# Tensors for signal and frequency estimation in subspace-based methods: when they are useful?

Nikita Khromov, Nina Golyandina

St. Petersburg State University
Department of Statistical Modeling

DD.09.2025, CDAM'2025

# Introduction to Singular Spectrum Analysis (SSA)

Problems that can be solved by SSA-related methods:

- Signal extraction
- Frequency estimation
- Smoothing and Noise reduction
- Signal decomposition (Trend and Periodicity extraction)
- Forecasting
- Missing data imputation
- Change in structure detection
- Many others...

# SSA Materials

Books:

- J.Elsner and A.Tsonis. Singular Spectrum Analysis: A New Tool in Time Series Analysis, Plenum, 1996.
- N.Golyandina, V.Nekrutrin and A.Zhigljavsky. Analysis of Time Series Structure: SSA and Related Techniques, CRC Press, 2001.
- S.Sanei and H.Hassani. Singular Spectrum Analysis for Biomedical Signals, CRC Press, 2016.
- N.Golyandina, A.Korobeynikov and A.Zhigljavsky. Singular spectrum analysis with R, Springer, 2018.
- N.Golyandina and A.Zhigljavsky. Singular Spectrum Analysis for Time Series, Springer, 2013, 2020 (2nd Edition).

Implementations:

- R Package: Rssa
  https://CRAN.R-project.org/package=Rssa
- Python Package: py-ssa-lib (less features)
  https://pypi.org/project/py-ssa-lib

# SSA Decomposition example

Decomposition of time series:

- Low-frequency component + high-frequency component
- Signal + noise
- Trend + Seasonality + Noise

example-image.pdf

*Some data*: demonstration of series decomposition with SSA

# ESPRIT Frequency estimation example

ESPRIT — SSA-related method for parameters estimation

example-image.pdf

*Pole motion data probably*

1. Estimate — interpretation
2. Estimate — interpretation

# Complex Time Series

Common origins of complex-valued time series:

- Can be constructed from two related features
- Arise as a result of applying the Fourier transform to real data

# SSA Algorithm: Embedding

**Input:** time series $\mathsf{X} = (x_1, x_2, \ldots, x_N)$, window length $L$, signal rank $r$.

1. **Embedding**. Constructing the $L$-*Trajectory* Hankel matrix $\mathbf{X} \in \mathbb{C}^{L\times K}$ from the series $\mathsf{X}$, where $K = N - L + 1$:

$$\mathbf{X} = \mathcal{T}^{(L)}(\mathsf{X}) = \begin{pmatrix} x_1 & x_2 & x_3 & \ldots & x_K \\ x_2 & x_3 & x_4 & \ldots & x_{K+1} \\ x_3 & x_4 & x_5 & \ldots & x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & x_{L+2} & \ldots & x_N \end{pmatrix}$$

# SSA Algorithm: Decomposition, Grouping, Reconstruction

2. **Decomposition**. Constructing the singular value decomposition (SVD) of matrix $\mathbf{X}$: $\mathbf{X} = \sum_{j=1}^{\operatorname{rank}\mathbf{X}} \sqrt{\lambda_j} U_j V_j^{\mathrm{H}} = \sum_{j=1}^{\operatorname{rank}\mathbf{X}} \widehat{\mathbf{X}}_j$ where $\mathrm{H}$ denotes Hermitian conjugation, $U_j$ and $V_j$ are left and right singular vectors of $\mathbf{X}$, $\sqrt{\lambda_j}$ — its singular values in descending order.

3. **Grouping**. Grouping the terms $\widehat{\mathbf{X}}_j$ from the decomposition related to the signal: $\mathbf{S} = \sum_{j=1}^{r} \widehat{\mathbf{X}}_j = \Pi_r \mathbf{X}$, where $\Pi_r$ is the projector onto the space of matrices with rank not greater than $r$.

4. **Reconstruction**. Applying projection onto the space of Hankel matrices: $\widetilde{\mathbf{S}} = \Pi_{\mathcal{H}} \widehat{\mathbf{S}}$, and return to the series form: $\widetilde{\mathsf{S}} = \left(\mathcal{T}^{(L)}\right)^{-1} (\widetilde{\mathbf{S}})$

# Series rank

## Definition

Series X has rank $d < N/2$, if the rank of its $L$-trajectory matrix equals $d$ for any $L$ such that $d \leqslant \min(L, N - L + 1)$.
If such $d$ exists, then X is called a series of finite rank.

If the signal S is a series of finite rank, then it is generally recommended to use $\mathrm{rank}(\mathsf{S})$ as parameter $r$ in the SSA method

Series rank examples

- rank of S with $s_n = A\sin(2\pi\omega n + \varphi)$, $0 < \omega < 1/2$, equals 2
- rank of S with $s_n = A\exp(\alpha n)$, $\alpha \in \mathbb{C}$, equals 1

# Signal Model

What we consider a signal $\mathsf{S} = (s_1, s_2, \ldots, s_N)$:

- The trajectory matrix $\mathbf{S} = \mathcal{T}^{(L)}(\mathsf{S})$ is rank-deficient ( $\implies$ the time series is of some finite rank: $\operatorname{rank}(\mathsf{S}) = r$)
- Any signal S can be represented in the form of a finite sum:

$$s_n = \sum_j p_j(n) \exp(\alpha_j n + \mathrm{i}(2\pi\omega_j n + \varphi_j)),$$

  where $p_j(n)$ is a polynomial in $n$

- Real case:

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

ESPRIT method estimates damping factors $\alpha_j$ and frequencies $\omega_j$

# ESPRIT Algorithm: General Idea

$$s_n = \sum_{j=1}^{2} \exp(\alpha_j n + \mathrm{i}(2\pi\omega_j n + \varphi_j)) = A_1 z_1^n + A_2 z_2^n$$

where $A_j = \exp(\mathrm{i}\varphi_j)$, $z_j = \exp(\alpha_j + \mathrm{i}2\pi\omega_j)$

Signal subspace basis is given by

$$\mathbf{M} = \begin{pmatrix} z_1 & z_2 \\ z_1^2 & z_2^2 \\ \vdots & \vdots \\ z_1^L & z_2^L \end{pmatrix} \Rightarrow \overline{\mathbf{M}} = \underline{\mathbf{M}} \begin{pmatrix} z_1 & \\ & z_2 \end{pmatrix} \Rightarrow \underline{\mathbf{M}}^{-} \overline{\mathbf{M}} = \begin{pmatrix} z_1 & \\ & z_2 \end{pmatrix}$$

where $\overline{\mathbf{M}}$ denotes $\mathbf{M}$ without the first row, $\underline{\mathbf{M}}$ — without the last
$\underline{\mathbf{M}}^{-}$ denotes the pseudoinverse of $\underline{\mathbf{M}}$

# ESPRIT Algorithm

**Input**: same as in SSA: X, $L$, $r$

1. **Embedding**. $\mathbf{X} = \mathcal{T}^{(L)}(\mathsf{X})$
2. **Decomposition**. $\mathbf{X} = \sum_{j=1}^{\operatorname{rank} \mathbf{X}} \sqrt{\lambda_j} U_j V_j^{\mathrm{H}}$, $\mathbf{U}_r = [U_1 : U_2 : \ldots : U_r]$
3. **Estimation**. Finding eigenvalues $z_j$ of matrix $\underline{\mathbf{U}}_r^{-} \overline{\mathbf{U}}_r$

   From $z_j = \exp(\alpha_j + \mathrm{i}2\pi\omega_j)$ parameters $\alpha_j$ and $\omega_j$ can be found

# Multi-Channel Time Series, MSSA

$\mathsf{X} = \left(\mathsf{X}^{(1)}, \mathsf{X}^{(2)}, \ldots, \mathsf{X}^{(P)}\right), \qquad \mathsf{X}^{(p)} = \left(x_1^{(p)}, x_2^{(p)}, \ldots, x_N^{(p)}\right)$ – channels

The only change in the algorithms — Embedding step:

$$\mathbf{X} = \mathcal{T}_{\mathrm{MSSA}}^{(L)}(\mathsf{X}) = \left[\mathbf{X}^{(1)} : \mathbf{X}^{(2)} : \ldots : \mathbf{X}^{(P)}\right],$$

$$\mathbf{X}^{(p)} = \mathcal{T}^{(L)}(\mathsf{X}^{(p)})$$

When to chose MSSA over SSA for each channel:

- All channels have "similar" structure
- "Supporting" channels with lower noise level

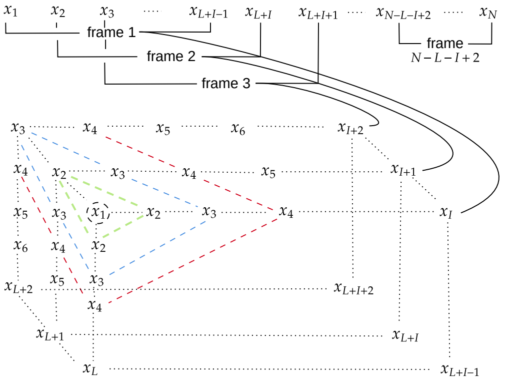
# Intoduction of Tensors

$$\begin{array}{rccccc} \text{Basic SSA:} & \text{Time series X} & \mapsto & \text{Matrix } \mathbf{X} & \mapsto & \mathrm{SVD}(\mathbf{X}) \\ \text{Tensor SSA:} & \text{Time series X} & \mapsto & \text{Tensor } \mathcal{X} & \mapsto & \underbrace{\mathrm{TD}}_{\text{Some Tensor Decomposition}}(\mathcal{X}) \end{array}$$

Tensor SVD Extensions:

- Higher-Order SVD (HOSVD)
- Canonical Polyadic Decomposition (CPD)
- T-SVD
- $(L_r, L_r, 1)$-Decomposition

# Mapping Time Series to Tensor




$$\mathsf{X} = (x_1,\, x_2,\, \ldots,\, x_N)$$

$$\mathsf{X} \mapsto \mathcal{T}^{(I,L)}_{\mathrm{T-SSA}}(\mathsf{X}) = \mathcal{X}$$

$$\mathcal{X} \in \mathbb{C}^{I \times L \times K}$$

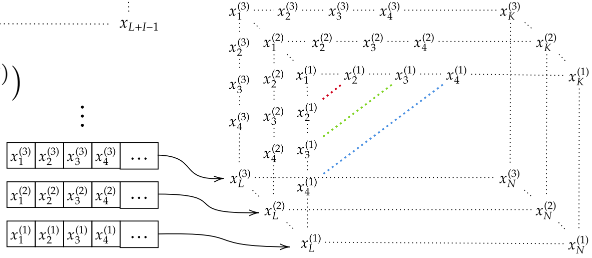
$$K = N - I - L + 2$$

$$\mathsf{X} = \left(\mathsf{X}^{(1)},\, \mathsf{X}^{(2)},\, \ldots,\, \mathsf{X}^{(P)}\right)$$

$$\mathsf{X} \mapsto \mathcal{T}^{(L)}_{\mathrm{T-MSSA}}(\mathsf{X}) = \mathcal{X}$$

$$\mathcal{X} \in \mathbb{C}^{P \times L \times K}$$

$$K = N - L + 1$$



# Higher-Order SVD. Higher-Order Orthogonal Iterations

$$\mathrm{SVD}(\mathbf{X}) = \sum_{j=1}^{\mathrm{rank}(\mathbf{X})} \sqrt{\lambda_j} U_j V_j^{\mathrm{H}}$$

$$\mathrm{HOSVD}(\mathcal{X}) = \sum_{i=1}^{\mathrm{rank}_{\color{red}1}(\mathcal{X})} \sum_{l=1}^{\mathrm{rank}_{\color{red}2}(\mathcal{X})} \sum_{k=1}^{\mathrm{rank}_{\color{red}3}(\mathcal{X})} \mathcal{Z}_{ilk} U_i^{(1)} \circ U_l^{(2)} \circ U_k^{(3)}$$

- $\widetilde{\mathbf{X}} = \sum_{j=1}^{R} \ldots \Rightarrow \left\| \mathbf{X} - \widetilde{\mathbf{X}} \right\|_F = \min_{\mathrm{rank}(\widehat{\mathbf{X}}) \leqslant R} \left\| \mathbf{X} - \widehat{\mathbf{X}} \right\|_F$
- $\widetilde{\mathcal{X}} = \sum_{i=1}^{R_1} \sum_{l=1}^{R_2} \sum_{k=1}^{R_3} \ldots \Rightarrow \left\| \mathcal{X} - \widetilde{\mathcal{X}} \right\|_F \geqslant \min_{\mathrm{rank}_m(\widehat{\mathcal{X}}) \leqslant R_m} \left\| \mathcal{X} - \widehat{\mathcal{X}} \right\|_F$

Truncation of SVD is optimal, but truncation of HOSVD is not

Iterative algorithm for finding optimal approximation – HOOI

# Higher-Order SSA, MSSA, ESPRIT

**Input:** time series X, window length: $(I, L)$ for single-channel or $L$ for multi-channel, signal ranks $(r_1, r_2, r_3)$, $d$ — estimation dimension for HO-ESPRIT.

1. **Embedding**.

| | |
|---|---|
| Single-channel | $\mathsf{X} \mapsto \mathcal{T}^{(I,L)}_{\mathrm{T-SSA}}(\mathsf{X}) = \mathcal{X}$ |
| Multi-channel | $\mathsf{X} \mapsto \mathcal{T}^{(L)}_{\mathrm{T-MSSA}}(\mathsf{X}) = \mathcal{X}$ |

2. **Decomposition & Approximation**. Using $(r_1, r_2, r_3)$
   $\mathcal{X} \mapsto \mathrm{Trunc}(\mathrm{HOSVD}(\mathcal{X})) = \widetilde{\mathcal{S}}$ or $\mathcal{X} \mapsto \mathrm{HOOI}(\mathcal{X}) = \widetilde{\mathcal{S}}$

3. **Reconstruction or Estimation**.
   - **Reconstruction**. $\mathsf{S} = (\mathcal{T})^{-1}\left(\Pi_{\mathcal{H}_\mathrm{T}}(\widetilde{\mathcal{S}})\right)$, $\Pi_{\mathcal{H}_\mathrm{T}}$ — projector onto the space of Hankel tensors
   - **Estimation**. Finding eigenvalues $z_j$ of matrix $\underline{\mathbf{U}}^{-}\overline{\mathbf{U}}$, where $\mathbf{U} = \left[U_1^{(d)} : U_2^{(d)} : \ldots : U_{r_d}^{(d)}\right]$. From $z_j = \exp(\alpha_j + \mathrm{i}2\pi\omega_j)$ damping factors $\alpha_j$ and frequencies $\omega_j$ of the signal can be found