in coincides with the problem of forecasting.

The subspace-based methods of gap filling are partly implemented in the RSSA package, see [24, Sect. 3.3.3], along with the iterative filling-in algorithm.

The Layout of the Algorithm

The general structure of the algorithm for the analysis of time series with missing data is the same as for Basic SSA, but the steps are somewhat different.

Assume that we have the initial time series $\mathbb{X}_N = (x_1, \ldots, x_N)$ consisting of N elements, with a part of $\mathbb{X}_N$ unknown. Let us describe the algorithm, using the notation of Sect. 3.1.4, in the case of reconstruction of the first component $\mathbb{X}_N^{(1)}$ of the observed series $\mathbb{X}_N = \mathbb{X}_N^{(1)} + \mathbb{X}_N^{(2)}$.

First Stage: Decomposition

Step 1. Embedding. Let us fix the window length L, $1 < L < N$. The embedding procedure transforms the initial time series into the sequence of L-dimensional lagged vectors $\{X_i\}_{i=1}^{K}$, where $K = N - L + 1$. Some of the lagged vectors may be incomplete, i.e., contain missing components. Let $\mathcal{C}$ be the set of indices such that the lagged vectors X_i with $i \in \mathcal{C}$ are complete. Let us collect all complete lagged vectors X_i, $i \in \mathcal{C}$, into the matrix $\widetilde{\mathbf{X}}$. Assume that this matrix is non-empty. If there are no missing values, then the matrix $\widetilde{\mathbf{X}}$ coincides with the trajectory matrix of the series $\mathbb{X}_N$. Note that the construction of $\widetilde{\mathbf{X}}$ is the same as in Shaped SSA, see Sect. 2.6.3 and [24, Sect. 2.6].

Step 2. Finding the basis. Let $\widetilde{\mathbf{S}} = \widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^{\mathrm{T}}$. Denote by $\lambda_1 \geq \ldots \geq \lambda_L \geq 0$ the ordered eigenvalues of the matrix $\widetilde{\mathbf{S}}$ and by $U_1, \ldots, U_L$ the orthonormal system of the eigenvectors of the matrix $\widetilde{\mathbf{S}}$ corresponding to these eigenvalues, $d = \max\{i : \lambda_i > 0\}$.

Second Stage: Reconstruction

Step 3a. Choosing the subspace and projection of the complete lagged vectors. Let a set of indices $I_r = \{i_1, \ldots, i_r\} \subset \{1, \ldots, d\}$ be chosen and the subspace $\mathcal{M}_r = \mathrm{span}(U_{i_1}, \ldots, U_{i_r})$ be formed. The choice of the eigenvectors (i.e., their indices) corresponding to $\mathbb{X}_N^{(1)}$ is the same as in Basic SSA. The complete lagged vectors can be projected onto the subspace $\mathcal{M}_r$ in the usual way:

$$\widehat{X}_i = \sum_{k \in I_r} (X_i, U_k) U_k, \quad i \in \mathcal{C}.$$

Step 3b. Projection of the incomplete lagged vectors. For each $\mathcal{Q}$-incomplete lagged vector with missing components in the positions from the set $\mathcal{Q}$, we perform this step which consists of two parts:

(α) calculation of $\widehat{X}_i\big|_{J_L \setminus \mathcal{Q}}, \quad i \notin \mathcal{C}$,

(β) calculation of $\widehat{X}_i\big|_{\mathcal{Q}}, \quad i \notin \mathcal{C}$.

Since adjacent lagged vectors have common information (the trajectory matrix (2.1) consisting of the lagged vectors is Hankel) there are many possible ways of solving the formulated problems. Some of these ways will be discussed in the following sections. The available information also enables processing of ‘empty’ vectors with $\mathcal{Q} = J_L = \{1, \ldots, L\}$. Note that step 3b may change the vectors $\widehat{X}_i$, $i \in \mathcal{C}$. The result of steps 3a and 3b is the matrix $\widehat{\mathbf{X}} = [\widehat{X}_1 : \ldots : \widehat{X}_K]$, which serves as an approximation to the trajectory matrix of the series $\mathbb{X}_N^{(1)}$, under the proper choice of the set I_r.

Step 4. Diagonal averaging. The matrix $\widehat{\mathbf{X}}$ is transformed into the new series $\widetilde{\mathbb{X}}_N^{(1)}$ (the reconstructed time series) by means of the diagonal averaging.

Clusters of Missing Data

Implementation of step 3b for projecting the incomplete vectors needs a definition of clusters of missing data and their classification assuming that L is fixed.

A sequence of missing data of a time series is called a *cluster of missing data* if every two adjacent missing values from this sequence are separated by less than L non-missing values and there is no missing data among L neighbours (if they exist) on the left/right element of the cluster. Thus, a group of at least L successive non-missing values of the series separates clusters of missing data. A cluster is called *left/right* if its left/right element is located at a distance of less than L from the left/right end of the series. A cluster is called *continuous* if it consists of successive missing data.

Step 3b can be performed independently for each cluster of missing data.

Methods for Step 3b

Different realizations of Step 3b are thoroughly considered in [16]. Here we briefly describe several typical versions and their relation to SSA forecasting methods formulated in Sect. 3.1. Let the window length L and the indices of the eigentriples corresponding to the chosen time series component be fixed. Propositions 3.2 and 3.3 (where we take $n = L$, $m = r$, $I_r = \{i_1, \ldots, i_r\}$, $\mathbf{P} = [U_{i_1} : \ldots : U_{i_r}]$) provide the theoretical ground for the methods of filling in.

If the considered cluster is continuous and is not left, then (3.9) with $\mathcal{Q} = \{L\}$ provides the coefficients of an LRR that can be applied to the reconstructed points that lie on the left from the missing data cluster (*sequential filling in from the left*). Similarly, setting $\mathcal{Q} = \{1\}$ and applying the backward recurrence relation (3.9) to the reconstructed data taken from the right side, *sequential filling in from the right* can be introduced. Different combinations of the sequential fillings in from the left and from the right (the so-called two-sided methods) can be constructed. For example, their average can be used.

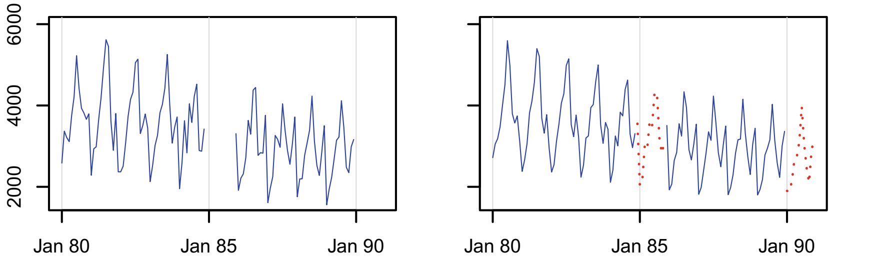
Remark 3.7 Consider a continuous cluster of missing data of length M, which is a right extreme cluster (and assume that there are no other clusters of missing data in the series). If the sequential method described above is applied to this cluster, then the result will coincide with the recurrent forecast for M terms ahead (Sect. 3.1.2.1), where the forecast is constructed on the first $N - M$ points of the time series and the same parameters L and I_r.

In the same manner as we have used for the vector forecasting (Sect. 3.1.2.2), the vector coordinates at the positions of non-missing components can be filled with the help of the adjacent complete vectors and then projected to $\mathcal{M}|_{J_L \setminus \mathcal{Q}}$ by the projector given by formula (3.11) (‘$\boldsymbol{\Pi}$ *Projector*’).

Also, in the same manner as the simultaneous forecasting was introduced (see Sect. 3.1.2.3), the vector coordinates at the positions of missing components can be filled in simultaneously, not one by one as in the sequential filling in, since Proposition 3.2 allows filling in several vector coordinates at once (‘*simultaneous filling in*’). This may simplify the imputation of not-continuous clusters of missing data.

Discussion

- As well as for forecasting, the approach above allows filling in missing values in any component of the time series, not necessary in the signal. For example, missing values in the trend can be filled in. Certainly, an approximate separability of the imputed component from the residual is required.
- If the time series component is exactly separated from the residual and has finite rank, it can be filled in exactly.
- The location of missing data is very important for the possibility of imputation by the subspace method, since the number of non-missing values should be large enough for achieving separability. At least, the number of the complete lagged vectors should be larger than the rank of the forecasted time series component.
- If there are many randomly located missing data, then it can be impossible to get a sufficient number of lagged vectors. However, it is possible to estimate the subspace by involving the lagged vectors with a few missing entries; see [16] for details.



(a) Initial time series with missing data (b) Reconstructed time series with filled in data

Fig. 3.5 FORT120: Filling in missing data

Example

To demonstrate the work of the methods of filling in missing data, let us consider the time series FORT120, which was investigated for forecasting in Sect. 3.6.

Let us remove 12 known values, starting with 60th point (i.e., we assume that the values for a year since December 1984 are unknown). For this artificially missing data we estimate the accuracy of their recovery for different versions of the algorithm. Also, to simulate forecast, we add 12 missing data after the last, 120th point of the series. The time series obtained is illustrated in Fig. 3.5a.

The first question is how to choose the window length L. In the case of no missing data, the general recommendation is to choose the window length close to $N/2$ and divisible by the period of expected periodicity (here, it is 12 months). The window length $L = 60$ meets these conditions. However, for $L = 60$ all lagged vectors will contain missing data. Hence we have to choose smaller L. The choice of $L = 36$ provides us with 38 complete lagged vectors with no missing data.

The analysis of the time series FORT120 in Sect. 3.6 shows that the eigenvectors with indices 1–7 provide the best forecast for 12 points ahead for the choice $L = 60$, while the whole signal is described by the 11 leading eigentriples. The structure of the eigentriples for $L = 36$ is similar and we can use the interpretation of the leading eigentriples found in Sect. 3.6.

The comparison of the filling in results with the values, which were artificially removed from the initial time series, shows an advantage of the version using 'Π Projector' with simultaneous filling in of the missing data and the choice $r = 11$, $I_r = \{1, 2, \ldots, 11\}$. This differs from the forecasting results obtained in Sect. 3.6. Note that the used method of filling in missing data at the end of the time series was not considered during forecasting. Therefore, the ideas of missing data imputation can extend the number of forecasting methods. However, since the precision of reconstruction influences forecasting accuracy much more than the quality of missing data imputation, when the missing data is somewhere in the middle of the series, some methods developed for missing value imputation can be inappropriate as methods of forecasting; an obvious example is the iterative approach.

The result of missing data imputation is illustrated in Fig. 3.5b. The reconstructed series is marked by the dotted line in the area of missing data. The relative MSD

Table 3.3 FORT120: MSD errors for Iterative and Subspace methods of filling in

Method	Middle	End	Total
Subspace $L = 36$	255.9	292.8	275.0
Iterative $L = 36$	221.2	333.0	282.7
Iterative $L = 60$	216.2	419.3	333.6

error (3.19) of reconstruction is approximately equal to 9% for the missing data and to 6% for the non-missing terms in the series.

Comparison with the iterative method. Let us apply the iterative method to the same FORT120 data with the same missing entries, at the middle of the series and at the end. If we replace the missing data by the average value of all valid series points, then 20 iterations are sufficient for convergence. The results are presented in Table 3.3. The errors are calculated as square root of the average squared deviations. For the missing values in the middle, the iterative method provides slightly smaller errors of reconstruction than the subspace method, while for the end points (that is, for forecasting) the iterative method is not stable with respect to the window length. Note that the choice $L = 60$ is not appropriate for the subspace method.

Simulations performed for noisy model series of finite rank in the form of a sum of several products of exponential and harmonic series confirm that the error of filling in missing data at the middle are similar for both methods, while the subspace method is more stable for forecasting.

3.8 Subspace-Based Methods and Estimation of Signal Parameters

While the problems of reconstruction and forecasting are traditionally included into the scope of problems solved by SSA, estimation of signal parameters is usually not. In contrast, estimation of signal parameters is the primary objective for many subspace-based methods of signal processing. In this section we follow [14] to describe the most common subspace-based methods and demonstrate their compatibility with SSA. For simplicity of notation we always assume $L \le K = N - L + 1$.

Let us shortly describe the problem. Consider a signal $\mathbb{S}_N = (s_1, \ldots, s_N)$ in the form $s_n = \sum_{j=1}^{r} c_j \mu_j^n$, $n = 1, \ldots, N$, where all μ_j are assumed to be different; the more complicated form (3.13) can be considered in a similar manner. The problem is to estimate μ_j observing the noisy signal. The $\mu_j = \rho_j e^{\mathrm{i}2\pi\omega_j}$ are expressed in terms of parameters ρ_j and ω_j, which can often be interpreted. In particular, ω_j are the frequencies presented in the signal. An estimator of μ_j provides the information about the structure of the signal, which is different from the information we get from the coefficients c_j. Note that if the time series is real-valued, then s_n can be written as the sum of modulated sinusoids $A_j \rho_j^n \cos(2\pi\omega_j n + \varphi_j)$.

The idea of subspace-based methods is as follows. Let $r < N/2$. The signal $\mathbb{S}_N$ with $s_n = \sum_{j=1}^{r} c_j \mu_j^n$ has rank r and is governed by LRRs like $s_n = \sum_{k=1}^{t} a_k s_{n-k}$, $t \geq r$. Then μ_j can be found as the signal roots of the characteristic polynomial of a governing LRR (see Sect. 3.2). Simultaneously, the L-trajectory space ($L > r$) of the signal (the so-called signal subspace) has dimension r and is spanned by the vectors $(1, \mu_j, \ldots, \mu_j^{L-1})^{\mathrm{T}}$. The coefficients of the governing LRRs of order $L - 1$ can also be found using the information about the signal subspace. Methods of estimating μ_j based on estimation of the signal subspace are called *subspace-based methods*.

Since finding signal roots of the characteristic polynomial of the LRR governing the signal is very important for estimation of the signal parameters, we start with several facts relating signal roots to eigenvalues of some matrix.

3.8.1 Basic Facts

The next statement follows from the properties of eigenvalues.

Proposition 3.5 *Roots of a polynomial* $p(\mu) = \mu^M + c_1\mu^{M-1} + \ldots + c_{M-1}\mu + c_M$ *coincide with eigenvalues of its companion matrix* $\mathbf{C}$ *defined by*

$$\mathbf{C} = \begin{pmatrix} 0 & 0 & \ldots & 0 & -c_M \\ 1 & 0 & \ldots & 0 & -c_{M-1} \\ 0 & 1 & \ldots & 0 & -c_{M-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & -c_1 \end{pmatrix}.$$

Note that the multiplicities of the roots of the polynomial $p(\mu)$ are equal to the algebraic multiplicities of the eigenvalues of its companion matrix (i.e., to the multiplicities of the roots of the characteristic polynomial of this matrix). However, these multiplicities do not always coincide with the geometric multiplicities, which are equal to the dimensions of the eigenspaces corresponding to the eigenvalues.

To derive an analytic form of the signal ($s_n = \sum_{j=1}^{t} c_j \mu_j^n$ or (3.13) in the general case), we need to find roots of the characteristic polynomial of the LRR governing the signal. By Proposition 3.5, we have to find either the roots of the characteristic polynomial or the eigenvalues of its companion matrix. The latter does not require the full knowledge of the LRR. Let us demonstrate that for finding the signal roots it is sufficient to know the basis of the signal trajectory space.

Let $\mathbf{C}$ be a full-rank $d \times d$ matrix, $Z \in \mathsf{R}^d$, and $\mathbf{Z}$ be a full-rank $L \times d$ matrix ($L > d$), which can be expressed as

$$\mathbf{Z} = \begin{pmatrix} Z^{\mathrm{T}} \\ Z^{\mathrm{T}}\mathbf{C} \\ \vdots \\ Z^{\mathrm{T}}\mathbf{C}^{L-1} \end{pmatrix}. \tag{3.20}$$

Let us again denote the matrix $\mathbf{Z}$ without the last row by $\underline{\mathbf{Z}}$ and the matrix $\mathbf{Z}$ without its first row by $\overline{\mathbf{Z}}$. It is clear that $\overline{\mathbf{Z}} = \underline{\mathbf{Z}}\mathbf{C}$. We call this property of $\mathbf{Z}$ the *shift property* generated by the matrix $\mathbf{C}$.

Proposition 3.6 *Let* $\mathbf{Z}$ *satisfy the shift property generated by the matrix* $\mathbf{C}$*,* $\mathbf{P}$ *be a full-rank* $d \times d$ *matrix, and* $\mathbf{Y} = \mathbf{Z}\mathbf{P}$*. Then the matrix* $\mathbf{Y}$ *satisfies the shift property generated by the matrix* $\mathbf{D} = \mathbf{P}^{-1}\mathbf{C}\mathbf{P}$*, i.e.,* $\overline{\mathbf{Y}} = \underline{\mathbf{Y}}\mathbf{D}$.

The proof of this proposition is straightforward.

Note that the multiplication by a nonsingular matrix $\mathbf{P}$ can be considered as a transformation of the vector coordinates in the column space of the matrix $\mathbf{Z}$. It is easily seen that the matrices $\mathbf{C}$ and $\mathbf{D} = \mathbf{P}^{-1}\mathbf{C}\mathbf{P}$ have the same eigenvalues; these matrices are called *similar*.

Remark 3.8 Let the matrix $\mathbf{Y}$ satisfy the shift property generated by the matrix $\mathbf{D}$. Then $\mathbf{D} = \underline{\mathbf{Y}}^{\dagger}\overline{\mathbf{Y}}$, where $\mathbf{A}^{\dagger}$ denotes the Moore–Penrose pseudoinverse of $\mathbf{A}$.

Proposition 3.7 *Let a time series* $\mathbb{S}_N = (s_1, \ldots, s_N)$ *satisfy the minimal LRR (3.12) of order* d*,* $L > d$ *be the window length,* $\mathbf{C}$ *be the companion matrix of the characteristic polynomial of this LRR. Then any* $L \times d$ *matrix* $\mathbf{Y}$ *with columns forming a basis of the trajectory space of* $\mathbb{S}_N$ *satisfies the shift property generated by some matrix* $\mathbf{D}$*. Moreover, the eigenvalues of this shift matrix* $\mathbf{D}$ *coincide with the eigenvalues of the companion matrix* $\mathbf{C}$ *and hence with the roots of the characteristic polynomial of the LRR.*

Proof Note that for any $1 \le i \le N - d$ we have

$$(s_i, s_{i+1}, \ldots, s_{i+(d-1)})\mathbf{C} = (s_{i+1}, s_{i+2}, \ldots, s_{i+d}).$$

Therefore, (3.20) holds for $Z = (x_1, x_2, \ldots, s_d)^{\mathrm{T}}$. It can be easily proved that for a time series governed by the minimal LRR of order d, any d adjacent columns of the trajectory matrix are linearly independent. Consequently, the matrix $\mathbf{Z} = [S_1 : \ldots : S_d]$ is of full rank and we can apply Proposition 3.6. □

Remark 3.9 The SVD of the L-trajectory matrix of a time series provides a basis of its trajectory space. Specifically, the left singular vectors which correspond to the nonzero singular values form such a basis. If we observe a time series of the form 'signal + residual', then the SVD of its L-trajectory matrix provides the basis of the signal subspace under the condition of exact strong separability of the signal and the residual.

3.8.2 ESPRIT

Consider a time series $\mathbb{X}_N = \{x_i\}_{i=1}^N$ with $x_i = s_i + p_i$, where $\mathbb{S}_N = \{s_i\}_{i=1}^N$ is a time series governed by an LRR of order r (that is, signal) and $\mathbb{P}_N = \{p_i\}_{i=1}^N$ is a residual (noise, perturbation). Let $\mathbf{X}$ be the trajectory matrix of $\mathbb{X}_N$. In the case of exact or approximate separability of the signal and the residual, there is a set I of eigenvector indices in (2.2), which correspond to the signal. If the signal dominates, then $I = \{1, \ldots, r\}$ and the subspace $\mathfrak{L}_r = \mathrm{span}\{U_1, \ldots, U_r\}$ can be considered as an estimate of the true signal subspace $\mathbb{S}$. Therefore, we can use $\widetilde{\mathbf{Y}} = \mathbf{U}_r = [U_1 : \ldots : U_r]$ as an estimate of $\mathbf{Y}$ from Proposition 3.7. Then the shift property is approximately met and $\underline{\mathbf{U}}_r\mathbf{D} \approx \overline{\mathbf{U}}_r$.

The method ESPRIT consists in estimation of the signal roots as the eigenvalues of a matrix $\widehat{\mathbf{D}}$, for which

$$\underline{\mathbf{U}}_r\widehat{\mathbf{D}} \approx \overline{\mathbf{U}}_r. \tag{3.21}$$

By estimating the signal roots, ESPRIT provides estimates of the signal parameters. See how to call the corresponding R functions from the RSSA package in [24, Sect. 3.1.3]

Let us study the methods of finding the matrix $\widehat{\mathbf{D}}$. The main idea of LS-ESPRIT was introduced in the paper [33] devoted to the problem of estimating frequencies in a sum of sinusoids, in the presence of noise. The method was given the name ESPRIT in [40]; this name was later used in many other papers devoted to the DOA (Direction of Arrival) problem. For time series processing, LS-ESPRIT is also called Hankel SVD (HSVD, [3]). Later the so-called TLS-ESPRIT modification was suggested (see e.g. [48], where the method was called Hankel Total Least Squares (HTLS)). There are papers devoted to the perturbation study of ESPRIT, see e.g. [2], where specific features of ESPRIT in the case of multiple roots are also described.

Remark 3.10 ESPRIT is able to estimate parameters of a separable time series component, not necessary the signal, if the matrix $\mathbf{U}_r$ consists of the corresponding eigenvectors.

Least Squares (LS-ESPRIT). The LS-ESPRIT estimate of the matrix $\mathbf{D}$ is

$$\widehat{\mathbf{D}} = \underline{\mathbf{U}}_r^{\dagger}\overline{\mathbf{U}}_r = (\underline{\mathbf{U}}_r^{\mathrm{T}}\underline{\mathbf{U}}_r)^{-1}\underline{\mathbf{U}}_r^{\mathrm{T}}\overline{\mathbf{U}}_r. \tag{3.22}$$

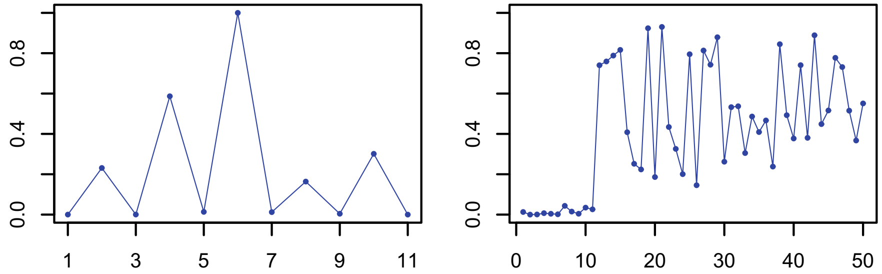
The eigenvalues of $\widehat{\mathbf{D}}$ do not depend on the choice of the basis of the subspace $\mathfrak{L}_r = \mathrm{span}\{U_1, \ldots, U_r\}$.

Total Least Squares (TLS-ESPRIT). As $\mathbf{U}_r$ is known only approximately then there are errors in both $\underline{\mathbf{U}}_r$ and $\overline{\mathbf{U}}_r$. Therefore, the solution of the approximate equality $\underline{\mathbf{U}}_r\mathbf{D} \approx \overline{\mathbf{U}}_r$, based on the method of Total Least Squares (TLS), can be more accurate.

Recall that to solve the equation $\mathbf{AX} \approx \mathbf{B}$, TLS minimizes the sum

$$\|\widetilde{\mathbf{A}} - \mathbf{A}\|_{\mathrm{F}}^2 + \|\widetilde{\mathbf{B}} - \mathbf{B}\|_{\mathrm{F}}^2 \longrightarrow \min \tag{3.23}$$

with respect to $\widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{B}}$ such that $\exists \mathbf{Z} : \ \widetilde{\mathbf{A}}\mathbf{Z} = \widetilde{\mathbf{B}}$.



(a) ‘Eggs’ (b) ‘White dwarf’

Fig. 3.6 Rank and separability detection by ESTER

Set $\mathbf{A} = \underline{\mathbf{U}}_r$, $\mathbf{B} = \overline{\mathbf{U}}_r$ in (3.23). Then the matrix $\mathbf{Z}$ that minimizes (3.23) is called the TLS-estimate of $\mathbf{D}$ (see [26] for explicit formulas). The TLS-ESPRIT estimate is the same for any orthonormal basis in $\mathfrak{L}_r$, see [14]. This is not true if the basis in $\mathfrak{L}_r$ is not orthogonal.

ESPRIT and rank estimation. ESPRIT deals with the matrix equation (3.21), which has a solution if the r leading components are exactly separated from the residual (the remaining $L - r$ components). Therefore, some measure of difference between the left-hand and the right-hand sides of the matrix equation can indicate the cut-off points of separability; that is, it can suggest the number of the leading SVD components that are separated from the residual. Therefore, the last cut-off point of separability corresponds to the rank estimation. In [1], the L_2-norm of the difference, we denote it $\rho_2(r)$, is used for estimating the rank of the signal (the ESTER method) provided that there are no separability points within the signal components. An attractive feature of the ESTER-type methods is that they assume only separability of the signal from noise and do not assume parametric forms for the signal and noise.

However, the ESTER-type estimates of rank appear to be unstable for real-world time series. Figure 3.6a shows how the ESTER reflects the points of separability: small values of $\rho_2(r)$ correspond to the cut-off points of separability (compare with Fig. 2.24a). In Fig. 3.6b, where the rank of the signal is estimated to be 11 (see Fig. 2.26), the behavior of $\rho_2(r)$ demonstrates that there are no small values of $\rho_2(r)$ for $r \le 11$.

3.8.3 Overview of Other Subspace-Based Methods

In this subsection, we demonstrate ideas of other subspace-based methods which are different from SSA and ESPRIT-like methods. These methods are applied to time series governed by LRRs and estimate the main (signal) roots of the corresponding characteristic polynomials. The most fundamental subspace-based methods were developed for the cases of a noisy sum of imaginary exponentials (cisoids) and of real sinusoids, for the purpose of the estimating their frequencies, see e.g. [45]. We are mostly interested in the methods that can be applied to any time series of finite rank given in the form (3.13). Below we write down familiar methods in a unified notation; references can be found in [45].

We start with a description of general methods in the complex-valued case. Most of these general methods use the correspondence between LRRs that govern the signal and vectors from the subspace orthogonal to the signal subspace, which was introduced in Sect. 3.2.2.

The first idea is to use the properties of signal and extraneous roots to distinguish between them. Let us introduce three possible realizations of this idea. As before, $\mathcal{S}$ is the signal subspace and $\mathcal{L}_r$ is its estimate.

Version 1. Consider an LRR that governs the signal (the best choice is the min-norm LRR, see Sect. 3.2.2; however, this is not essential). Then find all the roots μ_m of the characteristic polynomial of this LRR and then find coefficients c_{mj} in (3.13). The coefficients c_{mj} corresponding to the extraneous roots are equal to 0. In the case of a noisy signal, $\widehat{\mu}_m$ are the roots of a polynomial with coefficients taken from a vector that belongs to $\mathcal{L}_r^\perp$, and the extraneous roots have small absolute values of the LS estimates $\widehat{c}_{mj}$.

Version 2. Let us consider the forward and backward min-norm predictions. It is known that the corresponding characteristic polynomials have the conjugate extraneous roots and their signal roots are reciprocal, that is, connected by the relation $z' = z^{-1}$, see [5, Proposition 2]. Note that the forward prediction given by a vector $A \in \mathcal{S}^\perp$ corresponds to the roots of $\langle Z(z), A\rangle = 0$, where $Z(z) = (1, z, \ldots, z^{L-1})^\mathrm{T}$ and $\langle\cdot, \cdot\rangle$ is the inner product in the complex Euclidean space. At the same time, the backward prediction given by a vector $B \in \mathcal{S}^\perp$ corresponds to the roots of $\langle Z(1/z), B\rangle = 0$. If we consider the roots of the forward and backward min-norm polynomials together, then all extraneous roots lie inside the unit circle, while one of z' and z is located on or outside the unit circle. This allows us to detect the signal roots. For a noisy signal, A and B are specific vectors taken from $\mathcal{L}_r^\perp$: these vectors are the projections onto $\mathcal{L}_r^\perp$ of the unit vectors $\mathbf{e}_L$ and $\mathbf{e}_1$, correspondingly. If all coefficients of a polynomial are real, then the set of roots coincides with the set of their complex conjugates due to properties of roots of polynomials with real coefficients. However, if complex-valued time series and complex coefficients are considered, then it is convenient to consider the backward min-norm polynomials with conjugate coefficients; the corresponding LRR governs the reverse series of complex conjugates to the original time series.

Version 3. Let us take a set of vectors from $\mathcal{S}^\perp$. Each vector from $\mathcal{S}^\perp$ with nonzero last coordinate generates an LRR. The signal roots of the characteristic polynomials of these LRRs are equal, whereas the extraneous roots are arbitrary. For a noisy signal, the set of vectors is taken from $\mathcal{L}_r^\perp$. Then the signal roots correspond to clusters of roots if we consider pooled roots.

Several more methods are developed for estimation of frequencies in a noisy sum of undamped sinusoids or imaginary exponentials. Let for simplicity $s_n = \sum_{k=1}^{r} c_k e^{\mathrm{i}2\pi\omega_k n}$. In this case, the signal roots $e^{\mathrm{i}2\pi\omega_k}$ all have the absolute value 1 and can be parameterized by one parameter (frequency) only. Let $W = W(\omega) =$

$Z(e^{i2\pi\omega})$. As $W(\omega_k) \in \mathcal{S}$, $\langle W(\omega_k), A\rangle = 0$ for all $A \in \mathcal{S}^{\perp}$. Therefore, if $A \in \mathcal{S}^{\perp}$, then we can consider the square of the cosine of the angle between $W(\omega)$ and A as a measure of their orthogonality. This idea forms the basis for the Min-Norm and MUSIC methods. The modifications of the methods in which the roots are ordered by the absolute value of the deviation of their moduli from the unit circle have names with the prefix 'root-'.

Version 4. Min-Norm. Let $f(\omega) = \cos^2(\widehat{W(\omega), A})$, where A, the projection of $\mathbf{e}_L$ onto $\mathcal{L}_r^{\perp}$, is the vector corresponding to the min-norm forward prediction. The Min-Norm method is based on searching for the maximums of $1/f(\omega)$; this function can be interpreted as a pseudospectrum.

Version 5. MUSIC. Let $f(\omega) = \cos^2(\widehat{W(\omega), \mathcal{L}_r^{\perp}})$. If we take eigenvectors U_j, $j = r+1, \ldots, L$, as a basis of $\mathcal{L}_r^{\perp}$, $\mathbf{U}_{r+1,L} = [U_{r+1} : \ldots : U_L]$, then $\mathbf{U}_{r+1,L}\mathbf{U}^*_{r+1,L}$, where $*$ denotes conjugate transposition, provides the matrix of projection on $\mathcal{L}_r^{\perp}$ and therefore $f(\omega) = W^*(\mathbf{U}_{r+1,L}\mathbf{U}^*_{r+1,L})W/\|W\|^2 = \sum_{j=r+1}^{L} f_j(\omega)$, where $f_j(\omega) = \cos^2(\widehat{W(\omega), U_j})$. Thus, the MUSIC method can be considered from the viewpoint of the subspace properties and does not require the computation of the roots of the characteristic polynomials. Similar to the Min-Norm method, the MUSIC method is essentially the method of searching for the maximums of the pseudospectrum $1/f(\omega)$.

3.8.4 *Hankel SLRA*

3.8.4.1 Cadzow Iterations

The aim of Cadzow iterations is to extract the finite-rank signal $\mathbb{S}$ of rank r from an observed noisy signal $\mathbb{X} = \mathbb{S} + \mathbb{P}$. Cadzow iterations [8] were suggested as a method of signal processing, without any relation to SSA method. However, these two methods are very much related.

Cadzow iterations present an example of the procedure called alternating projections. A short form of M iterations is

$$\widetilde{\mathbb{S}} = \mathcal{T}^{-1} \circ \left(\boldsymbol{\Pi}_{\mathcal{H}} \circ \boldsymbol{\Pi}_r\right)^M \circ \mathcal{T}(\mathbb{X}). \tag{3.24}$$

Here the embedding operator $\mathcal{T}$ provides a one-to-one correspondence between time series and trajectory matrices for the fixed window length L, $\boldsymbol{\Pi}_r$ is the projection of a matrix to the space $\mathcal{M}_r$ of $L \times K$ matrices of rank not larger than r, the hankelisation operator $\boldsymbol{\Pi}_{\mathcal{H}}$ is also the projection into the space $\mathcal{H}$ of Hankel matrices in the Frobenius norm. See how to call the corresponding R functions from the RSSA package in [24, Sect. 3.4.3].

The Basic SSA with fixed L and fixed grouping $I = \{1, 2, \ldots, r\}$ is simply the first iteration of Cadzow iterations, see (2.22). This means that Cadzow iterations can be defined as a repeated application of Basic SSA with fixed L and fixed grouping $I = \{1, 2, \ldots, r\}$ to the series $\widetilde{\mathbb{X}}_I$, see (2.7), obtained by Basic SSA in the previous step; the initial Cadzow iteration is Basic SSA applied to the original series $\mathbb{X}$.

The result of Cadzow iterations is a signal of finite rank $\le r$. However, this does not guarantee that the limiting result is closer to the true signal than SSA result (that is, just one iteration). Among other factors, this depends on how well the true signal can be approximated by the series of rank r and the recommended choice of L: indeed, a usual recommendation in signal processing literature is to choose L which is just slightly larger than r; this, however, is unwise from the viewpoint of SSA.

Cadzow iterations are easily extended to the multidimensional case. Cadzow iterations can also be generalized to weighted iterations in a natural way, when the projectors in (3.24) are constructed with respect to a given weighted norm.

There are weights, when the implementation of the weighted projections has computational cost comparable with that in the unweighted case [50]: the inner product in $\mathsf{R}^{L\times K}$ is defined as $\langle \mathbf{X}, \mathbf{Y}\rangle = \langle \mathbf{X}, \mathbf{Y}\rangle_{\mathbf{L},\mathbf{R}} = \mathrm{tr}(\mathbf{LXRY}^{\mathrm{T}})$, where $\mathbf{L} \in \mathsf{R}^{L\times L}$, $\mathbf{R} \in \mathsf{R}^{K\times K}$; $\|\mathbf{X}\|_{\mathbf{L},\mathbf{R}}$ is the corresponding matrix norm in $\mathsf{R}^{L\times K}$. Below in this section we will explain the problem of choice of the weights for more accurate estimates.

Formally, Cadzow iterations constitute a method of solving the general HSLRA (Hankel-structured matrix low-rank approximation) problem considered next. The series $\widetilde{\mathbb{S}}_N$ obtained by (3.24) can be regarded as an estimator of the signal. Note, however, that Cadzow iterations may not converge even to a local optimum of the respective optimization problem (3.26).

3.8.4.2 General HSLRA Problem

We will formulate the HSLRA problem as the problem of extraction of a signal $\mathbb{S} = (s_1, s_2, \ldots, s_N)^{\mathrm{T}}$ of rank r from an observed noisy signal $\mathbb{X} = (x_1, x_2, \ldots, x_N)^{\mathrm{T}} = \mathbb{S} + \mathcal{E}$ of length N, where $\mathcal{E} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_N)^{\mathrm{T}}$ is a vector of (unobserved) random noise with zero mean and covariance matrix $\mathbf{\Sigma} = \mathsf{E}\mathcal{E}\mathcal{E}^{\mathrm{T}}$; see also [9, 34, 50].

The HSLRA problem can be stated in two forms: (a) vector form and (b) matrix form. The vector (time series) form is: for given $\mathbb{X} \in \mathsf{R}^N$ and positive integer $r < \lfloor N/2 \rfloor$,

$$\|\mathbb{X} - \mathbb{Y}\|^2_{\mathbf{W}} \to \min_{\mathbb{Y}:\mathrm{rank}\,\mathbb{Y} \le r} \tag{3.25}$$

where the matrix $\mathbf{W} \in \mathsf{R}^{N\times N}$ is positive definite and $\|\mathbb{Z}\|^2_{\mathbf{W}} = \mathbb{Z}\mathbf{W}\mathbb{Z}^{\mathrm{T}}$ for a row-vector $\mathbb{Z}$. The solution of (3.25) can be considered as a weighted least-squares estimate (WLSE) of the signal $\mathbb{S}$. If noise $\mathcal{E}$ is Gaussian with covariance matrix $\mathbf{\Sigma}$, then the WLSE with $\mathbf{W} = \mathbf{\Sigma}^{-1}$ is the maximum likelihood estimate (MLE). If the properties of the noise process are known then the vector form (3.25) is the most natural way of defining the HSLRA problem. However, solving the HSLRA problem in the vector form is extremely difficult, see e.g. [12]. Although the vector form allows fast implementations, these implementations are very complex and need a starting point close to the solution [35, 51].

The matrix form of the HSLRA problem allows one to use simple subspace-based alternating projection methods like Cadzow iterations considered above. It is the following optimization problem:

$$\|\mathbf{X}-\mathbf{Y}\|^2_{\mathbf{L},\mathbf{R}} \to \min_{\mathbf{Y}\in\mathcal{M}_r\cap\mathcal{H}} . \tag{3.26}$$

For reformulating the original HSLRA problem (3.25) in the matrix form (3.26), we have to choose $\mathbf{X}=\mathcal{T}_L(\mathbb{X})$ and $\mathbf{Y}=\mathcal{T}_L(\mathbb{Y})$; the remaining issue is matching the vector norm in (3.25) with the matrix norm in (3.26).

The general case of the correspondence between the vector-norm and matrix-norm formulations (3.25) and (3.26) of the HSLRA problem is established in the following theorem [13, 20]; $*$ means the matrix convolution.

Theorem 3.2 *For any* $\mathbb{Z}\in\mathsf{R}^N$, $\|\mathcal{T}_L(\mathbb{Z})\|_{\mathbf{L},\mathbf{R}}=\|\mathbb{Z}\|_{\mathbf{W}}$ *if and only if*

$$\mathbf{W}=\mathbf{L}*\mathbf{R}. \tag{3.27}$$

In a typical application, when the structure of the noise in the model ‘signal plus noise’ is assumed, the HSLRA problem is formulated in a vector form with a given matrix $\mathbf{W}$. As mentioned above, algorithms of solving the HSLRA problem are much easier if we have the matrix rather than vector form of the HSLRA problem. Therefore, in view of Theorem 3.2, for a given $\mathbf{W}$ we would want to find positive definite matrices $\mathbf{L}$ and $\mathbf{R}$ such that (3.27) holds; that is, we would want to perform a blind deconvolution of the matrix $\mathbf{W}$.

It follows from the results of [49] that in the case when the noise $\mathcal{E}$ is white, and therefore $\mathbf{W}=\mathbf{I}_N$, the matrix $\mathbf{W}$ cannot be blindly deconvoluted under the condition that $\mathbf{L}$ and $\mathbf{R}$ are positive definite matrices. The paper [20] extends the results of [49] to the case of banded matrices corresponding to the case where the noise $\mathcal{E}$ forms an autoregressive process.

Non-existence of symmetric positive definite weight matrices in the matrix form of Hankel SLRA means that there is no matrix form of the Hankel SLRA problem, which corresponds to the weighted least-squares problem in vector form with optimal weights. However, we can reasonably simply find an approximate solution to the matrix equation (3.27), see [50].

3.9 SSA and Filters

As demonstrated in Sect. 2.3, one of SSA’s capabilities is its ability to be a frequency filter. The relation between SSA and filtering was considered in a number of papers, see for example [6, 28]. These results are mostly related to the case where (a) the window length L is small (much less than $N/2$), and (b) Toeplitz SSA is considered and the filter properties are based on the properties of the eigenvectors of Toeplitz matrices (therefore, the time series is assumed to be stationary, see Sect. 2.5.2.2).

In this section we describe the relation between Basic SSA and filtering in a general form and also consider specific filters generated by Basic SSA.

3.9.1 Linear Filters and Their Characteristics

Let $\mathbf{x} = (\ldots, x_{-1}, \overset{\circ}{x}_0, x_1, x_2, \ldots)$ be an infinite sequence and the symbol 'o' over an element denotes its middle location. Finite series $\mathbb{X}_N = (x_1, \ldots, x_N)$ can be formally presented as a infinite sequence $(\ldots, 0, \ldots, \overset{\circ}{0}, x_1, x_2, \ldots, x_N, 0, \ldots)$. Each linear filter Φ can be expressed as $\big(\Phi(\mathbf{x})\big)_j = \sum\limits_{i=-\infty}^{+\infty} h_i x_{j-i}$. The sequence $\mathbf{h}_\Phi = (\ldots, h_{-1}, \overset{\circ}{h}_0, h_1, \ldots)$ is called *the impulse response*. A filter Φ is called FIR-filter (i.e. with Finite Impulse Response) if $\big(\Phi(\mathbf{x})\big)_j = \sum\limits_{i=-r_1}^{r_2} h_i x_{j-i}$. The filter Φ is called *causal* if $\big(\Phi(\mathbf{x})\big)_j = \sum_{i=0}^{r-1} h_i x_{j-i}$.

The following characteristics are standard for filters: $H_\Phi(z) = \sum_i h_i z^{-i}$ is a *transfer function*, $A_\Phi(\omega) = |H_\Phi(e^{\mathrm{i}2\pi\omega})|$ is a *frequency (amplitude) response* and $\varphi_\Phi(\omega) = \mathrm{Arg} H_\Phi(e^{\mathrm{i}2\pi\omega})$ is a *phase response*. The meaning of the amplitude and phase responses follows from: for the sequence $\mathbf{x}$ with $(\mathbf{x})_j = \cos(2\pi\omega j)$ we have $\big(\Phi(\mathbf{x})\big)_j = A_\Phi(\omega)\cos(2\pi\omega j + \varphi_\Phi(\omega))$.

An important filter characteristic reflecting its noise reduction capability is the filter *power* $\mathscr{E}\Phi = \|\mathbf{h}\|^2 = \sum_i h_i^2$. The following proposition is analogous to Proposition 3.4.

Proposition 3.8 *Let* $\mathbf{x} = \mathbf{s} + \varepsilon$, *where* $(\varepsilon)_j$ *are i.i.d,* $\mathsf{E}(\varepsilon)_j = 0$, $\mathsf{D}(\varepsilon)_j = \sigma^2$. *Let* $\Phi: \Phi(\mathbf{s}) = \mathbf{s}$ *and denote* $\widetilde{\mathbf{x}} = \Phi(\mathbf{x})$. *Then* $\mathsf{E}(\widetilde{\mathbf{x}})_j = (\mathbf{s})_j$ *and* $\mathsf{D}(\widetilde{\mathbf{x}})_j = \sigma^2 \cdot \mathscr{E}\Phi$.

Also, there is a relation between the filter power and the frequency response. Define $\Delta_a\Phi = \mathrm{meas}\{\omega \in (-0.5, 0.5] \colon A_\Phi(\omega) \geq a\}$. Parseval's identity has the following form for filters:

$$\mathscr{E}\Phi = \sum_j \mathrm{h}_j^2 = \int_{-0.5}^{0.5} A_\Phi(\omega)^2\, d\omega.$$

Therefore,

$$\Delta_a\Phi \leq \mathscr{E}\Phi / a^2. \tag{3.28}$$

The inequality (3.28) shows how the support of the frequency response (with threshold a) is related to the filter power.

3.9.2 SSA Reconstruction as a Linear Filter

Let us return to Basic SSA. Let L be the window length and $(\sqrt{\lambda}, U, V)$ be one of the eigentriples generated by the SVD of the trajectory matrix of $\mathbb{X}_N$ (see Sect. 2.1.1 for notation and definitions). Since the reconstruction operation in Basic SSA is the linear operation, it can be written in matrix form.

Let $K = N - L + 1$, $L^* = \min(L, K)$. Define the diagonal $N \times N$ matrix

$$\mathbf{D} = \mathrm{diag}(1, 2, 3, \ldots, L^* - 1, L^*, L^*, \ldots, L^*, L^* - 1, \ldots, 2, 1)$$

and the $K \times N$ matrix

$$\mathbf{W} = \begin{pmatrix} u_1 & u_2 & u_3 & \cdots & u_L & 0 & \cdots & 0 & 0 & 0 \\ 0 & u_1 & u_2 & u_3 & \cdots & u_L & 0 & \cdots & 0 & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & \cdots & \ddots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & u_1 & u_2 & u_3 & \cdots & u_L & 0 & \vdots \\ 0 & 0 & \cdots & 0 & u_1 & u_2 & u_3 & \cdots & u_L & 0 \\ 0 & 0 & 0 & \cdots & 0 & u_1 & u_2 & u_3 & \cdots & u_L \end{pmatrix}.$$

Proposition 3.9 *The time series component $\widetilde{\mathbb{X}}_N$ reconstructed by the eigentriple $(\sqrt{\lambda}, U, V)$ has the form*

$$\widetilde{\mathbb{X}}_N^{\mathrm{T}} = \mathbf{D}^{-1}\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbb{X}_N^{\mathrm{T}}.$$

Proof First, note that $\mathbf{W}\mathbb{X}_N^{\mathrm{T}} = \mathbf{X}^{\mathrm{T}}U = \sqrt{\lambda}V \in \mathsf{R}^K$. This yields that the vector $\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbb{X}_N^{\mathrm{T}}$ (of size N) consists of sums along N antidiagonals of the matrix $\sqrt{\lambda}UV^{\mathrm{T}}$, which is an elementary summand of the SVD. Multiplication by $\mathbf{D}^{-1}$ provides the normalization of the sums by the number of summands and therefore by the definition we obtain the elementary reconstructed component. □

Remark 3.11 Let us add the index i to $\widetilde{\mathbb{X}}$ to indicate that it corresponds to the i-th eigenvector $U = U_i$. Then, evidently, the reconstructed series $\widetilde{\mathbb{X}}^{(I)}$ by the set of eigentriples $\{(\sqrt{\lambda_i}, U_i, V_i),\ i \in I\}$ is equal to the sum of the reconstructed elementary series $\widetilde{\mathbb{X}}^{(i)}$. Therefore, the matrix form for $\widetilde{\mathbb{X}}^{(I)}$ immediately follows from (3.9).

Proposition 3.9 and Remark 3.11 allow us to describe the reconstruction produced by Basic SSA as an application of a set of linear filters.

Let $L \le K$. Define the linear filters Θ_L, Θ_{L-1}, ..., Θ_1 and Ψ by their impulse characteristics $\mathbf{h}_{\Theta_L}, \ldots, \mathbf{h}_{\Theta_1}$ and $\mathbf{h}_{\Psi}$:

$$
\begin{aligned}
\mathbf{h}_{\Theta_L} &= (\ldots, 0, \overset{\circ}{u}_L, 0, \ldots),\\
\mathbf{h}_{\Theta_{L-1}} &= (\ldots, 0, u_{L-1}, \overset{\circ}{u}_L, 0, \ldots),\\
&\ \ldots\\
\mathbf{h}_{\Theta_1} &= (\ldots, 0, u_1, \ldots, u_{L-2}, u_{L-1}, \overset{\circ}{u}_L, 0, \ldots);\\
\mathbf{h}_{\Psi} &= \operatorname{rev}\mathbf{h}_{\Theta_1} = (\ldots, 0, \overset{\circ}{u}_L, u_{L-1}, \ldots, u_1, 0, \ldots).
\end{aligned}
$$

Now we can introduce the reconstructing filters Φ_k, $k = 1, \ldots, L$, generated by the vector U:

$$\Phi_k = \Theta_k \circ \Psi/(L-k+1), \tag{3.29}$$

where '$\circ$' stands for the filter composition, which is equivalent to the convolution '$*$' of the filter impulse characteristics.

Proposition 3.10 *For $\mathbb{X}_N = (x_1, x_2, \ldots, x_N)$, the terms of the elementary reconstructed series $\widetilde{\mathbb{X}}_N$ corresponding to the eigentriple $(\sqrt{\lambda}, U, V)$ have the following representation:*

- $\widetilde{x}_s = (\Phi_{s-K+1}(\mathbb{X}_N))_s$ *for $K+1 \le s \le N$;*
- $\widetilde{x}_s = (\Phi_1(\mathbb{X}_N))_s$ *for $L \le s \le K$.*

This result is a direct consequence of Proposition 3.9. The set of filters providing the reconstruction of $\widetilde{x}_s$ for $1 \le s < L$ can be built in a similar way.

Let us examine two special filters: Φ_1, which is used for the reconstruction of the middle points of the time series with numbers $L, \ldots, K$, and Φ_L, which is used for the reconstruction of the last point only. The former is called *the MPF* (Middle Point Filter) and the latter is referred as *the LPF* (Last Point Filter). In next two sections we consider them separately.

3.9.3 Middle Point Filter

As above, we assume $L \le K$. According to Proposition 3.10, the MPF Φ_1 acts only at the L-th to $(N-L+1)$-th points. This leads to a limited use of the MPF in the case $L \sim N/2$. The explicit formula for the MPF filter $\Phi_1^{(i)}$ corresponding to the eigenvector $U_i = (u_1, \ldots, u_L)^{\mathrm{T}}$ has the following form:

$$\widetilde{x}_s = \sum_{j=-(L-1)}^{L-1} \left(\sum_{k=1}^{L-|j|} u_k u_{k+|j|}/L \right) x_{s-j}, \quad L \le s \le K. \tag{3.30}$$

It is clearly seen that the order of the MPF is equal to $2L-1$. Alternative representation of (3.30) is

$$\widetilde{x}_s = \sum_{j=1}^{L}\sum_{l=1}^{L} u_j u_l x_{s+j-l}/L, \;\; L \le s \le K. \tag{3.31}$$

Let us enumerate several properties of the MPF $\Phi_1^{(I)}$.

1. The filter $\Phi_1^{(I)}$ is symmetric. Hence the MPF is a zero-phase filter. In particular, the MPF does not change phases of sinusoids.
2. In a particular case of $I = \{i\}$, applying Jensen's inequality, we obtain that the sum of coefficients of the $\Phi_1^{(i)}$ given in (3.31) is not larger than 1:

$$\sum_{j=1}^{L}\sum_{l=1}^{L} u_j u_l/L = \left(\sum_{j=1}^{L} u_j\right)^2 \Big/ L \le \sum_{j=1}^{L} u_j^2 = 1.$$

3. If the matrix $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ is positive, then the leading eigenvector U_1 is positive too (Perron's theorem) and, therefore, the coefficients of the filter $\Phi_1^{(1)}$ are positive. This is true, for example, in the case of positive time series. If the time series is close to a constant (at the timescale of L), then the coordinates of U_1 will be close one to another and the MPF filter $\Phi_1^{(1)}$ will be close to the so-called triangle (Bartlett) filter. This implies, for instance, that the extraction of trend by the first run of Sequential SSA with small L (see Sect. 2.5.4) is similar to the application of a weighted moving average procedure with positive nearly triangular weights.
4. Power of the MPF satisfies the following inequalities.

Proposition 3.11 *Let the filter $\Phi_1^{(i)}$ be the MPF generated by an eigenvector $U_i = (u_1, \ldots, u_L)^{\mathrm{T}}$. Then its power satisfies the inequality $\mathscr{E}\Phi_1^{(i)} \le 1/L$.*

Proof The proof of the proposition results from the following inequality:

$$\|\mathbf{h}_\Psi * \operatorname{rev}\mathbf{h}_\Psi\| \le \sum_{j=1}^{L} |u_j| \cdot \|\mathbf{h}_\Psi\| = \sum_{j=1}^{L} |u_j| \cdot \|U\| = \sum_{j=1}^{L} |u_j| \le \sqrt{L}\|U\| = \sqrt{L}.$$

□

Proposition 3.12 *Let $\Phi_1^{(I)}$ be the MPF generated by eigenvectors $\{U_i, i \in I\}$ where $|I| = r$. Then its power satisfies the inequality $\mathscr{E}\Phi_1^{(I)} \le r^2/L$.*

Proof By the linearity of the grouping operation, $\Phi_1^{(I)} = \sum_{i\in I} \Phi_1^{(i)}$, and therefore, by Proposition 3.11 we have:

$$\mathscr{E}\Phi_1^{(I)} = \left\|\sum_{i\in I} \mathbf{h}_{\Phi_1^{(i)}}\right\|^2 \le \left(\sum_{i\in I} \left\|\mathbf{h}_{\Phi_1^{(i)}}\right\|\right)^2 = \left(\sum_{i\in I} \sqrt{\mathscr{E}\Phi_1^{(i)}}\right)^2 \le r^2/L.$$

□

5. A direct consequence of Proposition 3.12 and inequality (3.28) is the inequality $\Delta_a \Phi_1^{(I)} \le r^2/(a^2 L)$. This means that for any threshold a, the support of filter frequency response tends to 0 as $L \to \infty$. This effect is clearly seen in Fig. 2.22 (Sect. 2.4.3) showing the smoothing effect of Basic SSA.
6. Let us define for $\omega \in (-0.5, 0.5]$:

$$g_U(\omega) = \frac{1}{L}\left|\sum_{j=1}^{L} u_j e^{-\mathrm{i}2\pi\omega j}\right|^2 . \tag{3.32}$$

The function g_U is closely related to the periodogram Π_u^L introduced in (2.10) of Sect. 2.3.1.1: $g_U(k/L) = L\,\Pi_u^L(k/L)/2$ for $0 < k < N/2$ and $g_U(k/L) = L\,\Pi_u^L(k/L)$ otherwise. It appears that the frequency response of the MPF is almost the same as the periodogram of the vector U.

Proposition 3.13 *Let A_{Φ_1} be the frequency response of the MPF filter Φ_1. Then $g_U(\omega) = A_{\Phi_1}(\omega)$.*

Proof Recall that $\Phi_1 = \Theta_1 \circ \Psi/L$, where $\mathbf{h}_\Psi = (\ldots, 0, \overset{\circ}{u}_L, u_{L-1}, \ldots, u_1, 0, \ldots)$ and $\mathbf{h}_{\Theta_1} = \mathrm{rev}\,\mathbf{h}_\Psi$. Also, from the theory of linear filters [37] we have $A_{\Phi_1\circ\Psi}(\omega) \equiv A_{\Phi_1}(\omega)A_\Psi(\omega)$. Then

$$A_{\Phi_1}(\omega) = \frac{1}{L}\left|\sum_{j=0}^{L-1} u_{L-j} e^{-\mathrm{i}2\pi\omega j}\right| \cdot \left|\sum_{j=0}^{1-L} u_{L+j} e^{-\mathrm{i}2\pi\omega j}\right| = \frac{1}{L}\left|\sum_{j=1}^{L} u_j e^{-\mathrm{i}2\pi\omega j}\right|^2 .$$

□

7. It follows from Proposition 3.13 that for SSA identification and interpretation of the SVD components, the periodogram analysis of eigenvectors can be very helpful. Also, an automatic identification of components introduced in Sect. 2.4.5 is based on properties of periodograms of eigenvectors and therefore can also be expressed in terms of the frequency response of the MPF.

3.9.4 Last Point Filter and Forecasting

The last-point filter (LPF) is not really a filter as it is used only for the reconstruction of the last point: $\widetilde{x}_N = \sum_{i=0}^{L-1} u_L u_{i+1} x_{N-i}$. The reconstruction by the eigentriples with numbers from the set I has the following form:

$$\widetilde{x}_N^{(I)} = \sum_{k=0}^{L-1}\left(\sum_{i\in I} u_L^{(i)} u_{k+1}^{(i)}\right) x_{N-k}. \tag{3.33}$$

However, it is the only reconstruction filter that is causal. This has two consequences. First, the LPF of a finite-rank series is closely related to the LRR governing and forecasting this time series. Second, the so-called Causal SSA (or last-point SSA) can be constructed by means of the use of the last reconstructed points of the accumulated data. Since in Causal SSA the LPF is applied many times, studying properties of LPF is important.

Let the signal $\mathbb{S}_N$ has rank r and is governed by an LRR. Unbiased causal filters of order L and linear recurrence relations of order $L-1$ are closely related. In particular, if the causal filter is given by $s_j = \sum_{k=0}^{L-1} a_{L-k} s_{j-k}$ and $a_L \neq 1$, then this filter generates the following LRR of order $L-1$: $s_j = \sum_{k=1}^{L-1} c_{L-k} s_{j-k}$, where $c_k = a_k/(1-a_L)$.

Similar to the minimum-norm LRR, the minimum-power filters can be considered. It appears that the LPF has minimal power among all unbiased filters. This follows from the relation between LRRs and causal filters. Denote by P_r the orthogonal projector onto the signal subspace. The LPF has the form $s_N = (\mathrm{P}_r S_K)_L = A^{\mathrm{T}} S_K$, where $A = \mathrm{P}_r \mathbf{e}_L$, while the min-norm LRR is produced by the last-point filter, i.e. $R = \underline{A}/(1-a_L)$.

In the general case, the filter (3.33) can be rewritten as $\widetilde{x}_N = (\mathrm{P}_r X_K)_L$, where X_K is the last L-lagged vector, P_r is the projector on the SSA estimate of the signal subspace $\mathrm{span}(U_i, i \in I)$. Formally applying this filter to the whole time series, we obtain the series of length K consisting of the last points of the reconstructed lagged vectors $\widetilde{X}_k$.

Note that if we use other estimates of the signal subspace, then we obtain other versions of the last-point filter.

3.9.5 *Causal SSA (Last-Point SSA)*

Let $\mathbb{X}_\infty = (x_1, x_2, \ldots)$ be an infinite series, $\mathbb{X}_M = (x_1, \ldots, x_M)$ be its subseries of length M, L be fixed, $\mathfrak{L}(M)$ be a subspace of R^L, $\mathrm{P}(M)$ be a projector to $\mathfrak{L}(M)$, $A(M) = \mathrm{P}_r(M)\mathbf{e}_L$, $K = K(M) = M - L + 1$.

Introduce the series $\breve{\mathbb{X}}_\infty$ as follow. Define $(\breve{\mathbb{X}}_\infty)_M = (\mathrm{P}(M) X_{K(M)})_L = A(M)^{\mathrm{T}} X_{K(M)}$, where $X_{K(M)}$ is the last L-lagged vector of $\mathbb{X}_M$. Thus, $(\breve{\mathbb{X}}_\infty)_M$ is a linear combination of the last L terms of $\mathbb{X}_M$ with coefficients depending on M; that is, $\breve{\mathbb{X}}_\infty$ can be considered as a result of application of a sequence of different causal filters to $\mathbb{X}_\infty$.

If $\mathfrak{L}_r(M) = \mathfrak{L}(M) = \mathrm{span}(U_1(M), \ldots, U_r(M))$, where $U_1(M), \ldots, U_r(M)$ are the signal eigenvectors produced by SSA with window length L applied to $\mathbb{X}_M$, then this sequence of causal filters is called Causal SSA. In this case, $(\breve{\mathbb{X}})_M$ is equal to the last point of SSA reconstruction $\widetilde{\mathbb{X}}_M$, which in turn is equal to the last coordinate of the projection of the last lagged vector of $\mathbb{X}_M$ to $\mathfrak{L}_r(M)$.

Given that $\mathfrak{L}_r(M)$ is used as an estimate of the signal subspace, M should be large enough to provide a good estimate. Therefore, we need to introduce a starting point M_0 (with $M_0 > L$) in Causal SSA and consider $M \geq M_0$ only.

Since the result of application of Causal SSA is a sequence $\check{\mathbb{X}}_\infty$ built from SSA reconstructions of the last points of the subseries, Causal SSA can be called Last-point SSA.

Remark 3.12 Let us fix N, L and consider $\mathbb{X}_N$, the corresponding $U_1, \ldots, U_r$ and $\mathfrak{L}_r(M) = \mathrm{span}(U_1, \ldots, U_r)$ for any M. Then the series $\check{\mathbb{X}}_N$ is the result of application of the last-point filter to $\mathbb{X}_N$. Assuming that the estimates of the signal subspace on the base of $\mathbb{X}_M$ are stable for large enough M, we can conclude that the result of Causal SSA will be close to the result of the last-point filter (LPF) applied to the whole series.

Note also that in the considered particular case, $\check{\mathbb{X}}_N$ (more precisely, its last K points; the first $L-1$ points are not defined or can be set to zero) coincides with the last row of the reconstructed matrix $\widehat{\mathbf{X}}$ of the series $\mathbb{X}_N$. That is, $\check{\mathbb{X}}_N$ is similar to the result of SSA reconstruction before the diagonal averaging is made.

Causality yields the following relation: under the transition from $\mathbb{X}_M$ to $\mathbb{X}_{M+1}$, the first M points of the $\check{\mathbb{X}}_{M+1}$ coincide with $\check{\mathbb{X}}_M$. This is generally not true if we consider the reconstructions $\widetilde{\mathbb{X}}_M$ for $\mathbb{X}_M$ obtained by the conventional SSA. The effect $(\widetilde{\mathbb{X}}_M)_j \neq (\widetilde{\mathbb{X}}_{M+1})_j$, $j \leq M$, is called 'redrawing'. For real-life time series we usually have redrawing for all j and the amount of redrawing depends on j. Redrawing of only a few last points is usually of interest. Moreover, redrawing of local extremes of the series is more practically important than redrawing of regular series points. Small values of redrawing indicate stability of SSA decompositions and hence stability of time series structure. The amount of redrawing can be assessed by visual examination or measured using the variance of the redrawings at each time moment of interest. These variances can be averaged if needed.

Generally, the reconstruction has no delay (at least, the middle-point filter has zero phase shift). On the other hand, delays in Causal SSA are very likely. In a sense, a redrawing in SSA is converted to a delay in Causal SSA. If $\mathfrak{L}_r$ corresponds to an exactly separated component of the time series $\mathbb{X}$, then SSA has no redrawing and Causal SSA has no delay. In conditions of good approximate separability the redrawing is almost absent and the delay is small.

Example Let us demonstrate the described effects on the 'S&P500' example introduced in Sect. 2.5.1. Figure 3.7 shows the result of Causal SSA with window length $L = 30$, $\mathfrak{L}_r(M) = \mathrm{span}(U_1(M), U_2(M))$ and $M_0 = 200$. The delay is clearly seen. If we consider non-causal (Basic) SSA reconstructions of cumulative subseries, then the redrawing takes place (Fig. 3.8). This redrawing increases in the points of local maximums and minimums.

Finally, let us note that if the direction of change of the time series is of primary concern (rather than the values themselves), then, instead of taking differences of Causal SSA series, it may be worthwhile considering the time series which consists of differences of the last two points of reconstructions. The result should be expected to be more stable since the reconstruction of the last-but-one point has better accuracy than that of the last point.

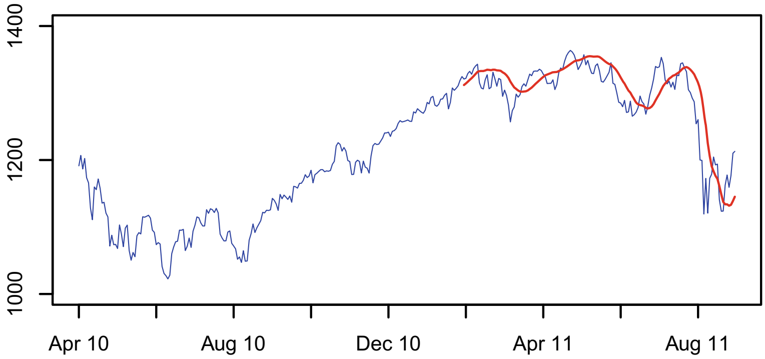


Fig. 3.7 S&P500: Causal SSA

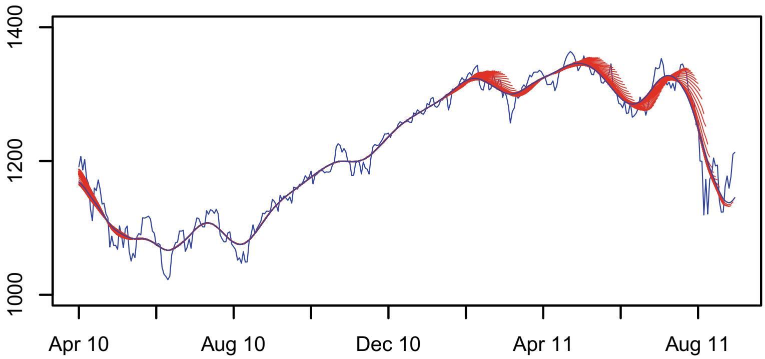


Fig. 3.8 S&P500: Non-causal SSA with redrawing

3.10 Multidimensional/Multivariate SSA

In Sect. 2.6, we briefly discussed the approach to construction of multidimensional extensions of SSA for decomposing the initial object into identifiable components. This general approach to SSA, developed in [23, 24], makes it easy to extend the algorithms of SSA-related methods developed for the analysis of one-dimensional (1D) time series to the cases of collections of time series and digital images.

This concerns the approaches to decomposition, refined decompositions including rotations, filtering, parameters estimation, forecasting, gap filling and also to the choice of parameters. In this section, we dwell on some differences between the 1D and multidimensional cases.

The main notions in SSA are signal subspace, separability, rank, LRR. Their definitions are the same for different dimensionalities.

The trajectory space is constructed as the span of columns or rows of the trajectory matrix. Signal subspace is defined as the trajectory space of the signal of interest. The (weak) separability is determined exactly in the same way as in the 1D case via orthogonality of the trajectory spaces of time series components which should be separated. The strong separability is determined as non-coincidence of the singular values of trajectory matrices of these components.

Rank of a multidimensional object is defined as rank of its trajectory matrix. If for all sufficiently large window sizes and object sizes: (a) the trajectory matrix is rank-deficient then the object is called rank deficient, and (b) the ranks of the trajectory matrices equal r then we say that the object has finite rank r.

SSA forecasting and gap filling are also similar to the 1D case: first, the signal subspace is estimated and then the imputation of unknown data is performed to keep the signal subspace.

The objects of finite rank produce a class of objects, which satisfy LRRs. Again, the objects satisfying LRRs have the parametric form of a finite sum of products of multivariate polynomials, exponentials and sinusoids (see e.g. [18, Proposition 7] and [23, Appendix B]). LRRs correspond to characteristic polynomials of several variables, whose roots are needed for estimation of parameters.

Although both 2D-SSA and MSSA comply with these principles (moreover, MSSA can be considered as a particular case of 2D-SSA), there are significant differences. For 2D objects, we consider both dimensions from the same viewpoint; in particular, the words ‘sufficiently large’ are related to both sizes of the object and the window in both dimensions. For multivariate time series, the number of time series in the collection is fixed and the window size with respect to this dimension is equal to 1.

3.10.1 MSSA

Despite formally MSSA is a particular case of 2D-SSA, from the viewpoint of analysis, MSSA is closer to 1D-SSA than to 2D-SSA. In particular, finite-rank collections of time series are described in terms of the collections of time series, where each series satisfies an LRR. Thus, we will not discuss the common things that have been thoroughly discussed for time series.

Ranks and common structure of time series. In MSSA analysis, it is important if all time series in the chosen collection have similar behaviour; if these time series are different, then it is better to analyse them separately. In the context of SSA, similar behaviour means a common structure of signal subspaces. From the viewpoint of LRRs, the characteristic polynomials of different series from the collection have the same (or with a large intersection) set of roots. For instance, sinusoids with the same periods have the same structure; the same is true for linear trends.

An indicator of the same structure is the relation between the MSSA rank r of the collection of signals and the SSA ranks r_i of signals, $i = 1, \dots, s$. The same structure of all signals corresponds to the case $r_1 = \dots = r_s = r$, while entirely different structures yield the equality $r = \sum_{i=1}^{s} r_i$.

Within the framework of SSA, we cannot talk about causality, which implies time shifts, since the MSSA method is invariant with respect to shifts in time. However, we can talk 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. The supportiveness depends on the similarity of the signals subspaces and on noise levels. Note that if one time series is supportive for another one, this does not imply that the second time series is supportive for the first one. An easy example illustrating this is: both time series consist of the same signal $s_n = \cos(2\pi\omega n)$ corrupted by different noises, very small and very large, respectively.

Forecasting. We have mentioned in Sect. 2.6.1 that for MSSA, the rows and columns of the trajectory matrix have different forms. Therefore, the methods of MSSA forecasting require special attention.

As in 1D-SSA, methods of MSSA forecasting can be subdivided into recurrent and vector forecasting. In contrast with 1D-SSA, there exist two kinds of MSSA forecasting: row forecasting and column forecasting; this depends on which of the two spaces the forecasting is made (row or column space respectively). In total, there are four main variants of MSSA forecasting: recurrent column forecasting, recurrent row forecasting, vector column forecasting and vector row forecasting.

Note that there are several different names for the same SSA forecasting methods, see [17, 23, 29]. In Sect. 2.6.1, we have explained the choice of orientation of the MSSA trajectory matrix and the connection between the horizontally-stacked and vertically-stacked trajectory matrices of separate time series. We follow [23, 24] and use the name 'column' and 'row' with respect to the horizontally-stacked trajectory matrices as defined in Sect. 2.6.1. In the column forecasting methods, each time series in the collection is forecasted separately in a given common subspace (that is, using a common LRR). In the row forecasting methods, each series is forecasted with the help of its own LRR applied to the whole set of series from the collection. The capabilities of the RSSA package for MSSA forecasting are discussed in [24, Sect. 4.3.3].

Missing data imputation, parameter estimation and other subspace-based methods are performed in the same manner as for 1D-SSA.

Numerical comparison of forecasts for a simulated data. In [17, 23, 29], a comparison was performed for series without trends. Here we add a linear trend to both time series. Let us assume that we observe $(\mathbb{X}^{(1)}, \mathbb{X}^{(2)}) = (\mathbb{S}^{(1)}, \mathbb{S}^{(2)}) + (\mathbb{N}^{(1)}, \mathbb{N}^{(2)})$, where $(\mathbb{S}^{(1)}, \mathbb{S}^{(2)})$ is a two-dimensional signal, $\mathbb{N}^{(1)}$ and $\mathbb{N}^{(2)}$ are realizations of independent white Gaussian noises. Then we can use the standard simulation techniques to obtain estimates of the mean square errors (MSE) for the reconstruction and forecasting of $(\mathbb{S}^{(1)}, \mathbb{S}^{(2)})$ by the indicated SSA methods. The resultant MSE is calculated as the mean of $\mathrm{MSE}^{(1)}$ and $\mathrm{MSE}^{(2)}$ for $\mathbb{S}^{(1)}$ and $\mathbb{S}^{(2)}$ correspondingly. We take the following parameters for the simulation of the time series: $N = 71$, the variance of each noise components is $\sigma^2 = 25$, the number of replications is 1000:

$$s_k^{(1)} = 30\cos(2\pi k/12) + k, \quad s_k^{(2)} = 30\cos(2\pi k/12 + \pi/4) + 100 - k, \tag{3.34}$$

$k = 1, \ldots, N$.

Table 3.4 MSE of signal forecasts

Example (3.34)	$L = 12$	$L = 24$	$L = 36$	$L = 48$	$L = 60$
Recurrent					
MSSA-column	1879.08	32.15	**20.55**	21.70	22.44
MSSA-row	177.20	34.02	23.54	**19.88**	20.64
1D-SSA	2020.78	331.35	206.63	**126.13**	157.24
Vector					
MSSA-column	400.12	33.80	18.95	**18.64**	19.31
MSSA-row	1809.04	58.67	**18.51**	18.98	20.06
1D-SSA	245.45	159.69	152.60	**148.96**	195.31

The results of investigation for different window lengths L are summarized in Table 3.4. The 24-term forecast was performed. The cells corresponding to the method with row-best forecasting accuracy are shown in bold and the overall best is in blue color.

Note that both signals are of the same structure and have rank 4. Therefore the rank of their collection is 4. One can see that 1D-SSA provides inaccurate forecasts. These numerical results are similar to the conclusions given in [23] and confirm that for recurrent and vector forecasting, the choice of window length differs for different forecasting methods. Also, vector forecasting is more accurate for the proper choice of parameters; this agrees with similar conclusions from [23, 29] for time series without trends.

Different forecasts of wine sales. Consider the time series with sales of 'Fortified wine' and 'Dry wine'. They have similar range of values and therefore we do not scale them. Let us compare different forecasts. As the structure of the time series is similar to the sum of a linear trend and sinusoids due to seasonality, the window length is chosen in view of results given in Table 3.4. The length of both time series is equal to $N = 176$. Therefore, we take $L = 118$ (approximately equal to $2N/3$) for the recurrent forecasting by rows and for vector forecasting by columns. The window length $L = 88$ (equal to $N/2$) was taken for the recurrent forecasting by columns and for vector forecasting by rows. Figure 3.9 shows that the forecasts are very similar.

3.10.2 2D-SSA

Recall that in 2D-SSA, the column space consists of spanned windows of size $L_1 \times L_2$ while the row space consists of spanned windows of size $K_1 \times K_2$, where $K_i = N_i - L_i + 1$. The trajectory matrix has size $L_1L_2 \times K_1K_2$. Formally, the column

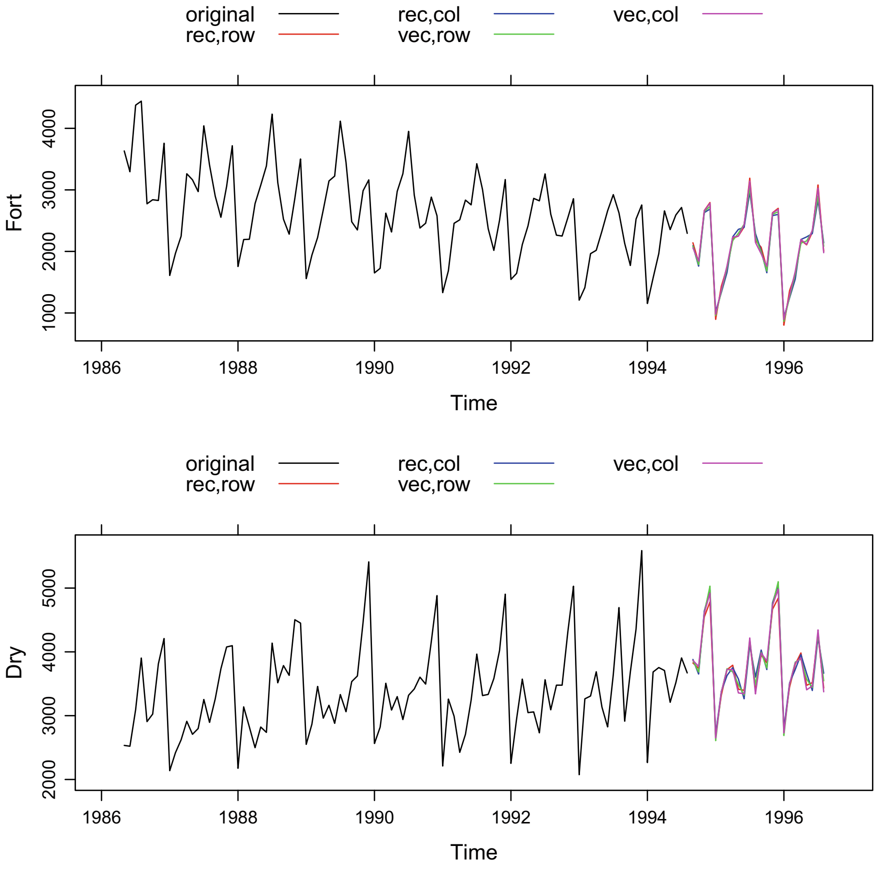


Fig. 3.9 Fortified and Dry wines: SSA forecasting

and row spaces have the same sense up to values of L_i and K_i. Let us briefly discuss the differences from the 1D case; they are mostly related to the rank and LRRs.

Ranks and separability. For 1D data, any rank-deficient infinite-length series has finite rank. For 2D-SSA, this is not so. For example, for the image with $x_{ij} = \sin i \cdot \ln(j+1)$ and any window size (L_x, L_y) such that $2 \le \min(L_x, K_x) \le \lfloor N_x/2 \rfloor$ $1 \le \min(L_y, K_y) \le \lfloor N_y/2 \rfloor$, the rank of the trajectory matrix is $2 \cdot \min(L_y, K_y)$.

Another difference from the 1D case is that the ranks of interpretable components in the multidimensional case are generally larger. For example, the SSA rank of the harmonic $x_n = A\cos(2\pi\omega n + \phi)$ is equal to 2 for $0 < \omega < 0.5$. As follows from [19, Proposition 4.7], the 2D-SSA rank of a 2D harmonic can vary from 2 to 4. For example the rank of $x_{kl} = A\cos(2\pi\omega^{(X)}k + 2\pi\omega^{(Y)}l)$ equals 2, whereas the rank of $x_{kl} = A\cos(2\pi\omega^{(X)}k)\cos(2\pi\omega^{(Y)}l)$ equals 4.

Consider a 2D array (x_{kl}) with elements $x_{kl} = y_k z_l$, where $\mathbb{Y} = (y_1, \ldots, y_{N_1})$ has rank d_1 and $\mathbb{Z} = (z_1, \ldots, z_{N_2})$ has rank d_2. It follows from [19, Theorem 2.1] that the 2D-SSA rank of this array is equal to $d_1 d_2$ and the singular values of its trajectory matrix consist of products of the singular values of $\mathbb{Y}$ and $\mathbb{Z}$. This explains why the ranks of 2D components should be expected to be large and why the singular values of 2D arrays could be very different. In the signal plus noise 2D model, the latter would almost inevitably imply the lack of strong separability.

Linear recurrence relations. The theory of nD-SSA is related to such notion as ideal of polynomials in n variables [18]. In Sect. 3.2, we described how LRRs for time series are connected with polynomials of one variable called characteristic polynomials. In a similar way, LRRs for nD objects are connected with polynomials in n variables. The existence of a unique minimal LRR in the 1D case is caused by the fact that each ideal in the ring of polynomials in one variable is principal; that is, it is generated by a single polynomial. If $n > 1$, this is not so. There is a set of generating polynomials and, therefore, a set of LRRs, which can be used e.g. for forecasting and gap filling. The set of generating polynomials is closely related to Gröbner bases. This technique is used in the case of arrays, which can be continued in all directions. Since collections of time series, where MSSA is applicable, can be continued in one direction only, the discussion above is not relevant to MSSA. This is despite the fact that from the algorithmic point of view, MSSA is a particular case of 2D SSA.

Forecasting, gap filling, parameter estimation. Forecasting for digital images is not an appropriate task except for the case when one of the dimensions is time while the other dimensions are spatial. Generally, in the nD case with $n > 1$, there is no single LRR that can perform forecasting in several directions and hence a set of LRRs (which is not uniquely defined!) performs this task. This specificity holds for the problem of subspace-based gap filling. Iterative gap filling can be performed in exactly the same manner as in the 1D case [7].

Estimation of parameters by the ESPRIT method (see Sect. 3.8.2) can be extended to the 2D case; the related 2D-modification is called 2D-ESPRIT [41]. The estimation of frequencies by 2D-ESPRIT based on the use of Hankel-block-Hankel trajectory matrices appearing in 2D-SSA, is rather popular. The R code for 2D-ESPRIT is considered in [24, Sect. 5.3.4].

Filtering. Adaptive filtering is an important application of SSA. Let us show an example, which visually illustrates the filtering ability of 2D-SSA. The image of Saturn's rings was taken in visible light with the Cassini spacecraft wide angle camera on Oct. 27, 2004, https://photojournal.jpl.nasa.gov/catalog/PIA06529 (image credit to NASA/JPL/Space Science Institute). We consider the version of this image in resolution 320 × 512 and note that we have removed the Saturn's moons from the image. After 2D-SSA decomposition with the window 50 × 50, we obtain the decomposition depicted in Fig. 3.10. The reconstruction by ET1–8 shows a general form of the rings' image, the reconstruction by ET9–30 reflects moderate changes and finally the residual shows sharp changes. The latter can help in distinguishing fine image details.

Real-life applications related to filtering abilities of 2D- and 3D-SSA can be found in [22, 43, 44].

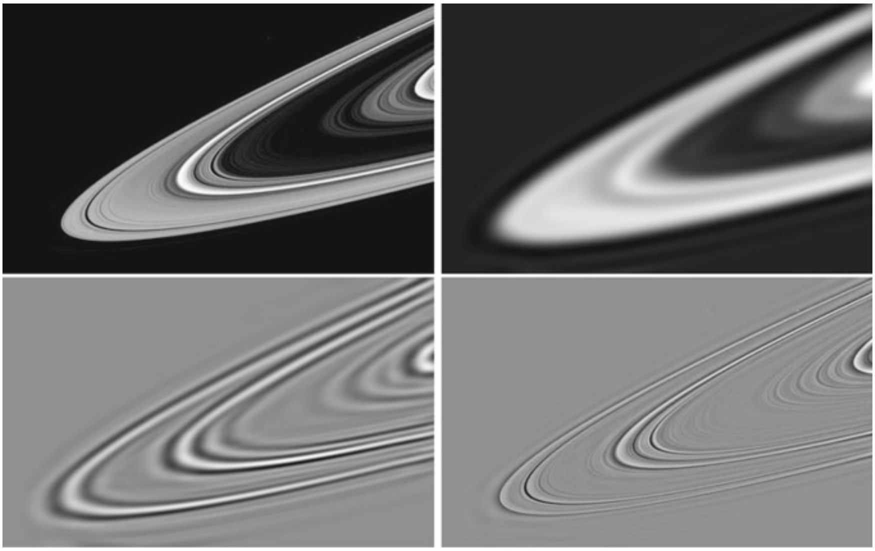


Fig. 3.10 Saturn's rings: Original image (left,top), reconstructions by ET1–8 (right,top), ET 9–30 (left,bottom) and residuals (right, bottom)

References

1. Badeau R, David B, Richard G (2004) Selecting the modeling order for the ESPRIT high resolution method: an alternative approach. Proc IEEE ICASSP 2:1025–1028
2. Badeau R, Richard G, David B (2008) Performance of ESPRIT for estimating mixtures of complex exponentials modulated by polynomials. IEEE Trans Signal Process 56(2):492–504
3. Barkhuijsen H, de Beer R, van Ormondt D (1987) Improved algorithm for noniterative time-domain model fitting to exponentially damped magnetic resonance signals. J Magn Reson 73:553–557
4. Beckers J, Rixen M (2003) EOF calculations and data filling from incomplete oceanographic data sets. Atmos Ocean Technol 20:1839–1856
5. Bezerra LH, Bazan FSV (1998) Eigenvalue locations of generalized companion predictor matrices. SIAM J Matrix Anal & Appl 19(4):886–897
6. 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
7. Jannis von Buttlar, Zscheischler J, Mahecha M (2014) An extended approach for spatiotemporal gapfilling: dealing with large and systematic gaps in geoscientific datasets. Nonlinear Process Geophys 21(1):203–215
8. Cadzow JA (1988) Signal enhancement: a composite property mapping algorithm. IEEE Trans Acoust 36(1):49–62
9. Chu MT, Funderlic RE, Plemmons RJ (2003) Structured low rank approximation. Linear Algebra Appl 366:157–172
10. Efron B, Tibshirani R (1986) Bootstrap methods for standard errors, confidence intervals and other measures of statistical accuracy. Stat Sci 1(1):54–75
11. Gel'fond A (1971) Calculus of finite differences. Translated from the Russian. International monographs on advanced mathematics and physics. Hindustan Publishing Corp., Delhi

12. Gillard J, Zhigljavsky A (2013) Optimization challenges in the structured low rank approximation problem. J Global Optim 57(3):733–751
13. Gillard J, Zhigljavsky A (2016) Weighted norms in subspace-based methods for time series analysis. Numer Linear Algebra Appl 23(5):947–967
14. Golyandina N (2010) On the choice of parameters in singular spectrum analysis and related subspace-based methods. Stat Interface 3(3):259–279
15. Golyandina N (2020) Particularities and commonalities of singular spectrum analysis as a method of time series analysis and signal processing. WIREs Comput Stat 12(4):e1487
16. 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
17. 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
18. Golyandina N, Usevich K (2009) An algebraic view on finite rank in 2D-SSA. In: Proceedings of the 6th St.Petersburg Workshop on Simulation, June 28-July 4, St. Petersburg, Russia, pp 308–313
19. 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
20. Golyandina N, Zhigljavsky A (2020) Blind deconvolution of covariance matrix inverses for autoregressive processes. Linear Algebra Appl 593:188–211
21. Golyandina N, Nekrutkin V, Zhigljavsky A (2001) Analysis of time series structure: SSA and related techniques. Chapman&Hall/CRC, London
22. Golyandina N, Usevich K, Florinsky I (2007) Filtering of digital terrain models by two-dimensional singular spectrum analysis. Int J Ecol Dev 8(F07):81–94
23. 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
24. Golyandina N, Korobeynikov A, Zhigljavsky A (2018) Singular spectrum analysis with R. Springer, Berlin
25. Golyandina N, Korobeynikov A, Zhigljavsky A (2018) Site-companion to the book 'Singular spectrum analysis with R'. https://ssa-with-r-book.github.io/
26. de Groen P (1996) An introduction to total least squares. Nieuw Archief voor Wiskunde 14:237–253
27. Hall MJ (1998) Combinatorial theory. Wiley, New York
28. Harris T, Yan H (2010) Filtering and frequency interpretations of singular spectrum analysis. Phys D 239:1958–1967
29. Hassani H, Mahmoudvand R (2013) Multivariate singular spectrum analysis: a general view and vector forecasting approach. Int J Energy Stat 01(01):55–83
30. Kondrashov D, Ghil M (2006) Spatio-temporal filling of missing points in geophysical data sets. Nonlinear Process Geophys 13(2):151–159
31. Kumaresan R, Tufts DW (1980) Data-adaptive principal component signal processing. In: Proceeding of the IEEE conference on decision and control, Albuquerque, pp 949–954
32. Kumaresan R, Tufts DW (1983) Estimating the angles of arrival of multiple plane waves. IEEE Trans Aerosp Electron Syst AES-19(1):134–139
33. Kung SY, Arun KS, Rao DVB (1983) State-space and singular-value decomposition-based approximation methods for the harmonic retrieval problem. J Opt Soc Amer 73(12):1799–1811
34. Markovsky I (2019) Low rank approximation: algorithms, implementation, applications (Communications and Control Engineering), 2nd edn. Springer, Berlin
35. Markovsky I, Usevich K (2014) Software for weighted structured low-rank approximation. J Comput Appl Math 256:278–292
36. Nekrutkin V (2010) Perturbation expansions of signal subspaces for long signals. Stat Interface 3:297–319

37. Oppenheim AV, Schafer RW (1975) Digital signal processing. Prentice-Hall, Upper Saddle River
38. Pakula L (1987) Asymptotic zero distribution of orthogonal polynomials in sinusoidal frequency estimation. IEEE Trans Inf Theor 33(4):569–576
39. Pepelyshev A, Zhigljavsky A (2010) Assessing the stability of long-horizon SSA forecasting. Stat Interface 3:321–327
40. Roy R, Kailath T (1989) ESPRIT: estimation of signal parameters via rotational invariance techniques. IEEE Trans Acoust 37:984–995
41. Sahnoun S, Usevich K, Comon P (2017) Multidimensional ESPRIT for damped and undamped signals: algorithm, computations, and perturbation analysis. IEEE Trans Signal Process 65(22):5897–5910
42. Schoellhamer D (2001) Singular spectrum analysis for time series with missing data. Geophys Res Lett 28(16):3187–3190
43. Shlemov A, Golyandina N, Holloway D, Spirov A (2015) Shaped 3D singular spectrum analysis for quantifying gene expression, with application to the early *Drosophila* embryo. BioMed Res Int 2015(Article ID 986436):1–18
44. Shlemov A, Golyandina N, Holloway D, Spirov A (2015) Shaped singular spectrum analysis for quantifying gene expression, with application to the early *Drosophila* embryo. BioMed Res Int 2015(Article ID 689745)
45. Stoica P, Moses R (1997) Introduction to spectral analysis. Prentice Hall, Upper Saddle River
46. Tufts DW, Kumaresan R (1982) Estimation of frequencies of multiple sinusoids: making linear prediction perform like maximum likelihood. Proc IEEE 70(9):975–989
47. Usevich K (2010) On signal and extraneous roots in Singular Spectrum Analysis. Stat Interface 3(3):281–295
48. Van Huffel S, Chen H, Decanniere C, van Hecke P (1994) Algorithm for time-domain NMR data fitting based on total least squares. J Magn Reson Ser A 110:228–237
49. Zhigljavsky A, Golyandina N, Gryaznov S (2016) Deconvolution of a discrete uniform distribution. Stat Probab Lett 118:37–44
50. Zvonarev N, Golyandina N (2017) Iterative algorithms for weighted and unweighted finite-rank time-series approximations. Stat Interface 10(1):5–18
51. Zvonarev N, Golyandina N (2018) Image space projection for low-rank signal estimation: Modified Gauss-Newton method. arXiv:1803.01419