Shaped extensions of Singular Spectrum Analysis

Alex Shlemov and Nina Golyandina

Saint-Petersburg State University, Russia
Faculty of Mathematics and Mechanics,
Department of Statistical Modelling

MTNS 2014, Groningen, The Netherlands,
June 2, 2015

Outline

Outline

Common scheme of Singular Spectrum Analysis (SSA)

Input: time series or array (or something else) $\mathbb{X} \in \mathsf{X} \sim \mathsf{R}^N$
Parameter: embedding operator $\mathcal{T} \colon \mathsf{X} \hookrightarrow \mathsf{R}^{L \times K}$, injective, linear

Decomposition

1. Embedding
   Trajectory matrix $\mathbf{X} = \mathcal{T}(\mathbb{X})$, structured (i.e. $\mathbf{X} \in \mathsf{H} = \operatorname{image} \mathcal{T}$)
2. Singular Value Decomposition (SVD)
   $$\mathbf{X} = \sum_{k=1}^{d} \sqrt{\lambda_k} U_k V_k^{\mathrm{T}}, \ \{U_k\}_{k=1}^{d} \text{ and } \{V_k\}_{k=1}^{d} \text{ are orthonormal}$$

Reconstruction

1. Grouping: $\{1, \dots, d\} = I_1 \sqcup \dots \sqcup I_c$
   $\mathbf{X} = \mathbf{X}_{I_1} + \dots + \mathbf{X}_{I_c}$, where $\mathbf{X}_I = \sum_{k \in I} \sqrt{\lambda_k} U_k V_k^{\mathrm{T}}$
2. Projection
   Matrix $\mathbf{X}_I \xrightarrow{\mathcal{H} = \Pi^{(\mathsf{H})}}$ structured matrix $\widetilde{\mathbf{X}}_I \xrightarrow{\mathcal{T}^{-1}}$ reconstructed $\widetilde{\mathbb{X}}^{(I)}$

Output: $\mathbb{X} = \widetilde{\mathbb{X}}^{(I_1)} + \dots + \widetilde{\mathbb{X}}^{(I_c)}$

Particular SSA cases: Classical 1D-SSA

Input: time series $\mathbb{X} = (x_1, x_2, \dots, x_N)$
Parameter: L, window length; $1 < L < N$; $K = N - L + 1$

The embedding operator:

$$\mathcal{T}(\mathbb{X}) = \mathcal{T}_L(\mathbb{X}) := [X_1 : \dots : X_K] = \begin{pmatrix} x_1 & x_2 & \dots & x_{K-1} & x_K \\ x_2 & \iddots & \dots & \iddots & x_{K+1} \\ \vdots & \vdots & & \vdots & \vdots \\ x_{L-1} & \iddots & \dots & \iddots & x_{N-1} \\ x_L & x_{L+1} & \dots & x_{N-1} & x_N \end{pmatrix},$$

where the columns X_j are lagged subvectors of length L:

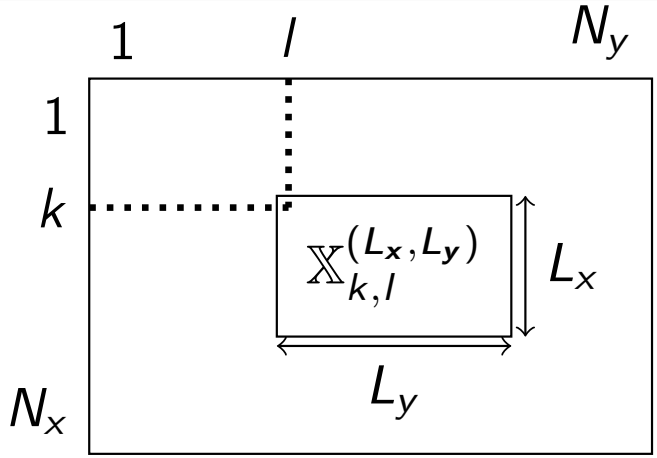
$$X_j = (x_j, x_{j+1}, \dots, x_{j+L-1})^{\mathrm{T}}$$

- $\mathsf{H} = \operatorname{image} \mathcal{T}_L$ is the set of Hankel matrices
- Projection $\Pi^{(\mathsf{H})}$ is diagonal averaging (or "hankelization")

Particular SSA cases: 2D-SSA

Input: 2D array $\mathbb{X} = \begin{pmatrix} x_{1,1} & \dots & x_{1,N_y} \\ \vdots & & \vdots \\ x_{N_x,1} & \dots & x_{N_x,N_y} \end{pmatrix}$

Parameters: window sizes L_x, L_y;
$1 < L_x < N_x$; $1 < L_y < N_y$;
$K_x = N_x - L_x + 1$; $K_y = N_y - L_y + 1$;



The embedding operator:

$$\mathcal{T}_{L_x,L_y}(\mathbb{X}) := \mathbf{X} = [X_1 : \dots : X_{K_xK_y}] \in \mathrm{R}^{L_xL_y \times K_xK_y},$$

where the columns X_j are vectorizations of shifted $L_x \times L_y$ subarrays:

$$X_{k+(l-1)K_x} = \mathrm{vec}(\mathbb{X}_{k,l}^{(L_x,L_y)})$$

- $\mathrm{H} = \mathrm{image}\,\mathcal{T}_{L_x \times L_y}$ is the set of Hankel-block-Hankel matrices
- Projection $\Pi^{(\mathrm{H})}$ is "block-hankelization" (also averaging)

Note: $\mathrm{R}^{L_x \times L_y} \sim \mathrm{R}^{L_xL_y}$, *thus any singular vector U can be considered as $L_x \times L_y$ array $\mathbb{U}$*

2D-SSA example: Gentlemen

Eigenvectors

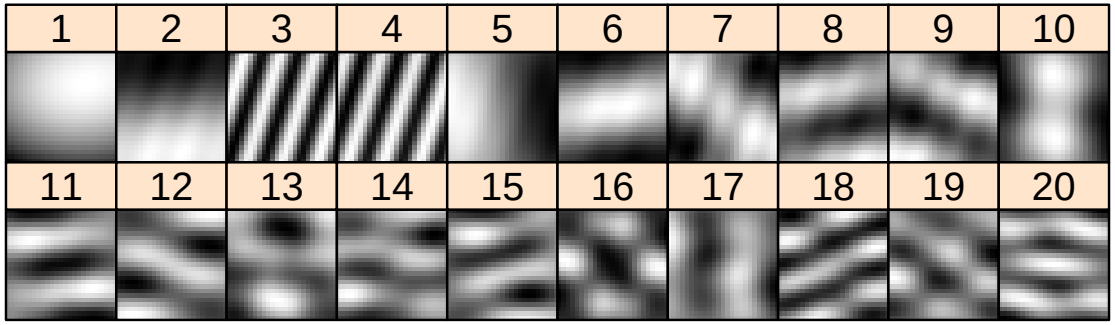


Figure: Gentlemen, 2D-SSA: Eigenarrays, $(L_x, L_y) = (25, 25)$

Reconstructions

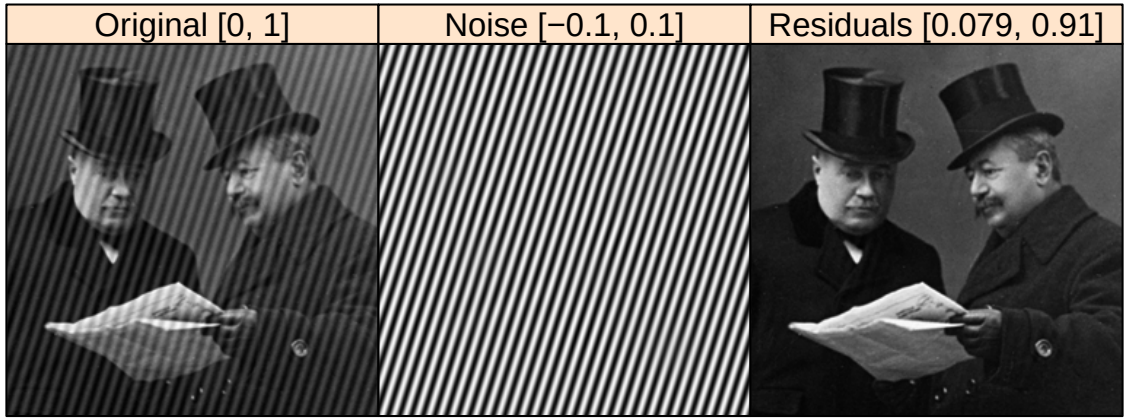


Figure: Gentlemen, 2D-SSA: Separated periodic noise, $(N_x, N_y) = (256, 256)$, $(L_x, L_y) = (25, 25)$, $I = \{3, 4\}$

Outline

Problem: non-rectangular arrays & edge effects

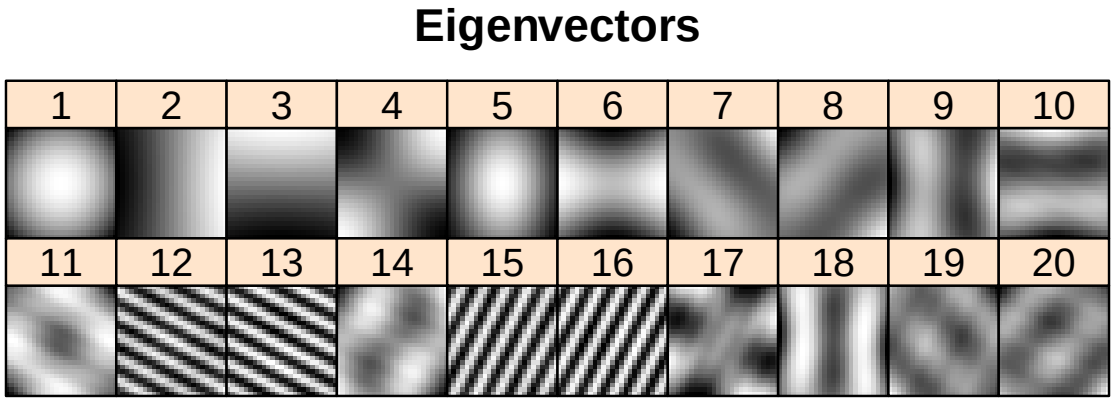


Figure: Mars, 2D-SSA: Eigenarrays, $(L_x, L_y) = (25, 25)$

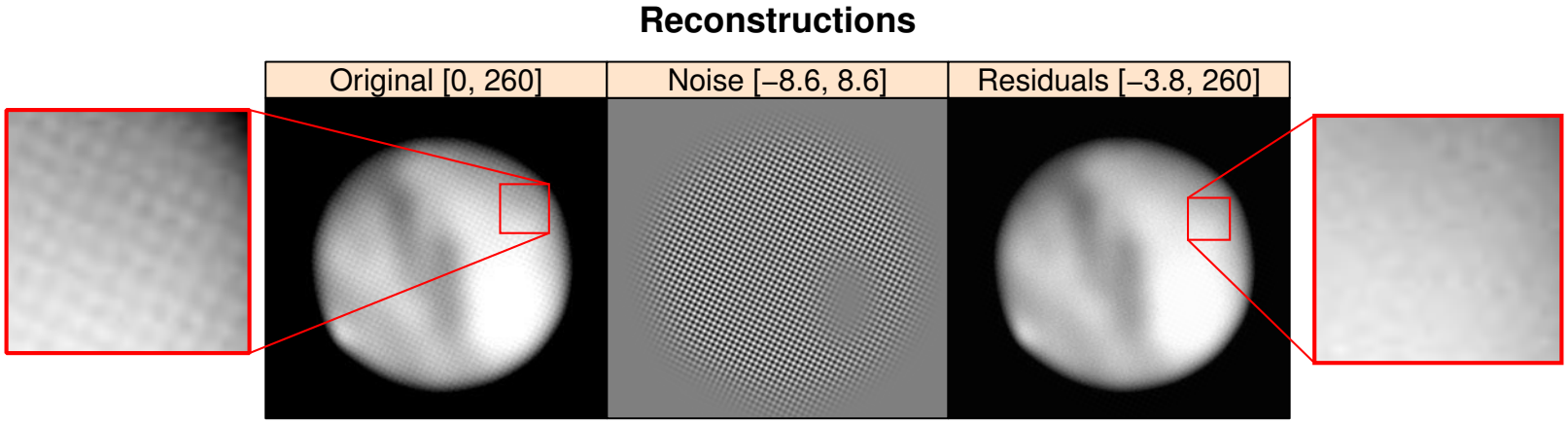
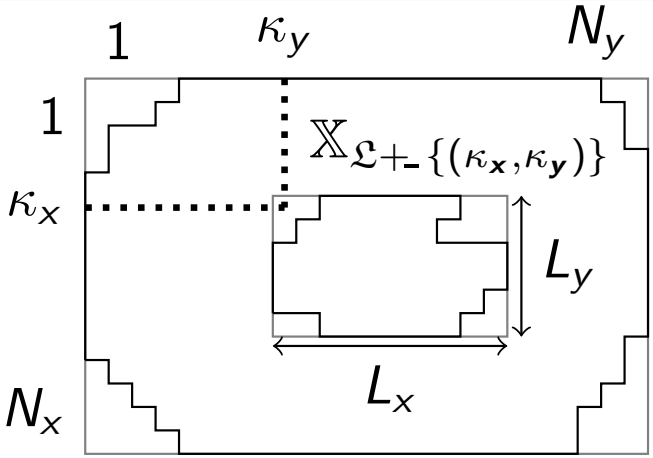


Figure: Mars, 2D-SSA: Separated periodic noise, $(N_x, N_y) = (258, 275)$, $(L_x, L_y) = (25, 25)$, $I = \{12, 13, 15, 16\}$

Let's develop "shaped" SSA, by analogy with shaped filters

New method: Shaped 2D-SSA

Input: $\mathfrak{N}$-shaped array $\mathbb{X} = \mathbb{X}_{\mathfrak{N}} = (x_\eta)_{\eta\in\mathfrak{N}}$, where $\mathfrak{N} = \{\eta_1, \dots, \eta_N\} \subseteq \{1, \dots, N_x\} \times \{1, \dots, N_y\}$
Parameter: window shape $\mathfrak{L} = \{\ell_1, \dots, \ell_L\} \subseteq$
$\subseteq \{1, \dots, L_x\} \times \{1, \dots, L_y\}$



The embedding operator:

$$\mathcal{T}_{\mathfrak{L}}(\mathbb{X}) := \mathbf{X} = [X_1 : \cdots : X_K] \in \mathsf{R}^{L\times K},$$

where columns $X_j = (x_{\ell_i +\!\!- \kappa_j})_{i=1}^{L}$ are vectorizations of shifted $\mathfrak{L}$-shaped arrays $\mathbb{X}_{\mathfrak{L}+\!\!-\{\kappa_j\}} = (x_\eta)_{\eta\in\mathfrak{L}+\!\!-\{\kappa_j\}}$

And $\mathfrak{K}$ is the set of all possible origin positions for $\mathfrak{L}$-shaped windows:

$$\mathfrak{K} = \{\kappa_1, \dots, \kappa_K\} = \{\kappa \in \mathsf{N}^2 \mid \mathfrak{L} +\!\!- \{\kappa\} \subseteq \mathfrak{N}\},$$

where $\cdot +\!\!- \eta$ means shape shift by vector η: $\mathfrak{A} +\!\!- \eta := \{\alpha + \eta - 1 \mid \alpha \in \mathfrak{A}\}$

- $\mathsf{H} = \operatorname{image} \mathcal{T}_{\mathfrak{L}}$ is the set of quasi-Hankel matrices
- Projection $\Pi^{(\mathsf{H})}$ is "quasi-hankelization" (again, averaging)

Note: $\mathsf{R}^{\mathfrak{L}} \sim \mathsf{R}^{L}$, *thus any singular vector U can be considered as $\mathfrak{L}$-shaped subarray $\mathbb{U}$*

Real life examples: Mars

Eigenvectors

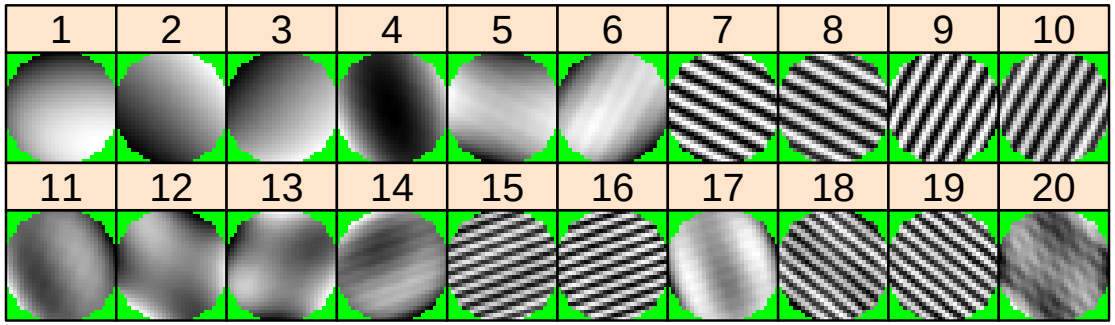


Figure: Mars, ShSSA: Eigenarrays, $\mathfrak{L}$ is circle of radius 15

Reconstructions

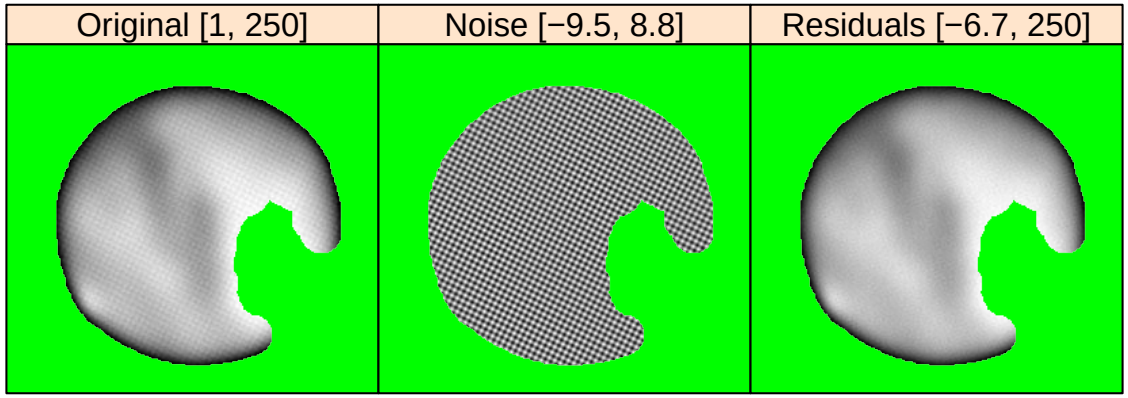


Figure: Mars: Separated period noise, $\mathfrak{L}$ is circle of radius 15, $I = \{7, 8, 9, 10\}$

Comparison: shaped vs non-shaped

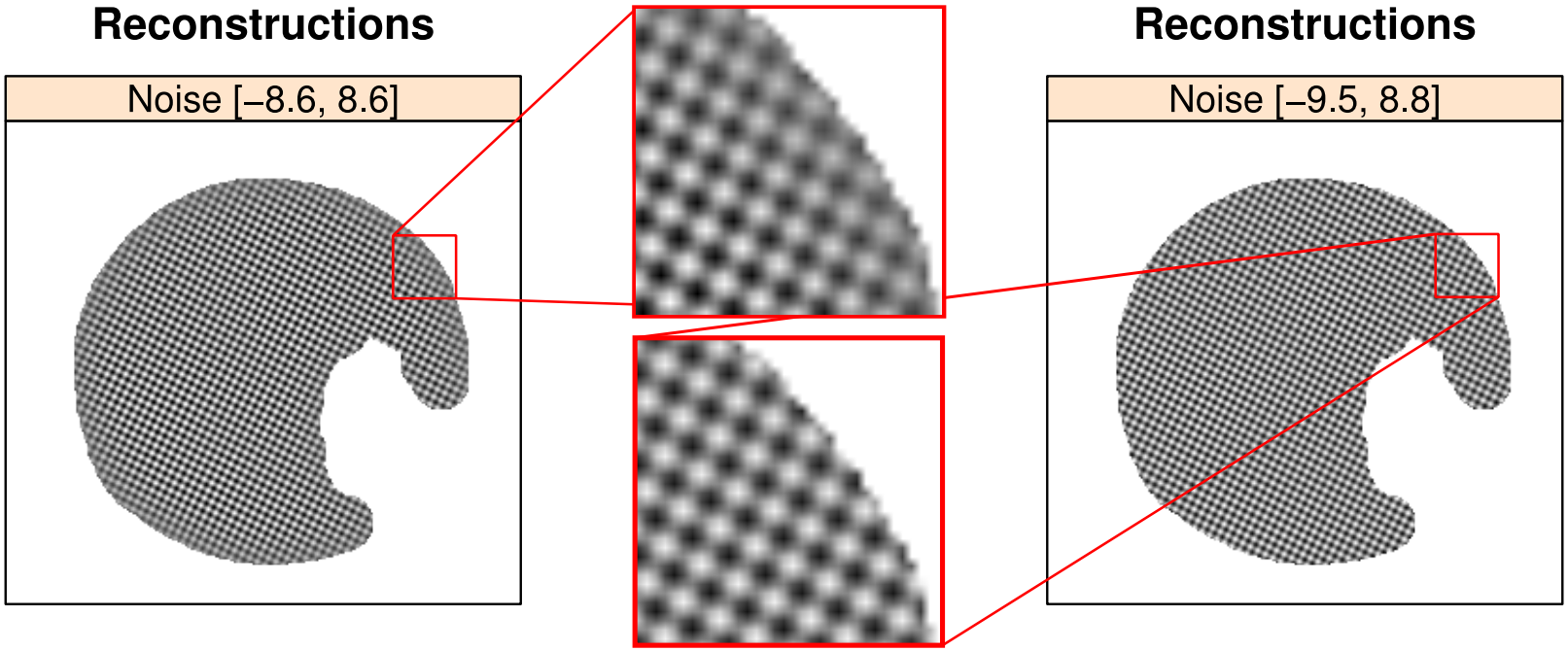


Figure: Left: cutted image of periodic noise from rectangular SSA; right: image of periodic noise from Shaped SSA

Outline

Problem: circular topology

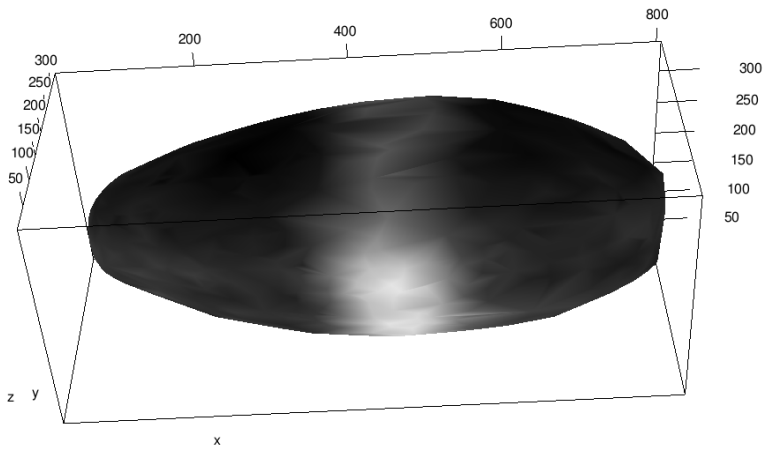
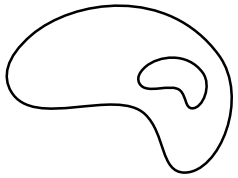
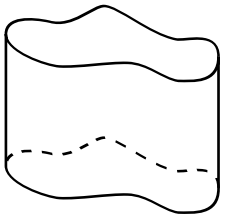


Figure: Drosophila, 3d intensity plot, "Krüppel" gene

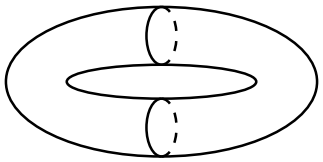
Three topology cases:



(a) Planar topology ($T_x = T_y = \infty$)



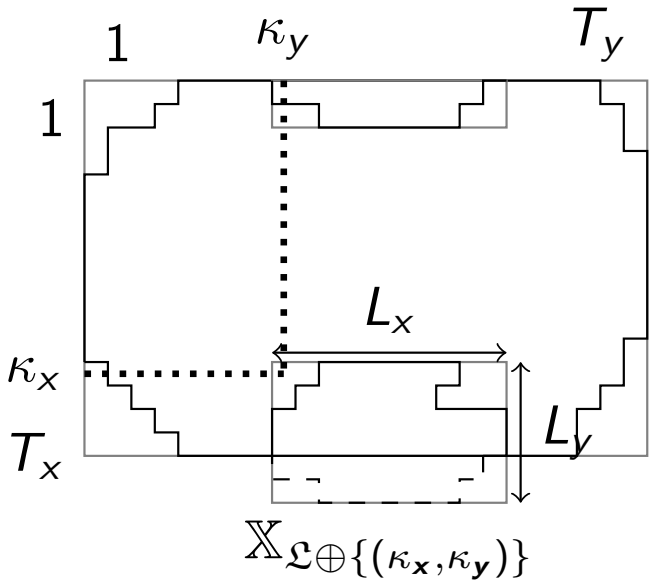
(b) Cylindrical topology ($T_x \neq \infty$; $T_y = \infty$)



(c) Toroidal topology ($T_x \neq \infty$; $T_y \neq \infty$)

New method: Circular Shaped 2D-SSA

Input: topology characteristics $T_x, T_y \in \mathsf{N} \cup \{\infty\}$, $\mathfrak{N}$-shaped array $\mathbb{X} = \mathbb{X}_{\mathfrak{N}} = (x_\eta)_{\eta \in \mathfrak{N}}$, where $\mathfrak{N} = \{\eta_1, \dots, \eta_N\} \subseteq \{1, \dots, T_x\} \times \{1, \dots, T_y\}$
Parameter: window shape $\mathfrak{L} = \{\ell_1, \dots, \ell_L\} \subseteq$
$\subseteq \{1, \dots, L_x\} \times \{1, \dots, L_y\}$



The embedding operator:

$$\mathcal{T}_{\mathfrak{L}}(\mathbb{X}) := \mathbf{X} = [X_1 : \cdots : X_K] \in \mathsf{R}^{L \times K},$$

where columns $X_j = (x_{\ell_i +\!\!- \kappa_j})_{i=1}^{L}$ are vectorizations of shifted $\mathfrak{L}$-shaped arrays

And $\mathfrak{K}$ is the set of all possible origin positions for $\mathfrak{L}$-shaped windows:

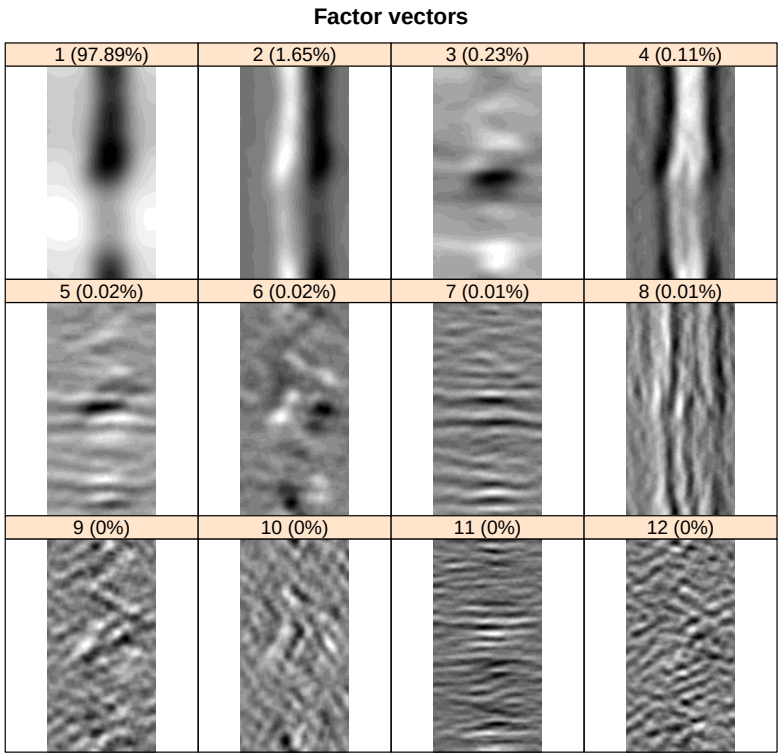
$$\mathfrak{K} = \{\kappa_1, \dots, \kappa_K\} = \{\kappa \in \mathsf{N}^2 \mid \mathfrak{L} \oplus \{\kappa\} \subseteq \mathfrak{N}\},$$

where $\cdot \oplus \eta$ means cycled shape shift by vector η:

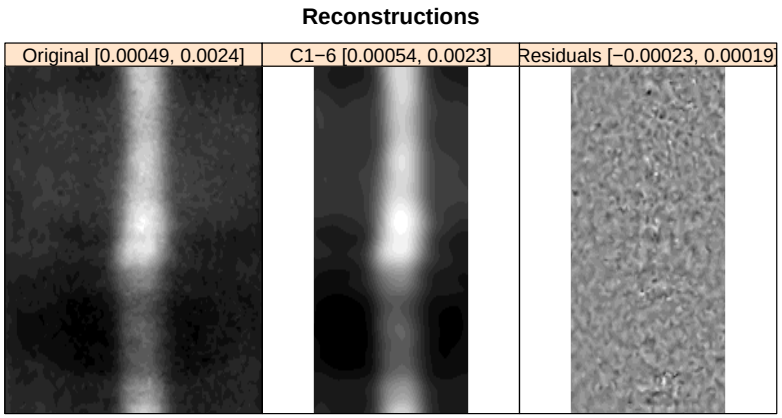
$$\mathfrak{A} \oplus \eta := \{((\alpha_x + \eta_x - 2) \bmod T_x + 1, (\alpha_y + \eta_y - 2) \bmod X_y + 1) \mid \alpha \in \mathfrak{A}\}$$

- $\mathsf{H} = \operatorname{image} \mathcal{T}_{\mathfrak{L}}$ is the set of partially circulant quasi-Hankel matrices
- Projection $\Pi^{(\mathsf{H})}$ is "quasi-hankelization" (and again, averaging)

Real life example: drosophila embryo



(d) Drosophila, factor vectors



(e) Drosophila, reconstruction and residuals

Figure: Drosophila, Circular SSA, cylindrical case ($T_x = N_x$ and $T_y = \infty$).
The data are downloaded from the BDTNP archive,
the file "v5_s11643-28no06-04.pca", gene "Krüppel".
Middle part from 20% to 80% of the embryo length was processed.
Interpolation step is 0.5% and both window sizes are 10% of the embryo length

Outline

2D-ESPRIT for shaped array

Parametric model: corrupted finite-rank array:
$(\mathbb{S})_{l,n} = \sum_{k=1}^{r} A_k \mu_k^l \nu_k^n, \quad A_k, \mu_k, \nu_k \in \mathrm{C}\backslash\{0\}$
$\mathbb{X} = \mathbb{S} + \mathbb{R}$, where $\mathbb{R}$ is noise or another signal

Problem: estimate nonlinear parameters $\{(\mu_k, \nu_k)\}_{k=1}^{r}$

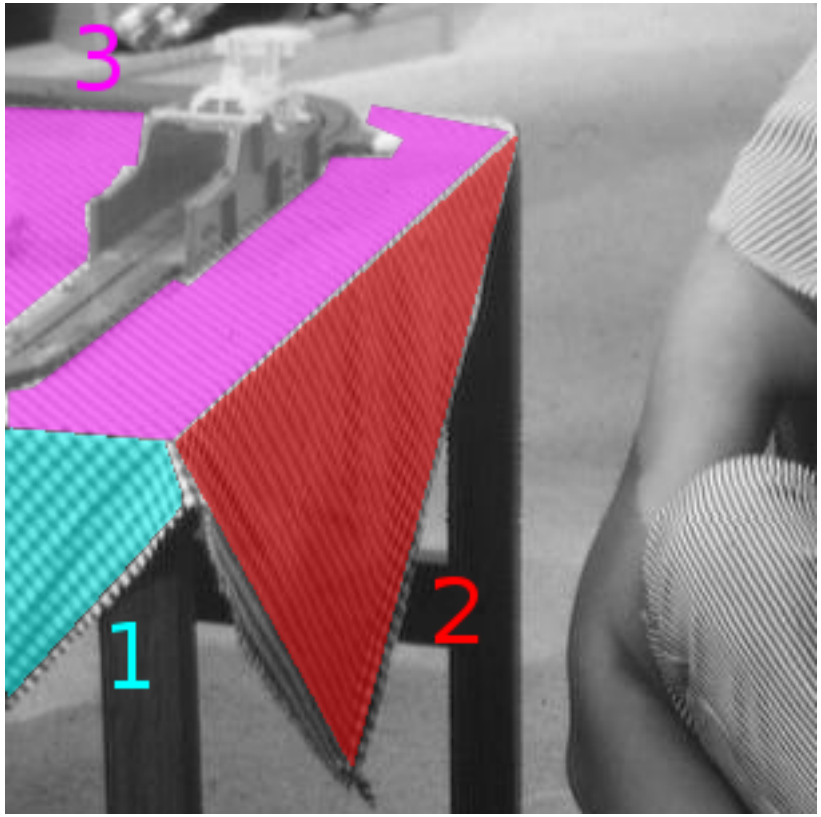
Algorithm rectangular 2D-ESPRIT (Rouquette and Najim, 2001):

1. **Estimation of the signal subspace:** Get the basis $\mathbb{U}_{i_1}, \ldots, \mathbb{U}_{i_r}$ of an estimate of signal subspace (e.g. from SSA)
2. **Construction of truncated matrices:**
   $\mathbf{P}_x = [\mathrm{vec}\,(\underline{\mathbb{U}_{i_1}}) : \cdots : \mathrm{vec}\,(\underline{\mathbb{U}_{i_r}})]$, $\mathbf{Q}_x = [\mathrm{vec}\,(\overline{\mathbb{U}_{i_1}}) : \cdots : \mathrm{vec}\,(\overline{\mathbb{U}_{i_r}})]$,
   $\mathbf{P}_y = [\mathrm{vec}\,(\mathbb{U}_{i_1}|) : \cdots : \mathrm{vec}\,(\mathbb{U}_{i_r}|)]$, $\mathbf{Q}_y = [\mathrm{vec}\,(|\mathbb{U}_{i_1}) : \cdots : \mathrm{vec}\,(|\mathbb{U}_{i_r})]$,
   where $(\underline{\cdot})$, $(\overline{\cdot})$, $(\cdot|)$, $(|\cdot)$ denote the array without last row, first row last column and first column respectively
3. **Construction of shift matrices:** Approximate solution of $\mathbf{P}_x\mathbf{M}_x \approx \mathbf{Q}_x$ and $\mathbf{P}_y\mathbf{M}_y \approx \mathbf{Q}_y$ in LS- or TLS-sense
4. **Estimation of the parameters:** EVD-decomposition of $\mathbf{M}_x$ and $\mathbf{M}_y$, eigenvalues $\widehat{\mu}_i$ and $\widehat{\nu}_j$ is an estimates of the parameters

This talk: from rectangular to shaped arrays

Real life example: Barbara's cloth

(a) Barbara, whole image



(b) Barbara, table with enumerated subimages

Real life example: Barbara's cloth

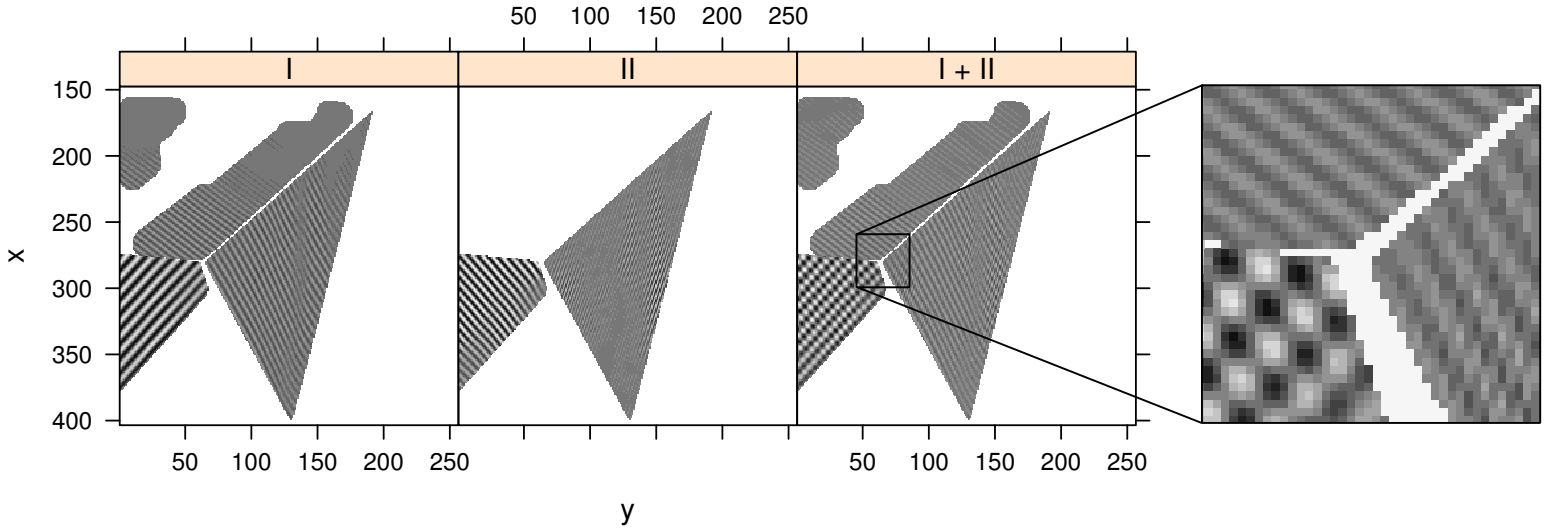


Figure: Barbara's table: reconstructions of texture.
$I_{\mathrm{I}} = \{2, 3\}$, $I_{\mathrm{II}} = \{4, 5\}$ (if present)

Table: Barbara, ESPRIT periods and rates

Part	t_x	t_y	Width	$\alpha,^\circ$	Rate_x	Rate_y
1	8.1	6.9	5.2	50	−0.0066	−0.0061
1	11.0	10.1	7.4	47	−0.0022	−0.0064
2	9.7	5.0	4.4	63	−0.0031	−0.0075
2	7.2	2.6	2.5	70	0.0025	0.017
3	5.1	6.8	4.1	37	0.0054	−0.0046

Outline

Efficient implementation

Main ideas:

1. The truncated SVD calculated by Lanczos methods is used, since only a number of leading SVD components correspond to the signal (ν-`TRLan` and `PROPACK` libraries are used)
2. In Lanczos methods, only multiplication of $\mathbf{X}$ (and $\mathbf{X}^{\mathrm{T}}$) by a vector is needed. Due to quasi-Hankel structure it can be implemented via FFT (`FFTW` library is used)
3. At the Reconstruction step, hankelization or quasi-hankelization of a matrix of rank 1 can be also implemented via FFT

Total complexity: $\mathcal{O}(rN \log N)$

`R` **implementation:**
`R` package "Rssa": `http://cran.r-project.org/web/packages/Rssa/`

Article:
N. Golyandina, A. Korobeynikov, A. Shlemov and K. Usevich (2014): *"Multivariate and 2D Extensions of Singular Spectrum Analysis with the* Rssa *Package"*, accepted to "Journal of Statistical Software"

Outline

Conclusions

- An extension of SSA (shaped SSA) was proposed for decomposition of shaped images including circular and toroidal topology
- An extension of 2D-ESPRIT method for parameter estimation in the signal-plus-noise model was described
- An efficient publicly available implementation is provided
- The algorithms and their implementation can be easily extended to higher dimensions

Thanks for your attention!