

Санкт-Петербургский государственный университет

Прикладная математика и информатика

Отчет по учебной практике 4 (научно-исследовательской работе)

МОДИФИКАЦИИ МЕТОДА АНАЛИЗА СИНГУЛЯРНОГО СПЕКТРА ДЛЯ  
АНАЛИЗА ВРЕМЕННЫХ РЯДОВ: CIRCULANT SSA И GENERALIZED SSA

Выполнил:

Погребников Николай Вадимович  
группа 21.Б04-мм

Научный руководитель:

д. ф.-м. н., доц.

Голяндина Нина Эдуардовна

Кафедра Статистического Моделирования

Санкт-Петербург

2025

# Содержание

<b>1 Введение</b>	<b>3</b>
<b>2 Метод Singular spectrum analysis (SSA)</b>	<b>5</b>
2.1 Алгоритм метода SSA . . . . .	5
2.2 Свойства SSA . . . . .	6
<b>3 Метод Generalized singular spectrum analysis (GSSA)</b>	<b>10</b>
3.1 Алгоритм метода GSSA . . . . .	10
3.2 Свойства GSSA . . . . .	11
<b>4 Сравнение SSA и GSSA</b>	<b>13</b>
4.1 Линейные фильтры . . . . .	13
4.2 Отделение сигнала от шума . . . . .	15
4.3 Фильтры в различных точках . . . . .	15
<b>5 Метод Circulant singular spectrum analysis (CiSSA)</b>	<b>18</b>
5.1 Алгоритм метода CiSSA . . . . .	18
5.2 Свойства . . . . .	22
5.3 Обзор литературы . . . . .	23
<b>6 Различия SSA, CiSSA и Фурье</b>	<b>25</b>
6.1 Автоматическая группировка . . . . .	25
6.2 Собственные пространства . . . . .	25
6.3 Точная разделимость . . . . .	25
6.4 Асимптотическая разделимость . . . . .	26
6.5 Выделение тренда . . . . .	27
6.6 Отделение сигнала от шума . . . . .	27
6.7 Разделение непериодических составляющих между собой . . . . .	29
6.8 Преимущества и недостатки методов SSA, Фурье и CiSSA . . . . .	29
6.9 Проверка алгоритмов на реальных данных . . . . .	30
<b>7 Многомерные варианты базового SSA</b>	<b>34</b>
7.1 MSSA . . . . .	34
7.2 Группировка . . . . .	34
7.3 2d-ssa . . . . .	35
<b>8 Метод Functional singular spectrum analysis (FSSA)</b>	<b>35</b>
<b>9 Заключение</b>	<b>36</b>
<b>7 Список литературы</b>	

# 1 Введение

Временные ряды представляют собой упорядоченную последовательность данных, собранных или измеренных в хронологическом порядке. Они играют ключевую роль в анализе и прогнозировании различных явлений в таких областях, как экономика, финансы, климатология и медицина. Понимание эволюции этих явлений во времени критично для выявления тенденций, циклов и аномалий.

Для уточнения терминологии, следует отметить, что **временной ряд длины  $N$**  представляет собой упорядоченную конечную последовательность значений, которая записывается как  $\mathbf{X} = (x_1, \dots, x_N)$ , где  $N > 2$ ,  $x_i \in \mathbb{R}$ . Одним из основных аспектов анализа временных рядов является разделение их на составляющие компоненты. Среди таких компонентов важными являются **тренд**, который отражает медленно изменяющуюся долгосрочную динамику ряда, и **сезонность**, представляющая собой периодические колебания, вызванные повторяющимися факторами, такими как климатические или экономические циклы.

Для эффективного анализа и понимания структуры временных рядов разработаны различные методы, позволяющие разделить ряд на его компоненты. Существует два вида разделимости: **точная разделимость**, которая характеризует способность метода точно выделять отдельные компоненты ряда, и **асимптотическая разделимость**, которая описывается следующим образом:

**Определение 1.** Есть метод разделения ряда на компоненты с параметрами  $\Theta$ , ряд  $\mathbf{X} = \mathbf{X}^{(1)} + \mathbf{X}^{(2)}$ . Существуют такой фиксированный набор параметров  $\hat{\Theta}$  и последовательность  $L = L(N)$ ,  $N \rightarrow \infty$ , что при разделении ряда на компоненты этим методом,  $\hat{\mathbf{X}}^{(1)}$  является оценкой  $\mathbf{X}^{(1)}$ , при этом,  $\text{MSE}(\mathbf{X}^{(1)}, \hat{\mathbf{X}}^{(1)}) \rightarrow 0$ , где MSE — среднеквадратическая ошибка. Тогда ряды  $\mathbf{X}^{(1)}$  и  $\mathbf{X}^{(2)}$  называются асимптотически  $L(N)$ -разделимыми данным методом.

**Замечание 1.**  $\hat{\mathbf{X}}^{(2)} = \mathbf{X} - \hat{\mathbf{X}}^{(1)}$  является оценкой для  $\mathbf{X}^{(2)}$ , выполнено  $\text{MSE}(\mathbf{X}^{(2)}, \hat{\mathbf{X}}^{(2)}) \rightarrow 0$ .

Методы разделения временных рядов играют ключевую роль в выделении тренда, сезонности и других структурных компонентов, что позволяет глубже понять и моделировать временные зависимости.

В данной работе будут рассмотрены следующие постановки задачи разделения временных рядов:

1. Разделение временного ряда на компоненты, соответствующие определенным частотным диапазонам;
2. Разделение временного ряда на компоненты без привязки к частотным характеристикам, то есть в их исходном виде.

Анализ сингулярного спектра (**SSA** [4]) — метод, целью которого является разложение оригинального ряда на сумму небольшого числа интерпретируемых компонентов, таких как медленно изменяющаяся тенденция (тренд), колебательные компоненты (сезонность) и шум. Позволяет решать как задачу в формулировке 1, так и её обобщение, представленное в 2. При этом, базовый алгоритм метода **SSA** не требует стационарности ряда, знания модели тренда, а также сведений о наличии в ряде периодиках, а за счет своего аддитивного базиса позволяет подстраиваться под любой входной ряд.

В данном исследовании рассматриваются модификации **SSA**, предложенные другими авторами, а именно, **GSSA** [7] и **CiSSA** [1].

**GSSA** отличается от базового **SSA** тем, что он добавляет веса на определенном этапе алгоритма **SSA**. В некоторых случаях это может оказаться полезным, в других — повлиять на разделимость в худшую сторону. Это исследование раскрывает смысловую ценность **GSSA** с точки зрения линейных фильтров и отмечает ситуации, где такой алгоритм предпочтительнее стандартного **SSA**.

В алгоритме **CiSSA** предложено решение задачи разделения временного ряда на заранее известные компоненты (задача в постановке 1), отвечающие конкретным периодикам. За счет этого можно автоматически группировать компоненты по частотам, однако именно поэтому алгоритм лишается адаптивности, которая имеется в **SSA**.

Целью работы является описание модификаций в контексте теории **SSA** и на этой основе сравнение методов по теоретическим свойствам и численно.

Далее кратко опишем структуру работы. В разделе 2 рассматривается базовый метод **SSA** и его ключевые свойства. В секции 3 показан алгоритм **GSSA**. В следующем разделе 5 представлен метод **CiSSA**, также с описанием его основных характеристик. Раздел ?? посвящен сравнению методов **SSA**, **GSSA**, разложения Фурье и **CiSSA** на модельных и реальных примерах. В заключительной секции 9 подведены основные итоги исследования.

В предыдущей научно-исследовательской работе было проведено сравнение алгоритмов **SSA** и **CiSSA** в контексте их способности к разделению временных рядов. В рамках текущего исследования продолжено изучение этих методов, а также впервые рассмотрен алгоритм **GSSA**, изучены его достоинства и недостатки перед **SSA**. Реализации алгоритмов были написаны на языке R.

## 2 Метод Singular spectrum analysis (SSA)

Рассмотрим базовый метод сингулярного спектрального анализа [4].

### 2.1 Алгоритм метода SSA

Пусть  $N > 2$ , вещественнозначный временной ряд  $\mathbf{X} = (x_1, \dots, x_N)$  длины  $N$ . Базовый алгоритм **SSA** состоит из четырех шагов.

#### 2.1.1 Вложение

Параметром этого шага является  $L$  — некоторое целое число (длина окна),  $1 < L < N$ . Строится  $L$ -траекторная матрица  $\mathbf{X}$ , состоящая из  $K = N - L + 1$  векторов вложения:

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & x_3 & \dots & x_K \\ x_2 & x_3 & x_4 & \dots & x_{K+1} \\ x_3 & x_4 & x_5 & \dots & x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & x_{L+2} & \dots & x_N \end{pmatrix}. \quad (1)$$

Полезным свойством является то, что матрица  $\mathbf{X}$  имеет одинаковые элементы на антидиагоналях. Таким образом,  $L$ -траекторная матрица является ганкелевой.

#### 2.1.2 Сингулярное разложение (SVD)

Результатом этого шага является сингулярное разложение (Singular Value Decomposition, **SVD**) траекторной матрицы ряда.

Пусть  $\mathbf{S} = \mathbf{XX}^T$ ,  $\lambda_1, \dots, \lambda_L$  — собственные числа матрицы  $\mathbf{S}$ , взятые в неубывающем порядке, и  $U_1, \dots, U_L$  — ортонормированная система собственных векторов, соответствующих собственным числам матрицы  $\mathbf{S}$ .

Определим  $d = \max\{i : \lambda_i > 0\}$  и  $V_i = \mathbf{X}^T U_i / \sqrt{\lambda_i}$ . Тогда сингулярным разложением называется представление матрицы в виде:

$$\mathbf{X} = \mathbf{X}_1 + \dots + \mathbf{X}_d = \sum_{i=1}^d \sqrt{\lambda_i} U_i V_i^T. \quad (2)$$

Набор  $(\sqrt{\lambda_i}, U_i, V_i^T)$  называется  $i$ -й собственной тройкой разложения (2).

#### 2.1.3 Группировка

На основе разложения (2) производится процедура группировки, которая делит все множество индексов  $\{1, \dots, d\}$  на  $m$  непересекающихся подмножеств  $I_1, \dots, I_m$ . Это разбиение является параметром шага группировки.

Пусть  $I = \{i_1, \dots, i_p\}$ , тогда  $\mathbf{X}_I = \mathbf{X}_{i_1} + \dots + \mathbf{X}_{i_p}$ . Такие матрицы вычисляются для каждого  $I = I_1, \dots, I_m$ . В результате получаются матрицы  $\mathbf{X}_{I_1}, \dots, \mathbf{X}_{I_m}$ . Тем самым разложение (2) может быть записано в сгруппированном виде:

$$\mathbf{X} = \mathbf{X}_{I_1} + \dots + \mathbf{X}_{I_m}.$$

#### 2.1.4 Диагональное усреднение

Пусть  $\mathbf{Y}$  — матрица размерности  $L \times K$ .  $L^* = \min(L, K)$ ,  $K^* = \max(L, K)$ . Диагональное усреднение переводит матрицу  $\mathbf{Y}$  в временной ряд  $g_1, \dots, g_N$ :

$$g_k = \begin{cases} \frac{1}{k+1} \sum_{m=1}^{k+1} y_{m,k-m+2}^* & \text{для } 1 \leq k < L^*, \\ \frac{1}{L^*} \sum_{m=1}^{L^*} y_{m,k-m+2}^* & \text{для } L^* \leq k < K^* + 1, \\ \frac{1}{N-k} \sum_{m=k-K^*+2}^{N-K^*+1} y_{m,k-m+2}^* & \text{для } K^* + 1 \leq k \leq N. \end{cases}$$

Применяя данную операцию к матрицам  $\mathbf{X}_{I_1}, \dots, \mathbf{X}_{I_m}$ , получаются  $m$  новых рядов:  $\mathbf{X}_1, \dots, \mathbf{X}_m$ . Результатом данного шага и всего алгоритма является разложение временного ряда  $\mathbf{X} = \mathbf{X}_1 + \dots + \mathbf{X}_m$ .

## 2.2 Свойства SSA

### 2.2.1 Ранг ряда

Зафиксируем ряд  $\mathbf{X} = (x_1, \dots, x_N)$  длины  $N > 3$  и длину окна  $L$ .

Рассмотрим базовый **SSA**. В процессе процедуры вложения получаем последовательность векторов вложения:

$$\mathbf{X}_i^{(L)} = \mathbf{X}_i = (x_{i-1}, \dots, x_{i+L-2}), \quad i = 1, \dots, K,$$

$\mathcal{L}^{(L)} = \mathcal{L}^{(L)}(\mathbf{X}) \stackrel{\text{def}}{=} \text{span}(\mathbf{X}_1, \dots, \mathbf{X}_K)$  — траекторное пространство ряда  $\mathbf{X}$ . При этом, если  $\dim \mathcal{L}^{(L)} = \text{rank } \mathbf{X} = d$ , то будем говорить, что ряд  $\mathbf{X}$  имеет  $L$ -ранг  $d$  и записывать это как  $\text{rank}_L = d$ .

### 2.2.2 Точная разделимость

Пусть временной ряд  $\mathbf{X} = \mathbf{X}^{(1)} + \mathbf{X}^{(2)}$  и задачей является нахождение этих слагаемых. В результате базового алгоритма **SSA** при  $m = 2$  также получаем 2 ряда. Возникает вопрос: в каких случаях мы можем так выбрать параметр алгоритма  $L$  и так сгруппировать собственные тройки, чтобы получить исходные ряды без смешиваний? При выборе длины окна  $L$  каждый из рядов  $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \mathbf{X}$  порождает траекторную матрицу  $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \mathbf{X}$ .

**Определение 2.** Будем говорить, что ряды  $\mathbf{X}^{(1)}$  и  $\mathbf{X}^{(2)}$  слабо  $L$ -разделимы, если пространства, порожденные строками  $\mathbf{X}^{(1)}$  и  $\mathbf{X}^{(2)}$  соответственно, ортогональны. То же самое должно выполняться для столбцов [4].

Если выполняется условие слабой  $L$ -разделимости, тогда существует такое сингулярное разложение траекторной матрицы  $\mathbf{X}$  ряда  $\mathbf{X}$ , что его можно разбить на две части, являющиеся сингулярными разложениями траекторных матриц рядов  $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$  [4].

**Определение 3.** Будем говорить, что ряды  $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$  сильно  $L$ -разделимы, если они слабо  $L$ -разделимы и после процедуры **SVD** множества сингулярных чисел траекторных матриц рядов не имеют совпадений [4].

Если выполняется условие сильной  $L$ -разделимости, тогда любое сингулярное разложение траекторной матрицы  $\mathbf{X}$  ряда  $\mathbf{X}$  можно разбить на две части, являющиеся сингулярными разложениями траекторных матриц рядов  $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$  [4]. Это будет означать, что для разложения

ряда базовым методом **SSA** с  $m = 2$  и таким  $L$  будет выполняться  $\text{MSE}(\mathbf{X}^{(1)}, \hat{\mathbf{X}}^{(1)}) = 0$  (а значит и  $\text{MSE}(\mathbf{X}^{(2)}, \hat{\mathbf{X}}^{(2)}) = 0$ ).

Рассмотрим таблицу, в которой знаком + отмечены пары рядов, для которых существуют параметры функций и параметры метода  $L$  и  $K = N - L + 1$ , при которых они разделимы (точно разделимы). Данная таблица 1 и условия разделимости с доказательствами взяты из книги [4].

Таблица 1: Точная разделимость

	const	cos	exp	exp cos	ak+b
const	-	+	-	-	-
cos	+	+	-	-	-
exp	-	-	-	+	-
exp cos	-	-	+	+	-
ak+b	-	-	-	-	-

Отметим, что + в таблице 1 для  $\mathbf{X}_n^{(\cos_1)} = A_1 \cos(2\pi\omega_1 n + \varphi_1)$ ,  $\mathbf{X}_n^{(\cos_2)} = A_2 \cos(2\pi\omega_2 n + \varphi_2)$  достигается, если  $L\omega_1 \in \mathbb{N}$ ,  $K\omega_1 \in \mathbb{N}$  или  $L\omega_2 \in \mathbb{N}$ ,  $K\omega_2 \in \mathbb{N}$ ,  $\omega_1 \neq \omega_2$  [4].

Однако, по таблице 1 видно, что условия точной разделимости достаточно жесткие и вряд ли выполнимы в реальных задачах. Тогда появляется такое понятие, как асимптотическая разделимость.

### 2.2.3 Асимптотическая разделимость

Для любого ряда  $\mathbf{X}$  длины  $N$  определим  $\mathbf{X}_{i,j} = (x_{i-1}, \dots, x_{j-1})$ ,  $1 \leq i \leq j < N$ . Пусть  $\mathbf{X}^{(1)} = (x_0^{(1)}, \dots, x_{N-1}^{(1)})$ ,  $\mathbf{X}^{(2)} = (x_0^{(2)}, \dots, x_{N-1}^{(2)})$ . Тогда определим коэффициент корреляции следующим образом:

$$\rho_{i,j}^{(M)} = \frac{\left( \mathbf{X}_{i,i+M-1}^{(1)}, \mathbf{X}_{j,j+M-1}^{(2)} \right)}{\left\| \mathbf{X}_{i,i+M-1}^{(1)} \right\| \left\| \mathbf{X}_{j,j+M-1}^{(2)} \right\|}.$$

**Определение 4** ([4]). Ряды  $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$  называются  $\varepsilon$ -разделимыми при длине окна  $L$ , если

$$\rho^{(L,K)} \stackrel{\text{def}}{=} \max \left( \max_{1 \leq i,j \leq K} |\rho_{i,j}^{(L)}|, \max_{1 \leq i,j \leq L} |\rho_{i,j}^{(K)}| \right) < \varepsilon.$$

**Определение 5** ([4]). Если  $\rho^{(L(N), K(N))} \rightarrow 0$  при некоторой последовательности  $L = L(N)$ ,  $N \rightarrow \infty$ , то ряды  $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$  называются асимптотически  $L(N)$ -разделимыми.

Как можно заметить по таблице 2, для гораздо большего класса функций асимптотическая разделимость имеет место [4].

Таблица 2: Асимптотическая разделимость

	const	cos	exp	exp cos	ak+b
const	-	+	+	+	-
cos	+	+	+	+	+
exp	+	+	+	+	+
exp cos	+	+	+	+	+
ak+b	-	+	+	+	-

#### 2.2.4 Алгоритмы улучшения разделимости

Для **SSA** существуют алгоритмы улучшения разделимости. По заданному набору компонент, они позволяют более точно отделять временные ряды друг от друга. В данной работе будут использоваться методы EOSSA и FOSSA. Подробнее про них можно почитать в [3].

Кроме того, применение алгоритмов улучшения разделимости позволяет не только понизить ошибку разделения **SSA**, но и автоматически группировать компоненты в соответствии с заранее заданными частотами.

#### 2.2.5 SSA как линейный фильтр

Разложение временного ряда методом **SSA** можно интерпретировать как применение линейных фильтров. Для дальнейшего исследования введем следующие определения.

**Определение 6.** Рассмотрим бесконечный временной ряд  $\mathbf{X} = (\dots, x_{-1}, x_0, x_1, \dots)$ . Линейный конечный фильтр — это оператор  $\Phi$ , который преобразует временной ряд  $\mathbf{X}$  в новый по следующему правилу:

$$y_j = \sum_{i=-r_1}^{r_2} h_i x_{j-i}; \quad r_1, r_2 < \infty.$$

Набор коэффициентов  $h_i$  — импульсная характеристика фильтра.

Там, где не оговорено обратного, будем называть линейный конечный фильтр просто линейным фильтром.

**Определение 7.** Передаточная функция линейного фильтра  $\Phi$ :

$$H_\Phi(z) = \sum_{i=-r_1}^{r_2} h_i z^{-i}.$$

**Определение 8.** Амплитудно-частотная характеристика (АЧХ) линейного фильтра  $\Phi$ :

$$A_\Phi(\omega) = |H_\Phi(e^{i2\pi\omega})|.$$

АЧХ фильтра — это график или функция, которая показывает, как фильтр изменяет амплитуды (силу) разных частот входного сигнала.

**Определение 9.** Фазово-частотная характеристика ( $\Phi$ ЧХ) линейного фильтра  $\Phi$ :

$$\phi_\Phi(\omega) = \text{Arg}(H_\Phi(e^{i2\pi\omega})).$$

Посмотрим, как это выглядит для косинуса. Пусть исходный ряд  $X_{\cos} = \cos 2\pi\omega n$ . Тогда:

$$y_j = A_\Phi(\omega) \cos(2\pi\omega j + \phi_\Phi(\omega))$$

Теперь рассмотрим алгоритм **SSA** с точки зрения линейных фильтров [5]. Пусть  $\mathbf{X} = (x_1, \dots, x_N)$  — временной ряд длины  $N$ ,  $K = N - L + 1$ ,  $L^* = \min(L, K)$ . Пусть  $L$  будет длиной окна, а  $(\sqrt{\lambda}, U, V)$  — одной из собственных троек. Определим диагональную матрицу  $N \times N$ :

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

и матрицу  $K \times N$

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

Здесь  $U = (u_1, \dots, u_L)$  — собственный вектор матрицы  $\mathbf{S}$ .

**Теорема 1.** Компонента временного ряда  $\tilde{\mathbf{X}}$ , восстановленная с использованием собственной тройки  $(\sqrt{\lambda}, U, V)$ , имеет вид:

$$\tilde{\mathbf{X}}^T = \mathbf{D}^{-1} \mathbf{W}^T \mathbf{W} \mathbf{X}^T.$$

*Доказательство.* Доказательство можно найти в [5] (неплохо бы расписать).  $\square$

Таким образом, для восстановления методом **SSA** средних точек (индексы от  $L$  до  $K$ ) имеем следующий фильтр:

$$\tilde{x}_s = \sum_{j=-(L-1)}^{L-1} \left( \sum_{k=1}^{L-|j|} u_k u_{k+|j|}/L \right) x_{s-j}, \quad L \leq s \leq K. \quad (3)$$

Похожим образом можно переписать **SSA** через линейные фильтры для точек в начале и конце.

### 3 Метод Generalized singular spectrum analysis (GSSA)

В этом разделе описана модификация **SSA** на основе добавления определенных весов к строкам  $L$ -траекторной матрицы  $\mathbf{X}$  [7]. Это делается для уменьшения растекания частоты (spectral leakage). Авторы метода называют его обобщенным, поскольку базовый **SSA** является частным случаем **GSSA** с параметром  $\alpha = 0$ .

#### 3.1 Алгоритм метода GSSA

Алгоритм **GSSA** сильно схож с базовым **SSA**. Пусть  $N > 2$ , вещественнозначный временной ряд  $\mathbf{X} = (x_1, \dots, x_N)$  длины  $N$ . Фиксируется параметр  $\alpha \geq 0$ , отвечающий за веса:

$$\mathbf{w}^{(a)} = (w_1, w_2, \dots, w_L) = \left( \left| \sin \left( \frac{\pi n}{L+1} \right) \right| \right)^\alpha, \quad \text{для } n = 1, 2, \dots, L.$$

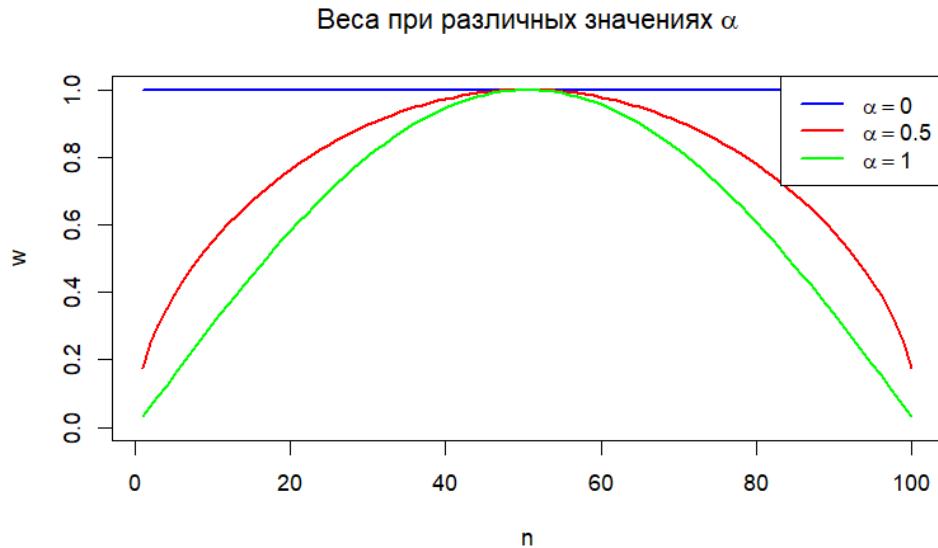


Рис. 1: График весов для различных значений  $\alpha$

##### 3.1.1 Вложение

$L$  — некоторое целое число (длина окна),  $1 < L < N$ . Строится  $L$ -траекторная матрица  $\mathbf{X}^{(\alpha)}$ :

$$\mathbf{X}^{(\alpha)} = \begin{pmatrix} w_1 x_1 & w_1 x_2 & w_1 x_3 & \dots & w_1 x_K \\ w_2 x_2 & w_2 x_3 & w_2 x_4 & \dots & w_2 x_{K+1} \\ w_3 x_3 & w_3 x_4 & w_3 x_5 & \dots & w_3 x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_L x_L & w_L x_{L+1} & w_L x_{L+2} & \dots & w_L x_N \end{pmatrix}. \quad (4)$$

##### 3.1.2 Сингулярное разложение (SVD)

Этот шаг такой же, как и в **SSA**, только матрица  $\mathbf{X}$  заменяется на  $\mathbf{X}^{(\alpha)}$ . Будем обозначать собственные тройки в этом случае так:  $(\sqrt{\lambda^{(\alpha)}}, U^{(\alpha)}, V^{(\alpha)})$ .

### 3.1.3 Группировка

В точности как в **SSA**. Тем самым, разложение может быть записано в сгруппированном виде:

$$\mathbf{X}^{(\alpha)} = \mathbf{X}_{I_1}^{(\alpha)} + \cdots + \mathbf{X}_{I_m}^{(\alpha)}.$$

### 3.1.4 Взвешенное диагональное усреднение

Поскольку траекторная матрица была изменена весами, то диагональное усреднение тоже будет зависеть от весов.

Пусть  $\mathbf{Y}$  — матрица размерности  $L \times K$ . Взвешенное диагональное усреднение переводит матрицу  $\mathbf{Y}$  в временной ряд  $g_1, \dots, g_N$ :

$$g_k = \begin{cases} \frac{1}{\sum_{n=1}^k w_n} \sum_{m=1}^{k+1} y_{m,k-m+2}^* & \text{для } 1 \leq k < L, \\ \frac{1}{\sum_{n=1}^L w_n} \sum_{m=1}^L y_{m,k-m+2}^* & \text{для } L \leq k < K+1, \\ \frac{1}{\sum_{n=k-K+1}^N w_n} \sum_{m=k-K+2}^{N-K+1} y_{m,k-m+2}^* & \text{для } K+1 \leq k \leq N. \end{cases}$$

Применяя данную операцию к матрицам  $\mathbf{X}_{I_1}^{(\alpha)}, \dots, \mathbf{X}_{I_m}^{(\alpha)}$ , получаются  $m$  новых рядов:  $\mathbf{X}_1^{(\alpha)}, \dots, \mathbf{X}_m^{(\alpha)}$ . Результатом данного шага и всего алгоритма является разложение временного ряда  $\mathbf{X}_1^{(\alpha)} + \cdots + \mathbf{X}_m^{(\alpha)} = \mathbf{X}^{(\alpha)}$ .

## 3.2 Свойства GSSA

### 3.2.1 Веса

Для минимизации эффекта спектрального размывания (spectral leakage), связанного с конечностью временного интервала наблюдений, к исходному ряду применялось оконное преобразование (tapering [9]). В качестве оконной функции используются степенные синус-косинусные веса (power-of-sine/cosine window).

Данное преобразование выполняет две ключевые функции:

1. Снижение краевых эффектов: умножение исходного ряда на убывающую к краям функцию  $w(t)$ ;
2. Сглаживание периодограммы: веса используются для усреднения значений периодограммы в частотной области.

Такой подход позволяет более точно отделять компоненты ряда друг от друга.

### 3.2.2 Ранг ряда

Зафиксируем ряд  $\mathbf{X} = (x_1, \dots, x_N)$  длины  $N > 3$  и длину окна  $L$ .

В секции 2.2.1 было введено понятие ранга ряда для базового **SSA**. Теперь рассмотрим **GSSA** и поймем, что для того же ряда  $\text{rank } \mathbf{X}^{(\alpha)} = \text{rank } \mathbf{X}$ , а значит, что для **GSSA** также применимы понятия  $L$ -ранга ряда. Из вида (4)  $\mathbf{X}^{(\alpha)}$  можно получить, что  $\mathbf{X}^{(\alpha)} = \text{diag}(w_1, w_2, \dots, w_L) \mathbf{X} = \text{diag}(\mathbf{w}^{(a)}) \mathbf{X}$ . Поскольку матрица  $\text{diag}(\mathbf{w}^{(a)})$  имеет ранг равный  $L$ , она диагональна, то и  $\text{rank } \mathbf{X}^{(\alpha)} = \text{rank } \text{diag}(\mathbf{w}^{(a)}) \mathbf{X} = \text{rank } \mathbf{X}$ .

### 3.2.3 GSSA как линейный фильтр

Аналогично **SSA**, метод **GSSA** можно переписать с помощью линейных фильтров. Пусть  $\mathbf{X} = (x_1, \dots, x_N)$  — временной ряд длины  $N$ ,  $K = N - L + 1$ ,  $L^* = \min(L, K)$ . Пусть  $L$  будет длиной окна, а  $(\sqrt{\lambda^{(\alpha)}}, U^{(\alpha)}, V^{(\alpha)})$  — одной из собственных троек. Определим диагональную матрицу  $N \times N$ :

$$\mathbf{D}^{(\alpha)} = \text{diag}(w_1, w_1 + w_2, \dots, \sum_{i=1}^{L^*-1} w_i, \sum_{i=1}^{L^*} w_i, \sum_{i=1}^{L^*} w_i, \dots, \sum_{i=1}^{L^*} w_i, \sum_{i=2}^{L^*} w_i, \dots, w_{L^*-1} + w_{L^*}, w_{L^*})$$

и две матрицы  $K \times N$ :

$$\mathbf{W}^{(\alpha)} = \begin{pmatrix} u_1^{(\alpha)} & u_2^{(\alpha)} & u_3^{(\alpha)} & \cdots & u_L^{(\alpha)} & 0 & \cdots & 0 & 0 & 0 \\ 0 & u_1^{(\alpha)} & u_2^{(\alpha)} & u_3^{(\alpha)} & \cdots & u_L^{(\alpha)} & 0 & \cdots & 0 & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & \cdots & \ddots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & u_1^{(\alpha)} & u_2^{(\alpha)} & u_3^{(\alpha)} & \cdots & u_L^{(\alpha)} & 0 & \vdots \\ 0 & 0 & \cdots & 0 & u_1^{(\alpha)} & u_2^{(\alpha)} & u_3^{(\alpha)} & \cdots & u_L^{(\alpha)} & 0 \\ 0 & 0 & 0 & \cdots & 0 & u_1^{(\alpha)} & u_2^{(\alpha)} & u_3^{(\alpha)} & \cdots & u_L^{(\alpha)} \end{pmatrix},$$

$$\mathbf{W}_w^{(\alpha)} = \begin{pmatrix} w_1 u_1^{(\alpha)} & w_2 u_2^{(\alpha)} & w_3 u_3^{(\alpha)} & \cdots & w_L u_L^{(\alpha)} & 0 & \cdots & 0 & 0 & 0 \\ 0 & w_1 u_1^{(\alpha)} & w_2 u_2^{(\alpha)} & w_3 u_3^{(\alpha)} & \cdots & w_L u_L^{(\alpha)} & 0 & \cdots & 0 & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & \cdots & \ddots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & w_1 u_1^{(\alpha)} & w_2 u_2^{(\alpha)} & w_3 u_3^{(\alpha)} & \cdots & w_L u_L^{(\alpha)} & 0 & \vdots \\ 0 & 0 & \cdots & 0 & w_1 u_1^{(\alpha)} & w_2 u_2^{(\alpha)} & w_3 u_3^{(\alpha)} & \cdots & w_L u_L^{(\alpha)} & 0 \\ 0 & 0 & 0 & \cdots & 0 & w_1 u_1^{(\alpha)} & w_2 u_2^{(\alpha)} & w_3 u_3^{(\alpha)} & \cdots & w_L u_L^{(\alpha)} \end{pmatrix}.$$

Здесь  $U = (u_1, \dots, u_L)$  — собственный вектор матрицы  $\mathbf{S}$ .

**Теорема 2.** Компонента временного ряда  $\tilde{\mathbf{X}}$ , восстановленная с использованием собственной тройки  $(\sqrt{\lambda^{(\alpha)}}, U^{(\alpha)}, V^{(\alpha)})$ , имеет вид:

$$\tilde{\mathbf{X}}^T = \mathbf{D}^{(\alpha)-1} \mathbf{W}^{(\alpha)T} \mathbf{W}_w^{(\alpha)T} \mathbf{X}^T.$$

*Доказательство.* Доказательство проводится аналогично доказательству теоремы 1.  $\square$

Таким образом, для восстановления методом **GSSA** средних точек (индексы от  $L$  до  $K$ ) имеем следующий фильтр:

$$\tilde{x}_s = \sum_{j=-(L-1)}^{L-1} \left( \sum_{k=1}^{L-|j|} u_k^{(\alpha)} u_{k+|j|}^{(\alpha)} w_k / \sum_{i=1}^L w_i \right) x_{s-j}, \quad L \leq s \leq K. \quad (5)$$

Похожим образом можно переписать **GSSA** через линейные фильтры для точек в начале и конце.

## 4 Сравнение SSA и GSSA

В данном разделе сравниваются алгоритмы базового **SSA** и **GSSA** с параметром  $\alpha \neq 0$ . Все вычисления, а также код метода **GSSA** можно найти в github репозитории [8].

### 4.1 Линейные фильтры

Чтобы понять их принципиальное отличие, рассмотрим методы с точки зрения линейных фильтров: по представлениям (3) и (5) можно построить амплитудно-частотные характеристики.

Рассмотрим временной ряд  $X = \sin\left(\frac{2\pi}{12}x\right)$ ,  $N = 96 \cdot 2 - 1$ ,  $L = 48$ . Построим АЧХ для  $\alpha$  равных 0 (базовый **SSA**),  $\frac{1}{2}$ , 1, 2:

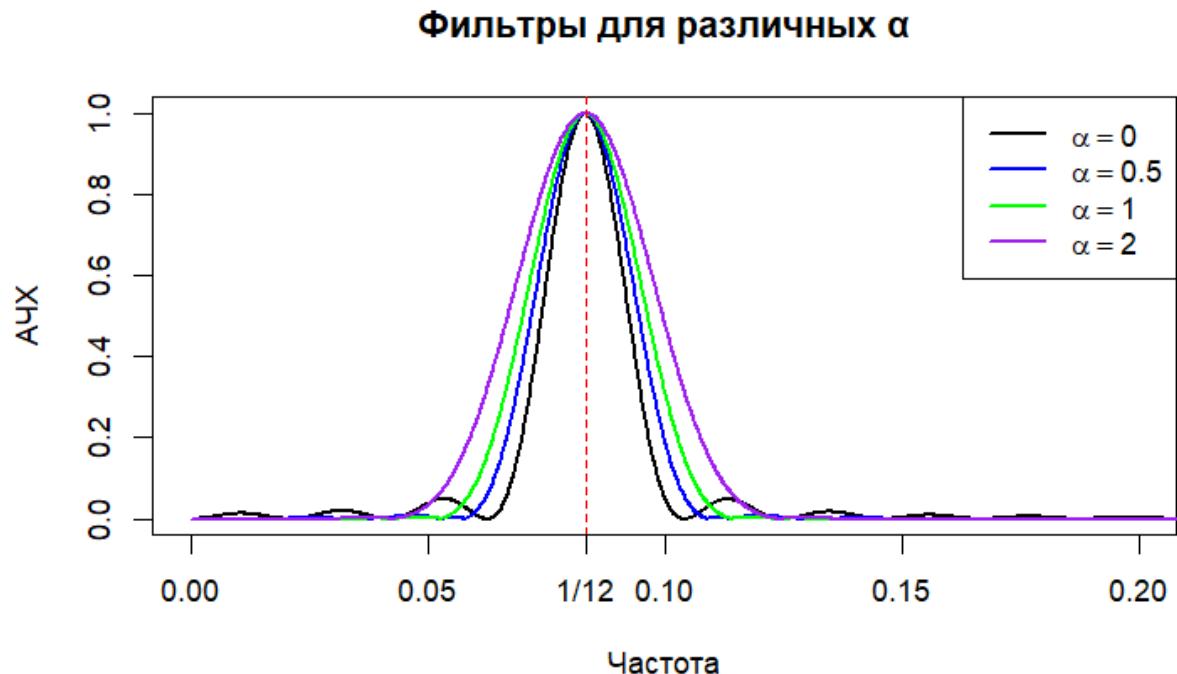


Рис. 2: АЧХ фильтров, отвечающих за  $X = \sin\left(\frac{2\pi}{12}x\right)$ , при разных  $\alpha$

На рисунке 2 показано, как фильтры ведут себя для различных значений параметра  $\alpha$ . Для всех рассмотренных значений  $\alpha$  фильтры подавляют частоты, значительно отличающиеся от частоты синуса  $\omega = \frac{1}{12}$ . При малых значениях  $\alpha$ , таких как  $\alpha = 0$ , наблюдается волнообразное поведение фильтра, что указывает на частичное захватывание соседних частот, хотя и не близких к частоте синуса. С увеличением  $\alpha$  это волнообразное поведение уменьшается, и фильтр начинает захватывать больше частот, максимально близких к  $\frac{1}{12}$ .

Таким образом, метод **GSSA** должен работать лучше **SSA** в случае, когда в временном ряде содержится пара периодических функций, частота одной из которых попадает в вершину волны АЧХ фильтра для другой функции. Например, добавим к  $X_{\sin} = \sin\left(\frac{2\pi}{12}n\right)$  косинус с частотой  $\frac{1}{19}$ . Тогда  $X = X_{\sin} + X_{\cos} = \sin\left(\frac{2\pi}{12}n\right) + \frac{1}{2} \cos\left(\frac{2\pi}{19}n\right)$ , и можем рассмотреть АЧХ,

отвечающие за синус, при базовом **SSA** ( $\alpha = 0$ ) и **GSSA** при  $\alpha = \frac{1}{2}$ . При этом,  $N = 96 \cdot 2 - 1$ ,  $L = 48$ .

### АЧХ для суммы фильтров собственных троек синуса

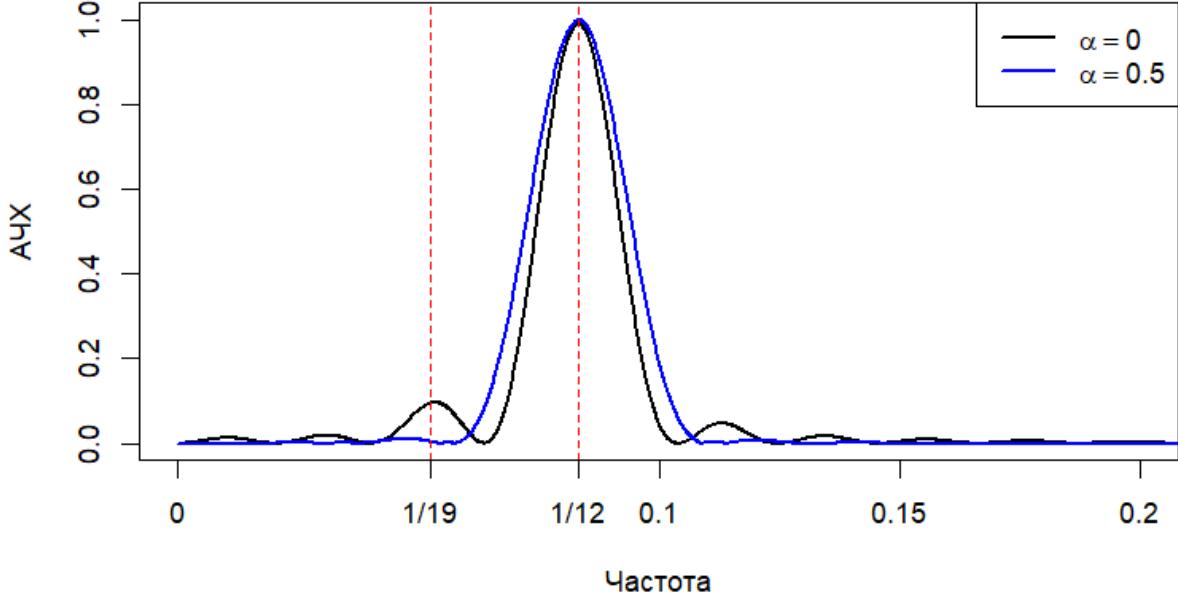


Рис. 3: Ряд  $X = X_{\sin} + X_{\cos}$ . АЧХ фильтров, отвечающих за  $X_{\sin} = \sin(\frac{2\pi}{12}n)$ , при разных  $\alpha$

По рисунку 3 заметно, что фильтр для синуса в базовом **SSA** также частично захватит периодику с частотой  $\frac{1}{19}$ , в то время, как **GSSA** не будет испытывать таких проблем. Сравним результаты по среднеквадратичной ошибке:

Таблица 3: MSE разложений ряда  $X = X_{\sin} + X_{\cos}$  для **SSA** и **GSSA** с  $\alpha = \frac{1}{2}$

Метод/Ошибка	$X_{\sin}$	$X_{\cos}$	$X$
<b>SSA</b>	5.15e-03	5.15e-03	<b>6.01e-30</b>
<b>GSSA, <math>\alpha = 0.5</math></b>	<b>3.68e-04</b>	<b>3.68e-04</b>	<b>9.53e-30</b>

Как видно из таблицы 3, **GSSA** справился с разделением на порядок лучше **SSA**.

Однако, у **GSSA** есть другая проблема. Если добавить к ряду шум, то оба алгоритма будут воспринимать этот шум как что-то близкое к частотам периодик, содержащихся в исходном ряде. А поскольку **GSSA** захватывает больше частот, максимально близких к периодикам, то и больше шума попадет в компоненты, отвечающие за периодики.

Добавим к  $X$  шумовую компоненту:  $X = X_{\sin} + X_{\cos} + X_{\text{noise}} = \sin(\frac{2\pi}{12}x) + \frac{1}{2} \cos(\frac{2\pi}{19}x) + \varepsilon_n$ , где  $\varepsilon_n \sim N(0, 0.1^2)$ ,  $N = 96 \cdot 2 - 1$ ,  $L = 48$ . Проводилось 100 тестов, в таблице 4 указаны средние значения ошибки для одних и тех же реализаций шума.

Таблица 4: MSE разложений ряда  $X = X_{\sin} + X_{\cos} + X_{\text{noise}}$  для **SSA** и **GSSA** с  $\alpha = \frac{1}{2}$

Метод	$X_{\sin}$	$X_{\cos}$	$X$
<b>SSA</b>	5.68e-03	5.44e-03	<b>7.48e-04</b>
<b>GSSA, <math>\alpha = \frac{1}{2}</math></b>	<b>1.21e-03</b>	<b>1.25e-03</b>	1.04e-03

По таблице 4 видно, что **GSSA** все же справился лучше **SSA**, однако порядок ошибки теперь одинаковый для рассмотрения косинуса или синуса. Но при этом, отделение сигнала от шума получилось лучше у **SSA**. Также был проведен парный t-критерий для зависимых выборок с целью проверки гипотезы о равенстве средних значений ошибки для каждой компоненты. В качестве нулевой гипотезы ( $H_0$ ) предполагалось, что средние значения двух сравниваемых выборок равны. Критический уровень значимости был установлен на уровне  $\alpha_{\text{hypothesis}} = 0.05$ . Результаты анализа показали, что во всех случаях  $p$ -значение оказалось меньше 0.05, что позволяет отвергнуть нулевую гипотезу.

## 4.2 Отделение сигнала от шума

Сингулярное разложение матрицы обладает наилучшими аппроксимационными свойствами в смысле минимизации нормы Фробениуса (или спектральной нормы) для заданного ранга. Из этого следует, что разложение матрицы  $\mathbf{S} = \mathbf{X}\mathbf{X}^T$  методом SVD будет наилучшим образом отделять сигнал от шума.

По результатам сравнений методов из таблиц 3 и 4, а также предыдущих рассуждений, получается, что есть смысл использовать базовый **SSA** для выделения сигнала, а уже сам сигнал разделять на различные компоненты с помощью **GSSA**.

Рассмотрим  $\mathbf{X} = \mathbf{X}_{\sin} + \mathbf{X}_{\cos} + \mathbf{X}_{\text{noise}} = \sin\left(\frac{2\pi}{12}x\right) + \frac{1}{2}\cos\left(\frac{2\pi}{19}x\right) + \varepsilon_n$ , где  $\varepsilon_n \sim N(0, 0.1^2)$ , только теперь сначала применим **SSA**, а затем **GSSA**.  $N = 96 \cdot 2 - 1$ ,  $L = 48$  для обоих методов.

Таблица 5:  $\mathbf{X}_{\sin} + \mathbf{X}_{\cos} + \varepsilon_n$ ,  $\varepsilon_n \sim N(0, 0.1^2)$ , MSE оценок

Метод/Ошибка	$\mathbf{X}_{\sin}$	$\mathbf{X}_{\cos}$	$\mathbf{X}$
<b>SSA</b>	5.68e-03	5.44e-03	<b>7.48e-04</b>
<b>GSSA</b> , $\alpha = 0.5$	<b>1.21e-03</b>	<b>1.25e-03</b>	1.04e-03
<b>SSA + GSSA</b> , $\alpha = 0.5$	<b>1.06e-03</b>	<b>1.12e-03</b>	<b>7.15e-04</b>

Анализ данных, представленных в таблице 5, позволяет сделать вывод, что комбинирование алгоритмов привело к улучшению как в выделении полезного сигнала, так и в разделении компонент между собой. Полученные результаты демонстрируют более высокую точность по сравнению с данными, приведёнными в таблице 4.

Таким образом, по приведенным примерам можно сделать вывод, что **GSSA** позволяет улучшить разделимость периодических компонент ряда. Однако, вместе с тем, разложение будет захватывать большие шума в сравнении с базовым **SSA**.

## 4.3 Фильтры в различных точках

В зависимости от точек ряда, линейные фильтры будут отличаться друг от друга. Рассмотрим тот же пример  $\mathbf{X} = \mathbf{X}_{\sin} + \mathbf{X}_{\cos} + \mathbf{X}_{\text{noise}} = \sin\left(\frac{2\pi}{12}x\right) + \frac{1}{2}\cos\left(\frac{2\pi}{19}x\right) + \varepsilon_n$ , где  $\varepsilon_n \sim N(0, 0.1^2)$ .

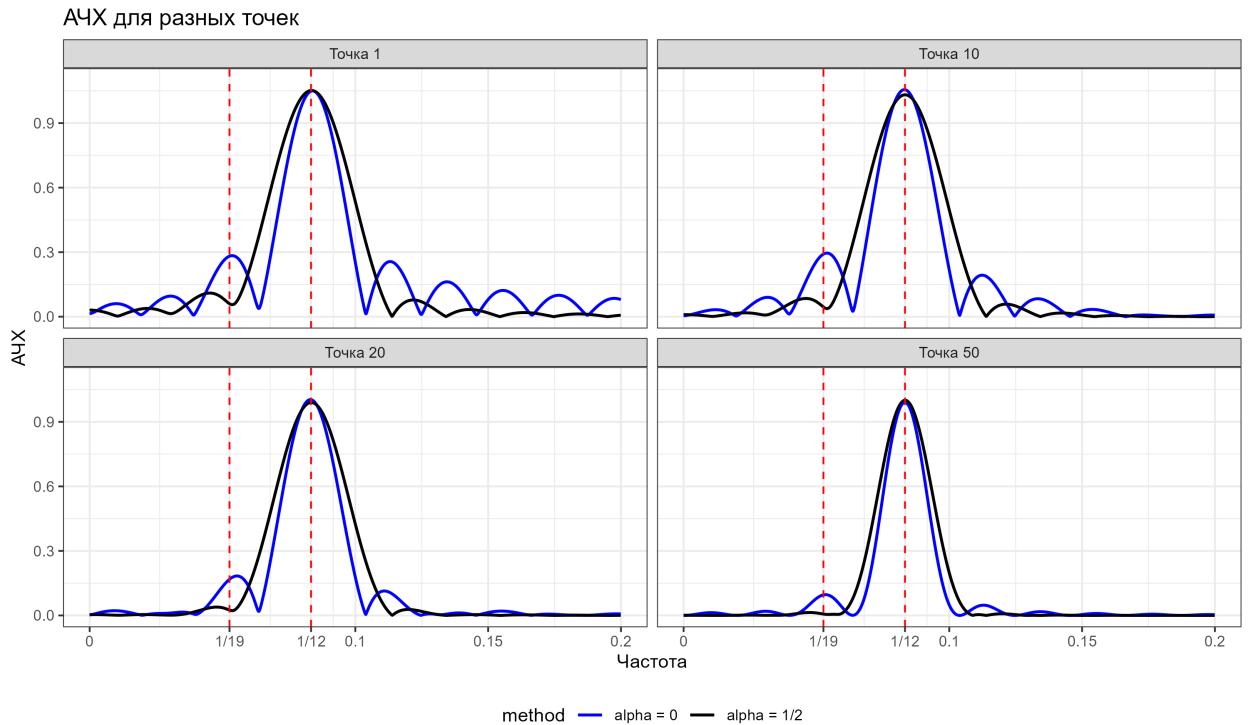


Рис. 4: Ряд  $X = X_{\sin} + X_{\cos} + X_{\text{noise}}$ . АЧХ фильтров в разных точках, отвечающих за  $X_{\sin} = \sin\left(\frac{2\pi}{12}n\right)$ , при разных  $\alpha$

По рисунку 4 видно, что когда точка  $s$  приближается по времени к средним точкам временного ряда ( $L \leq s \leq K$ ), полоса пропускания фильтра становится уже, а также фильтр начинает все меньше и меньше захватывать соседние частоты.

Для этого примера также можно посмотреть на график средней MSE ошибки в зависимости от точки ряда. Эксперимент проводился 1000 раз.

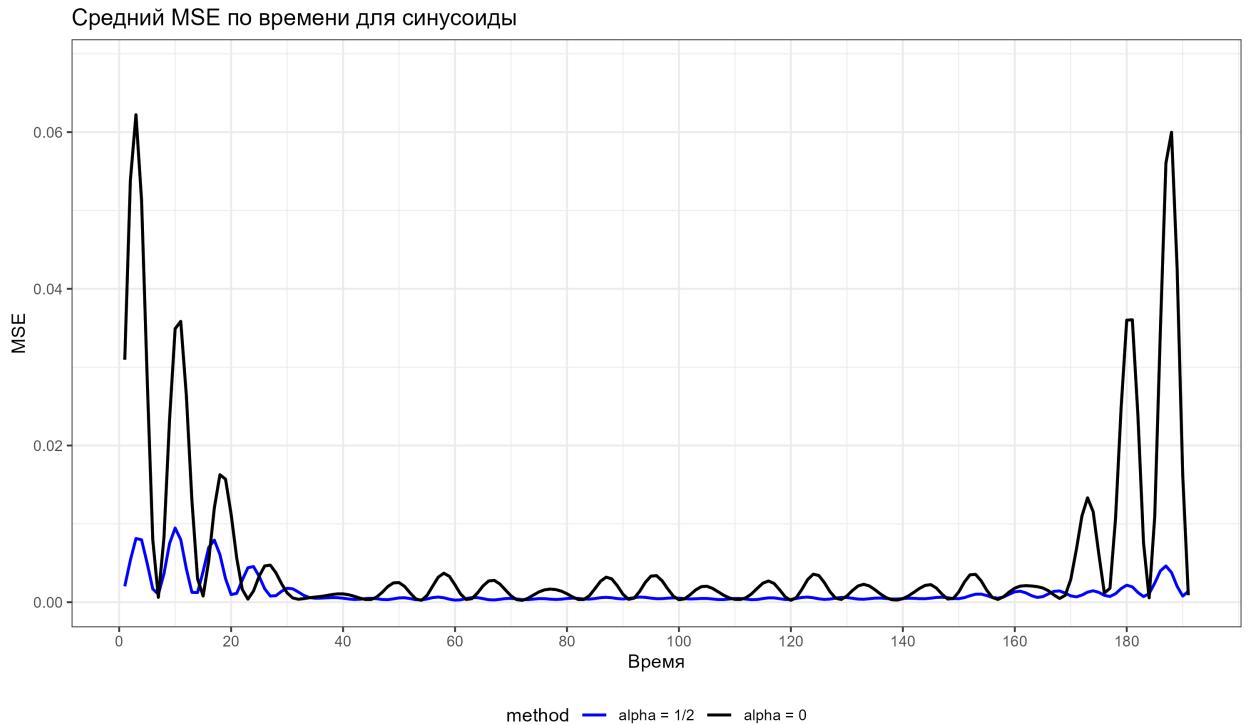


Рис. 5: Ряд  $X = X_{\sin} + X_{\cos} + X_{\text{noise}}$ . MSE ошибки выделения синуса в зависимости от точки ряда

Таким образом, можно сделать вывод, что разделимость также зависит от точки ряда. В средних точках достигаются наилучшие значения ошибки. Однако это не означает, что нужно брать маленькое  $L$ , поскольку чем больше длина окна, тем лучше происходит разделение компонент между собой в целом [4].

## 5 Метод Circulant singular spectrum analysis (CiSSA)

В этом разделе описана модификация **SSA** на основе циркулярной матрицы [1]. В отличие от базового **SSA**, в **CiSSA** для каждого конкретного  $L$  базис разложения остается одинаковым для любого входного временного ряда. Поскольку из-за этого повышается интерпретируемость каждой компоненты в разложении, авторы метода назвали **CiSSA** автоматизированной версией **SSA**. Причем автоматизированная в том смысле, что компоненты ряда группируются по частотам самим алгоритмом. Сначала будет рассмотрен метод только для стационарного случая, затем показана применимость модифицированной версии **CiSSA** при использовании нестационарного ряда.

Стационарность подразумевает неизменность статистических свойств ряда во времени. Определим это понятие формально [4].

**Определение 10.** Пусть  $\mathbf{X} = (x_1, \dots, x_n, \dots)$  — временной ряд. Ряд  $\mathbf{X}$  называется стационарным, если существует функция  $R_{\mathbf{X}}(k)$  ( $-\infty < k < +\infty$ ) такая, что для любых  $k, l \geq 1$

$$R_{\mathbf{X}}^{(N)}(k, l) \stackrel{\text{def}}{=} \frac{1}{N} \sum_{m=1}^N x_{k+m} x_{l+m} \xrightarrow{N \rightarrow \infty} R_{\mathbf{X}}(k - l). \quad (6)$$

Если (6) выполняется, тогда  $R_{\mathbf{X}}$  называется ковариационной функцией стационарного ряда  $\mathbf{X}$ .

**Теорема 3.** Пусть  $R_{\mathbf{X}}$  — ковариационная функция стационарного ряда  $\mathbf{X}$ . Тогда существует конечная мера  $m_{\mathbf{X}}$ , определенная на борелевских подмножествах  $(-1/2, 1/2]$ , такая, что

$$R_{\mathbf{X}}(k) = \int_{(-\frac{1}{2}, \frac{1}{2}]} e^{i2\pi k \omega} m_{\mathbf{X}}(d\omega).$$

Мера  $m_{\mathbf{X}}$  называется спектральной мерой ряда  $\mathbf{X}$ .

*Доказательство.* Доказательство в [4]. □

### 5.1 Алгоритм метода CiSSA

Данный алгоритм, как и **SSA**, состоит из четырех основных шагов.

Зафиксируем стационарный временной ряд  $\mathbf{X}$  состоящий из  $N$  элементов и выберем длину окна  $L$ .

#### 5.1.1 Вложение

Такой же, как и в **SSA**. Считаем матрицу  $\mathbf{X}$ , заданную в (1).

#### 5.1.2 Разложение

Для каждого  $k = 1 : L$  вычисляются собственные векторы  $U_k$ :

$$U_k = L^{-1/2}(u_{k,1}, \dots, u_{k,L}), \text{ где } u_{k,j} = \exp\left(-i2\pi(j-1)\frac{k-1}{L}\right), \text{ причем } U_k = U_{L+2-k}^*,$$

где  $U^*$  — комплексное сопряжение вектора  $U$ .

### Элементарное разложение

Для каждой частоты  $w_k = \frac{k-1}{L}$ ,  $k = 1 : \lfloor \frac{L+1}{2} \rfloor$ , есть два собственных вектора:  $U_k$  и  $U_{L+2-k}$ . За частоту  $w_0$  отвечает один собственный вектор —  $U_0$ . Если же  $L$  — четное, то частоте  $w_{\frac{L}{2}+1}$  будет соответствовать один вектор  $U_{\frac{L}{2}+1}$ .

Следовательно, индексы группируются следующим образом:

$$B_1 = \{1\}; B_k = \{k, L+2-k\}, \text{ для } k = 1 : \lfloor \frac{L+1}{2} \rfloor; B_{\frac{L}{2}+1} = \left\{ \frac{L}{2} + 1 \right\}, \text{ если } L \bmod 2 = 0.$$

Таким образом, получается элементарная группировка по частотам  $w_k$ :

$$\begin{aligned} \mathbf{X}_{B_k} &= \mathbf{X}_k + \mathbf{X}_{L+2-k} = U_k U_k^H \mathbf{X} + U_{L+2-k} U_{L+2-k}^H \mathbf{X}, \text{ для } k = 1 : \lfloor \frac{L+1}{2} \rfloor; \\ \mathbf{X}_{B_{\frac{L}{2}+1}} &= \mathbf{X}_{\frac{L}{2}+1} = U_{\frac{L}{2}+1} U_{\frac{L}{2}+1}^H \mathbf{X}, \text{ если } L \bmod 2 = 0, \end{aligned}$$

где  $U^H$  — это комплексное сопряжение и транспонирование вектора  $U$ .

Пусть  $d = \lfloor \frac{L+1}{2} \rfloor$ , если  $L \bmod 2 \neq 0$ , иначе  $d = \frac{L}{2} + 1$ . Тогда результатом данного шага будет разложение исходной матрицы  $\mathbf{X}$  в сумму матриц  $\mathbf{X}_{B_k}$ , отвечающих периодикам с определенными частотами  $w_k$ :

$$\mathbf{X} = \sum_{k=1}^d \mathbf{X}_{B_k}.$$

#### 5.1.3 Группировка

Такой же шаг, как и в базовом SSA. Однако группировка будет производиться на непересекающиеся подгруппы по частотам от  $w_k$ , которые находятся в диапазоне от 0 до 0.5. То есть, заранее заданному произвольному количеству непересекающихся диапазонов  $I_i = [w_{i0}, w_{i1}]$ ,  $w_{i0} \leq w_{i1}$  и  $0 \leq w_{i0}, w_{i1} \leq 0.5$ , строятся матрицы  $\mathbf{X}_{I_i}$ , в которые входят суммы  $\mathbf{X}_{B_k}$ , отвечающие частотам  $w_k : w_{i0} \leq w_k \leq w_{i1}$ .

#### 5.1.4 Диагональное усреднение

Такой же шаг, как и в базовом SSA.

**Замечание 2.**  $U_k$  можно получить по аналогии с SSA.

Будем рассматривать временной ряд как выборку после эксперимента, а не как случайную величину. Соответственно, все формулы будут выборочными.

Определим автоковариации:

$$\hat{\gamma}_m = \frac{1}{N-m} \sum_{t=1}^{N-m} x_t x_{t+m}, m = 0 : (L-1).$$

На основе  $\hat{\gamma}_m$  определим матрицу:

$$\hat{\gamma}_L = \begin{pmatrix} \hat{\gamma}_1 & \hat{\gamma}_2 & \dots & \hat{\gamma}_L \\ \hat{\gamma}_2 & \hat{\gamma}_1 & \dots & \hat{\gamma}_{L-1} \\ \vdots & \vdots & \vdots & \vdots \\ \hat{\gamma}_L & \hat{\gamma}_{L-1} & \dots & \hat{\gamma}_1 \end{pmatrix}. \quad (7)$$

Данная матрица  $L \times L$  называется Тэплицевой и используется в методе Toeplitz SSA (подробнее про данный метод можно прочитать в книге [4]). На ее основе составим циркулярную матрицу для алгоритма Circulant SSA [1]:

$$\hat{C}_L = \begin{pmatrix} \hat{c}_1 & \hat{c}_2 & \dots & \hat{c}_L \\ \hat{c}_2 & \hat{c}_1 & \dots & \hat{c}_{L-1} \\ \vdots & \vdots & \vdots & \vdots \\ \hat{c}_L & \hat{c}_{L-1} & \dots & \hat{c}_1 \end{pmatrix}, \quad (8)$$

где  $\hat{c}_m = \frac{L-m}{L}\hat{\gamma}_m + \frac{m}{L}\hat{\gamma}_{L-m}$ ,  $m = 0 : L - 1$ . Собственные числа матрицы  $\hat{C}_L$ , определенной в (8) задаются по формуле:

$$\lambda_{L,k} = \sum_{m=0}^{L-1} \hat{c}_m \exp\left(i2\pi m \frac{k-1}{L}\right), \quad k = 1 : L, \text{ причем } \lambda_{L,k} = \lambda_{L,L+2-k},$$

а собственные вектора, связанные с  $\lambda_{L,k}$  — это векторы  $U_k$ .

**Замечание 3.**  $U_k U_k^H + U_{L+2-k} U_{L+2-k}^H$  является оператором проектирования на подпространство, которое порождено синусами и косинусами с частотой  $w_k = \frac{k-1}{L}$ . Это пространство соответствует компонентам синусоидальной структуры временного ряда, связанных с конкретной частотой, выделяемой методом.

*Доказательство.* Рассмотрим на примере одного вектора-столбца  $X_i = (x_i, \dots, x_{i+L})^T$ , где  $i = 1, \dots, K$ . Возьмем для наглядности  $i = 1$ .

$$U_k = L^{-\frac{1}{2}} \left( 1, e^{-i2\pi \frac{k-1}{L}}, e^{-i2\pi 2 \frac{k-1}{L}}, \dots, e^{-i2\pi (L-1) \frac{k-1}{L}} \right)^T,$$

$$U_k^H = L^{\frac{1}{2}} \left( 1, e^{i2\pi \frac{k-1}{L}}, e^{i2\pi 2 \frac{k-1}{L}}, \dots, e^{i2\pi (L-1) \frac{k-1}{L}} \right).$$

$$L^{-\frac{1}{2}} c_k = U_k^H X_1 = x_1 + e^{i2\pi \frac{k-1}{L}} x_2 + e^{i2\pi 2 \frac{k-1}{L}} x_3 + \dots + e^{i2\pi (L-1) \frac{k-1}{L}} x_L.$$

$$X_1^k = c_k U_k = \left( c_k, c_k e^{-i2\pi \frac{k-1}{L}}, c_k e^{-i2\pi 2 \frac{k-1}{L}}, \dots, c_k e^{-i2\pi (L-1) \frac{k-1}{L}} \right)^T.$$

Таким образом, получилось проектирование на пространство синусов и косинусов, если разложить комплексную экспоненту. Если брать всю матрицу  $\mathbf{X}$ , выйдет  $K$  столбцов, спроектированных на данное пространство.  $\square$

**Замечание 4.** В разделе 5.2.1 рассмотрена связь между матрицей  $\mathbf{X}_{B_k}$  и разложениями Фурье для векторов вложсения.

## Нестационарный случай

Для применения данного алгоритма на нестационарных временных рядах, нужно применить процедуру расширения ряда. Как утверждается авторами статьи [1], после расширения, CiSSA можно применить к нестационарному ряду. Сама процедура расширения ряда  $\mathbf{X}$  производится с использованием авторегрессионной (AR) модели. Эта процедура позволяет предсказывать значения временного ряда за его пределами (экстраполяция) как вправом, так и влевом направлениях на заданное число шагов  $H$ . Таким образом, трендовая (нелинейная) компонента ряда будет выделяться заметно лучше. В ходе работы алгоритм выполняет следующие шаги:

- Определение порядка AR-модели:** Метод определяет порядок  $p$  AR-модели как целую часть от деления длины ряда  $N$  на 3. Это значение порядка модели  $p$  будет использовано для построения авторегрессионной модели на дифференцированном временном ряде;
- Построение дифференцированного ряда:** Временной ряд  $X$  сначала преобразуется в дифференцированный ряд  $dX$ , чтобы удалить трендовые компоненты;
- Построение AR-модели:** После этого для дифференцированного ряда вычисляются коэффициенты авторегрессионной модели  $A$  с использованием метода Юла-Уокера, основываясь на определенном ранее порядке  $p$ ;
- Правое расширение ряда:** С помощью AR-модели ряд  $dX$  прогнозируется на  $H$  шагов вправо. Затем возвращается к своему изначальному состоянию путем интегрирования  $dX$ . Получается расширение исходного ряда  $X$  на  $H$  шагов вправо;
- Левое расширение ряда:** Аналогично предыдущему пункту, ряд прогнозируется на  $H$  шагов влево;
- Возвращение расширенного ряда:** В конце метод возвращает расширенный временной ряд  $X_{\text{extended}}$ , который содержит как левое, так и правое расширение на  $H$  шагов от исходного ряда  $X$ .

Таким образом, алгоритм расширения ряда позволяет выполнять предсказания временного ряда по обе стороны от его границ, основываясь на авторегрессионной модели, построенной на дифференцированном ряде, что полезно для выделения тренда.

### Расширение временного ряда IP values

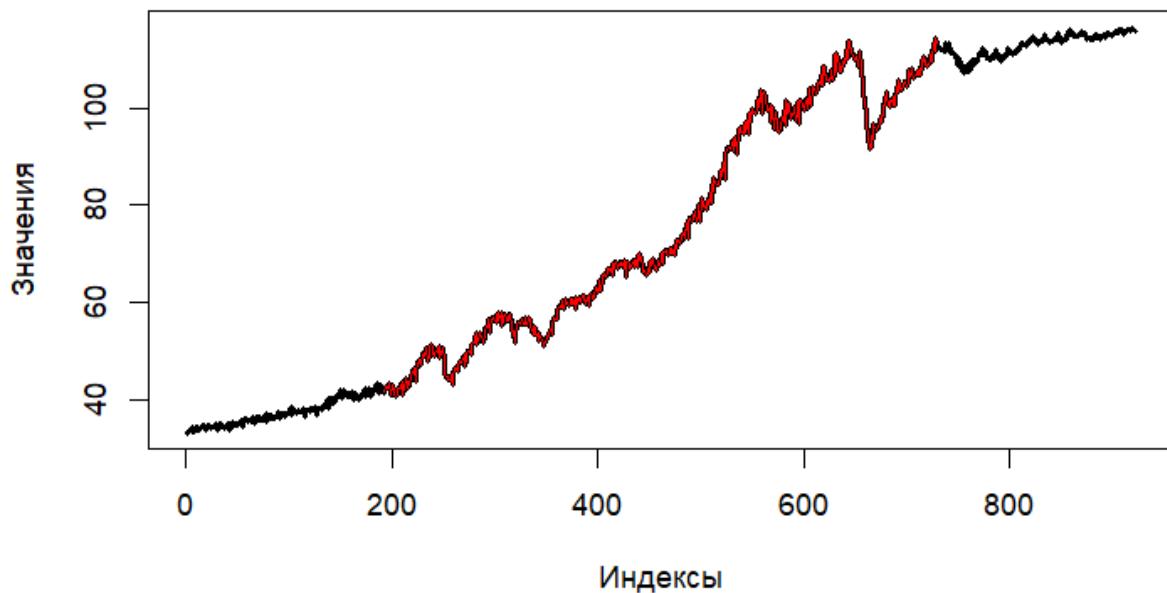


Рис. 6: Расширение временного ряда IP values. Красным показан настоящий ряд, черным — его расширение

Однако поскольку мы рассматриваем расширенный ряд, то и периодические компоненты будут строиться по нему. Поэтому в угоду лучшего выделения трендовой составляющей, будет несколько жертвоваться точность разделения периодических компонентов.

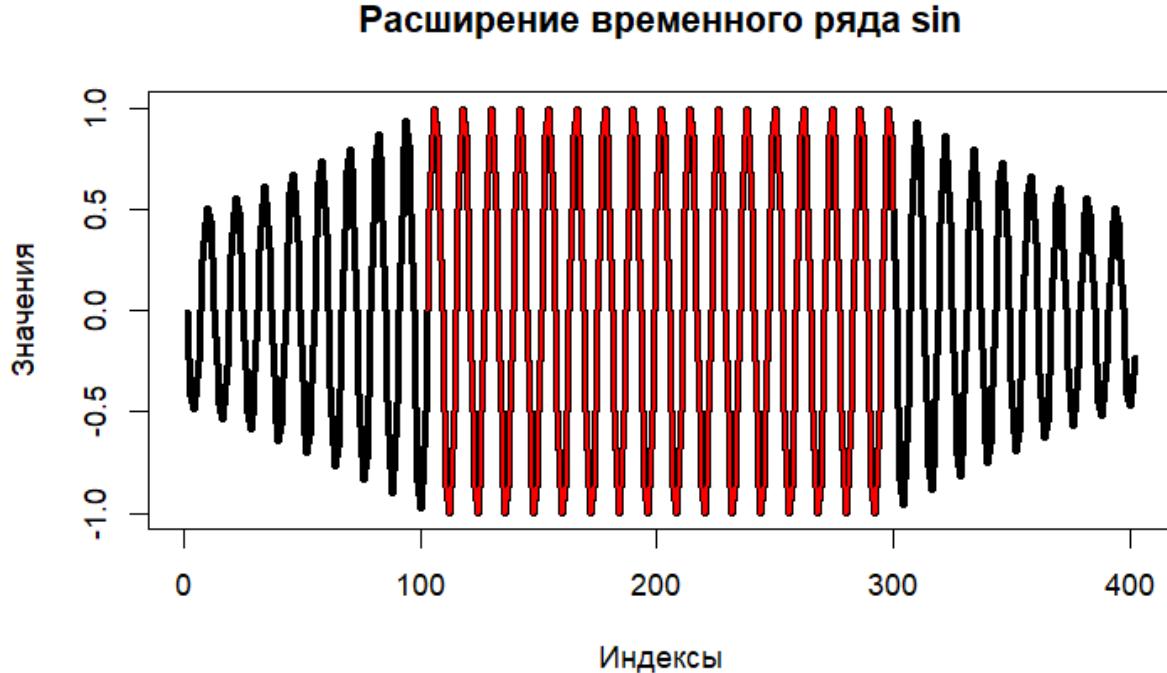


Рис. 7: Расширение временного ряда синуса. Красным показан настоящий ряд, черным — его расширение

На рисунке 7 видно, что синус расширился неправильно, от концов настоящего ряда до концов расширенного значения постепенно уменьшались. Как будет показано в секции ??, это повлияет на значения ошибки.

## 5.2 Свойства

### 5.2.1 Связь CiSSA с разложением Фурье

Для описания конечных, но достаточно длинных рядов можно использовать разложение Фурье. Пусть  $\mathbf{X} = (x_1, \dots, x_N)$  — временной ряд

**Определение 11.** *Разложение*

$$x_n = c_0 + \sum_{k=1}^{\lfloor \frac{N+1}{2} \rfloor} (c_k \cos(2\pi n k / N) + s_k \sin(2\pi n k / N)), \quad (9)$$

где  $1 \leq n \leq N$  и  $s_{N/2} = 0$  для четного  $N$ , называется разложением Фурье ряда  $\mathbf{X}$ .

Таким образом, можно выделить компоненту ряда, отвечающую за частоту  $w_k = \frac{k-1}{L}$ ,  $k = 1 : \lfloor \frac{N+1}{2} \rfloor$ ;

Алгоритм **CiSSA** тесно связан с разложением Фурье. По замечанию 3 видно, что при вычислении  $\mathbf{X}_{B_k} = \mathbf{X}_k + \mathbf{X}_{L+2-k} = U_k U_k^H \mathbf{X} + U_{L+2-k} U_{L+2-k}^H \mathbf{X}$ , воспроизводится разложение Фурье для  $K$  векторов матрицы  $\mathbf{X}$ . Затем вычисляется диагональное усреднение  $\mathbf{X}_{B_k}$ . А именно, **CiSSA** можно представить так:

1. Вычисляем разложение Фурье для каждого вектора вложения  $L$ -траекторной матрицы  $\mathbf{X}$ , состоящей из  $K = N - L + 1$  векторов. Получается  $K$  разложений Фурье по частотам  $w_k = \frac{k-1}{L}, k = 1 : \lfloor \frac{L+1}{2} \rfloor$ ;
2. По получившимся разложениям Фурье усредняем значения для соответствующих  $x_i$  и частот  $w_k$ .

### 5.2.2 Точная разделимость

Поскольку данный метод является аналогом разложения Фурье, то в смысле сильной разделимости можно точно разделить ряд, в котором одной из компонентов является  $\cos(2\pi\omega + \varphi)$  с частотой  $\omega$  такой, что  $L\omega = k \in \mathbb{N}$ , или константа. Для сравнения, при применении базового **SSA**, условие накладывалось не только на  $L\omega \in \mathbb{N}$ , но и на  $K\omega \in \mathbb{N}$ .

Поэтому до применения алгоритма необходимо выделить интересующие частоты, то есть знать их заранее, и, исходя из них, выбирать значение  $L$ .

### 5.2.3 Асимптотическая разделимость

Асимптотическая разделимость в данном случае будет означать, что при увеличении  $L$  разбиение сетки будет увеличиваться, а значит, и частоты в сетке начнут сближаться к истинным частотам периодических компонентов (либо становиться равными им), что будет снижать ошибку вычислений.

То есть, в случае непопадания периода определенной компоненты в разбиение частот алгоритма, будет выполняться **CiSSA**-асимптотическая  $L(N)$ -разделимость по определению 1.

## 5.3 Обзор литературы

В данной секции рассмотрены применения **CiSSA** на практике.

### 5.3.1 Cognitive Load Detection through EEG Lead-Wise Feature Optimization and Ensemble Classification

В статье [10] рассматриваются несколько наборов данных ЭЭГ под различной нагрузкой, два из которых наиболее значимы:

- **MAT (Mental Arithmetic Task)**: участие студентов в задаче на математический счёт.
- **STEW (Simultaneous Task EEG Workload)**: параллельное выполнение нескольких заданий.

Задача исследования заключалась в следующем:

- С помощью метода **CiSSA** (и других различных методов) разлагают исходные сигналы ЭЭГ на несколько компонент, каждая из которых несёт информацию о разных частотных диапазонах и временных структурах.

- После разложения из полученных компонент (или их комбинаций) вычисляют числовые признаки: энергетические, энтропийные и другие.
- Эти признаки затем подают на вход классификатору (например,  $k$ NN, SVM или другому алгоритму машинного обучения), чтобы автоматически определить уровень когнитивной нагрузки. Классификация может быть как бинарной (наличие/отсутствие нагрузки), так и многоуровневой (лёгкая/средняя/высокая нагрузка).

Таким образом, метод CiSSA был выбран в качестве одного из подходов разложения сигнала по частотам.

Выводом исследования является таблица 5 статьи [10], в которой авторы сравнивают по метрикам различные решения задачи. Подходы, связанные с **CiSSA**, являются не самыми наилучшими.

### **5.3.2 Application of visual stratigraphy from line-scan images to constrain chronology and melt features of a firn core from coastal Antarctica**

В работе [2] исследовалось таяние ледников, а также анализировались данные визуальной стратиграфии (VS) для построения хронологии фирнового керна из прибрежной Антарктиды. Основной задачей было отделение долгосрочного тренда, связанного с изменением плотности фирна, от сезонных сигналов, обусловленных включениями пыли и морской соли. Для разложения сигнала по частотам был выбран алгоритм **CiSSA**. Длина окна была установлена равной 10, поскольку "дальнейшее её увеличение не оказывало существенного влияния на результаты".

#### **Ключевые этапы анализа**

- **Первая компонента (RC1):** отвечала за долгосрочный тренд, связанный с постепенным увеличением плотности фирна с глубиной;
- **Компоненты со второй по пятую (RC2-RC5):** отражали сезонность, обусловленную изменениями в содержании пыли и морских солей;
- **Остальные компоненты:** содержали шумовой сигнал и не учитывались в дальнейшем анализе.

Таким образом, метод CiSSA был использован как инструмент частотного разложения.

### **5.3.3 Выводы**

В рассмотренных работах **CiSSA** используется как прикладной инструмент для разделения сигнала по частотам. В работе [10] это конкретные частоты, а для [2] это инструмент для выделения тренда. По целям использования алгоритмов, можно сделать вывод, что с таким же успехом можно было применить базовый **SSA** с автоматической группировкой по компонентам как сказано в секции 2.2.4.

## 6 Различия SSA, CiSSA и Фурье

В данной секции проводится сравнение методов разложения временного ряда: базовый **SSA**, **SSA** с использованием EOSSA для улучшения разделимости, разложения Фурье и Фурье с расширением ряда, базового **CiSSA** и **CiSSA** с расширением ряда. Все вычисления, а также код метода **CiSSA** можно найти в github репозитории [8]

### 6.1 Автоматическая группировка

Авторы статьи [1] выделяют главным преимуществом то, что **CiSSA** автоматически разделяет компоненты ряда по частотам. Однако есть метод, позволяющий сделать автоматическое объединение частот по периодограмме в методе **SSA** [6]. При этом, прежде чем применять его, стоит выполнить процедуру улучшения разделимости. В данной работе будут использоваться методы EOSSA и FOSSA [3]. Отсюда следует, что все рассматриваемые в данной секции алгоритмы могут по заранее заданным диапазонам частотам выдать временные ряды, отвечающие за эти диапазоны.

### 6.2 Собственные пространства

Каждый алгоритм после группировки порождает построенными матрицами собственные подпространства. В случае базового **SSA** базис подпространств является аддитивным, то есть зависящим от  $X, L, N$ .

В случае **CiSSA** базис зависит только от  $L, N$ . Если зафиксировать данные параметры, и менять  $X$ , базис никак не поменяется.

На базис разложения Фурье влияет только  $N$ .

### 6.3 Точная разделимость

В свойствах методов были приведены классы функций и условия, при которых методы могут безошибочно разделить два ряда друг от друга. Сравним эти условия.

Разложение Фурье может отличить друг от друга периодические компоненты, попадающие в решетку его частот. Другими словами, разложением Фурье может быть точно разделен ряд  $X = X_{w_1} + X_{w_2}$ , где  $X_{w_1} = A_1 \cos(2\pi w_1 n + \varphi_1)$ ,  $X_{w_2} = A_2 \cos(2\pi w_2 n + \varphi_2)$  и  $Nw_1, Nw_2 \in \mathbb{N}$ ,  $w_1 \neq w_2$ .

Похожие условия точной разделимости у метода **CiSSA**. С помощью данного алгоритма может быть точно разделен ряд  $X = X_{w_1} + X_{w_2}$ , где  $X_{w_1} = A_1 \cos(2\pi w_1 n + \varphi_1)$ ,  $X_{w_2} = A_2 \cos(2\pi w_2 n + \varphi_2)$ , только  $Lw_1, Lw_2 \in \mathbb{N}$ ,  $w_1 \neq w_2$ .

У алгоритма **SSA** для разделения  $X = X_{w_1} + X_{w_2}$  накладываются более жесткие ограничения:  $Lw_1, Lw_2, Kw_1, Kw_2 \in \mathbb{N}$ ,  $w_1 \neq w_2$ ,  $A_1 \neq A_2$ . Однако также могут быть точно разделены ряды  $X = X_{\exp_1} + X_{\exp_2} = A_1 \exp(\alpha_1 n) \cos(2\pi w_1 n + \varphi_1) + A_2 \exp(\alpha_2 n) \cos(2\pi w_2 n + \varphi_2)$ , где  $Lw_1, Lw_2, Kw_1, Kw_2 \in \mathbb{N}$ ,  $w_1 \neq w_2$ ,  $A_1 \neq A_2$ ,  $\alpha_1 \neq \alpha_2$ .

**Пример.** Будем разделять временной ряд  $X = X_{\sin} + X_{\cos} = \sin \frac{2\pi}{12} n + \frac{1}{2} \cos \frac{2\pi}{3} n$ . Рассмотрим разложения методов в лучшем и худшем случае. Для всех алгоритмов кроме базового **SSA** выделялись периодические компоненты по диапазонам  $(w \pm \Delta)$ , где  $\Delta = \frac{1}{N+1}$ ,  $w = \frac{1}{12}, \frac{1}{3}$ .

Таблица 6: MSE ошибки разложений методов ряда  $\mathbf{X} = \mathbf{X}_{\sin} + \mathbf{X}_{\cos}$  в лучших и худших случаях

Метод	Параметры	MSE( $\mathbf{X}_{\sin}$ )	MSE( $\mathbf{X}_{\cos}$ )	MSE( $\mathbf{X}$ )	
SSA SSA EOSSA, $r = 4$	$L = 96, K = 96$ ( $Lw, Kw \in \mathbb{N}$ )	6.8e-30	1.5e-29	1.8e-29	
	$L = 96, K = 96$ ( $Lw, Kw \in \mathbb{N}$ )	1.5e-29	7.5e-30	2.0e-29	
	$N = 96 \cdot 2$ ( $Nw \in \mathbb{N}$ )	1.7e-28	3.5e-28	5.1e-28	
	$N = 96 \cdot 2$ ( $Nw \in \mathbb{N}$ )	6.2e-04	2.6e-03	3.2e-03	
	CiSSA	$L = 96$ ( $Lw \in \mathbb{N}$ )	1.9e-29	5.3e-30	2.1e-29
	CiSSA extended	$L = 96$ ( $Lw \in \mathbb{N}$ )	2.0e-04	8.6e-04	1.1e-03
SSA SSA EOSSA, $r = 4$	$L = 96, K = 97$ ( $Lw \in \mathbb{N}, Kw \notin \mathbb{N}$ )	2.2e-06	2.2e-06	2.0e-29	
	$L = 96, K = 97$ ( $Lw \in \mathbb{N}, Kw \notin \mathbb{N}$ )	1.5e-29	8.8e-30	1.9e-29	
	Fourier	$N = 96 \cdot 2 - 1$ ( $Nw \notin \mathbb{N}$ )	9.4e-03	3.5e-03	1.3e-02
	Fourier extended	$N = 96 \cdot 2 - 1$ ( $Nw \notin \mathbb{N}$ )	1.1e-05	4.9e-04	4.9e-04
	CiSSA	$L = 97$ ( $Lw \notin \mathbb{N}$ )	1.7e-02	7.0e-03	2.3e-02
	CiSSA extended	$L = 97$ ( $Lw \notin \mathbb{N}$ )	2.4e-03	6.9e-04	3.1e-03

Метод	sin_err	cos_err	all_err
SSA, Lw, Kw in N	6.8e-30	1.5e-29	1.8e-29
SSA EOSSA, Lw, Kw in N	1.5e-29	7.5e-30	2.0e-29
Fourier, Nw in N	9.8e-29	3.4e-28	4.0e-28
CiSSA, Lw in N	6.5e-30	1.1e-29	7.8e-30
CiSSA extended, Lw in N	5.5e-06	1.6e-06	3.7e-06
Fourier extended, Nw in N	1.4e-06	8.4e-07	5.9e-07
SSA, Lw in N, Kw not in N	2.2e-06	2.2e-06	2.0e-29
SSA EOSSA, Lw in N, Kw not in N	1.3e-29	8.8e-30	1.7e-29
Fourier, Nw not in N	7.9e-04	4.9e-04	2.4e-04
CiSSA, Lw not in N	1.7e-03	1.4e-03	2.5e-04
CiSSA extended, Lw not in N	1.0e-05	5.8e-06	3.1e-06
Fourier extended, Nw not in N	1.2e-05	2.3e-06	1.4e-05

Таблица 7: Example Table

Таблица 6 подтверждает теоретические результаты. Кроме того, можно заметить, что разложение Фурье справляется лучше при невыполнении условий точной разделимости, чем **CiSSA**. Это объяснимо тем, что для разложения Фурье частоты делятся на  $N$  частей, а для **CiSSA** на  $L$ . В данном примере, разбиение сетки частот у разложения Фурье в два раза меньше, чем у **CiSSA** ( $N = 96 \cdot 2, L = 96$ ). В условиях точной разделимости результаты примерно одинаковы.

Таким образом, условия на разделение синусов, слабее у методов **CiSSA** и Фурье, чем у **SSA**. Однако **SSA** может точно отличать друг от друга больше классов функций.

#### 6.4 Асимптотическая разделимость

Как было сказано, асимптотически разделимы в методе **SSA** полиномы, гармонические функции (косинус, косинус помноженный на экспоненту) [4].

В алгоритме **CiSSA** при увеличении длины окна  $L$  меняется сетка разбиения частот. Из-за этого, даже если не удастся выбрать подходящее  $L$ , при котором будет точно отделить косинус, но постоянно его увеличивать, в конечном счете получится снизить ошибку выделения

нужной компоненты косинуса, если брать соседние частоты с частотой компоненты. Однако в таком подходе есть две проблемы. Во-первых, в этом случае нужно выбирать диапазон частот, которые стоит объединить. Во-вторых, в реальности это труднореализуемо, слишком большое  $N$  и  $L$  придется выбрать, чтобы значимо снизить ошибку. Поэтому, при использовании **CiSSA** обязательно нужно заранее понимать, какие частоты интересуют. Аналогичная ситуация для разложения Фурье.

Теперь рассмотрим разложение непериодических компонент. Поскольку все непериодические компоненты относятся к частотам достаточно близким к нулю, то и разделить между собой непериодические компоненты методы **CiSSA** и Фурье не могут даже асимптотически, в отличие от **SSA**.

## 6.5 Выделение тренда

Рассмотрим, влияние непериодических компонент на разложение ряда.

Базовый алгоритм **SSA** может выделять трендовую составляющую за счет своего адаптивного базиса. Для алгоритмов **CiSSA** и разложения Фурье нужно применять процедуры расширения временного ряда, чтобы использовать их для выделения тренда.

**Пример.** Рассмотрим ряд из примера в секции 6.3 и добавим к нему тренд.  $X = X_c + X_e + X_{\sin} + X_{\cos} = 1 + e^{\frac{n}{100}} + \sin \frac{2\pi}{12}n + \frac{1}{2} \cos \frac{2\pi}{3}n$ . Кроме того, для всех алгоритмов кроме базового **SSA** выделялись периодические компоненты по диапазонам  $(w \pm \Delta)$ , где  $\Delta = \frac{1}{N+1}$ ,  $w = \frac{1}{12}, \frac{1}{3}$ , а непериодичность соответствовала диапазону частот  $[0, \frac{1}{24})$ . Будем искать экспоненту и константу по низким частотам, назовем это трендовой составляющей ряда.

Таблица 8: MSE разложений ряда  $X = X_c + X_e + X_{\sin} + X_{\cos}$

Метод	Параметры	MSE( $X_c + X_e$ )	MSE( $X_{\sin}$ )	MSE( $X_{\cos}$ )	MSE( $X$ )
SSA	$L = 96, K = 96$	6.1e-05	8.9e-07	5.2e-05	2.1e-28
SSA EOSSA, $r = 6$	$L = 96, K = 96$	1.7e-28	1.6e-29	8.7e-30	1.6e-28
Fourier	$N = 96 \cdot 2$	1.1e-01	6.1e-04	6.8e-03	1.1e-01
Fourier extended	$N = 96 \cdot 2$	1.4e-03	1.3e-03	8.4e-03	9.6e-03
CiSSA	$L = 96$	5.3e-02	1.6e-05	4.9e-04	4.4e-02
CiSSA extended	$L = 96$	5.0e-04	2.1e-04	1.1e-03	6.0e-04
SSA	$L = 96, K = 97$	7.3e-05	4.2e-06	6.2e-05	1.1e-27
SSA EOSSA, $r = 6$	$L = 96, K = 97$	1.0e-27	2.3e-29	9.7e-30	9.5e-28
Fourier	$N = 96 \cdot 2 - 1$	1.2e-01	1.9e-02	2.2e-02	1.0e-01
Fourier extended	$N = 96 \cdot 2 - 1$	2.7e-03	3.1e-04	3.1e-03	5.9e-03
CiSSA	$L = 97$	7.6e-02	4.1e-02	1.4e-02	1.1e-01
CiSSA extended	$L = 97$	5.8e-04	1.3e-02	2.0e-03	1.4e-02

По таблице 8 видно, что алгоритмы **CiSSA** и Фурье без модификаций достаточно плохо определяют тренд. Ситуация с выделением тренда улучшается при использовании процедуры расширения ряда, однако есть недостаток у такого решения. В примерах, рассматривавшихся для таблиц 6 и 8, можно заметить, что при применении расширения ряда ухудшается отделение периодических составляющих. Подобной проблемы нет с улучшением разделимости в методе **SSA**.

## 6.6 Отделение сигнала от шума

Рассмотрим влияние шума на результаты разделимости предыдущих примеров.

**Пример.** Вернемся к примеру из секции 6.3 и добавим к нему шум:  $X = X_{\sin} + X_{\cos} + X_{\text{noise}} = \sin \frac{2\pi}{12}n + \frac{1}{2} \cos \frac{2\pi}{3}n + \varepsilon_n$ , где  $\varepsilon_n \sim N(0, 0.1^2)$ . Кроме того, для всех алгоритмов кроме базового **SSA** выделялись периодические компоненты по диапазонам  $(w \pm \Delta)$ , где  $\Delta = \frac{1}{N+1}$ ,  $w = \frac{1}{12}, \frac{1}{3}$ . Проводилось 100 тестов, в таблице 9 указаны средние значения ошибки для одних и тех же реализаций шума.

Таблица 9: MSE разложений ряда  $X = X_{\sin} + X_{\cos} + X_{\text{noise}}$

Метод	Параметры	MSE( $X_{\sin}$ )	MSE( $X_{\cos}$ )	MSE( $X$ )
SSA	$L = 96, K = 96$	2.9e-04	3.1e-04	5.9e-04
SSA EOSSA, $r = 4$	$L = 96, K = 96$	2.9e-04	3.1e-04	5.9e-04
Fourier	$N = 96 \cdot 2$	1.0e-04	1.1e-04	2.2e-04
Fourier extended	$N = 96 \cdot 2$	1.2e-03	3.9e-03	5.1e-03
CiSSA	$L = 96$	1.6e-04	1.8e-04	3.4e-04
CiSSA extended	$L = 96$	6.6e-04	1.9e-03	2.5e-03
SSA	$L = 96, K = 97$	2.9e-04	3.1e-04	5.9e-04
SSA EOSSA, $r = 4$	$L = 96, K = 97$	2.9e-04	3.0e-04	5.9e-04
Fourier	$N = 96 \cdot 2 - 1$	1.8e-02	7.6e-03	2.6e-02
Fourier extended	$N = 96 \cdot 2 - 1$	1.2e-03	8.4e-04	2.0e-03
CiSSA	$L = 97$	4.1e-02	1.2e-02	5.2e-02
CiSSA extended	$L = 97$	1.4e-02	3.0e-03	1.7e-02

По таблице 9 видно, что зашумление ряда сильно повлияло на ошибку, теперь она не является машинным нулем ни для одного из методов. Также был проведен парный t-критерий для зависимых выборок с целью проверки гипотезы о равенстве средних значений ошибки для каждой компоненты, попарно для всех методов. В качестве нулевой гипотезы ( $H_0$ ) предполагалось, что средние значения двух сравниваемых выборок равны. Критический уровень значимости был установлен на уровне  $\alpha = 0.05$ . Результаты анализа показали, что во всех случаях, кроме сравнения **SSA** и **SSA** с EOSSA,  $p$ -значение оказалось меньше 0.05, что позволяет отвергнуть нулевую гипотезу.

**Пример.** Теперь вновь добавим трендовую составляющую к ряду:  $X = X_c + X_e + X_{\sin} + X_{\cos} + X_{\text{noise}} = 1 + e^{\frac{x}{100}} + \sin \frac{2\pi}{12}x + \frac{1}{2} \cos \frac{2\pi}{3}x + \varepsilon_n$ , где  $\varepsilon_n \sim N(0, 0.1^2)$ . Кроме того, для всех алгоритмов кроме базового **SSA** выделялись периодические компоненты по диапазонам  $(w \pm \Delta)$ , где  $\Delta = \frac{1}{N+1}$ ,  $w = \frac{1}{12}, \frac{1}{3}$ , а непериодичность соответствовала диапазону частот  $[0, \frac{1}{24}]$ . Проводилось 100 тестов, в таблице 10 указаны средние значения ошибки для одних и тех же реализаций шума.

Таблица 10: MSE разложений ряда  $\mathbf{X} = \mathbf{X}_{\sin} + \mathbf{X}_{\cos} + \mathbf{X}_c + \mathbf{X}_e + \mathbf{X}_{\text{noise}}$  методов

Метод	Параметры	MSE( $\mathbf{X}_c + \mathbf{X}_e$ )	MSE( $\mathbf{X}_{\sin}$ )	MSE( $\mathbf{X}_{\cos}$ )	MSE( $\mathbf{X}$ )
SSA	$L = 96, K = 96$	5.2e-03	2.9e-04	3.6e-04	5.2e-03
SSA EOSSA, $r = 6$	$L = 96, K = 96$	9.5e-04	2.9e-04	3.1e-04	1.5e-03
Fourier	$N = 96 \cdot 2$	1.2e-01	6.9e-04	7.2e-03	1.1e-01
Fourier extended	$N = 96 \cdot 2$	3.0e-03	1.9e-03	9.6e-03	1.2e-02
CiSSA	$L = 96$	5.5e-02	1.7e-04	7.0e-04	4.6e-02
CiSSA extended	$L = 96$	2.7e-03	6.8e-04	2.1e-03	3.1e-03
SSA	$L = 96, K = 97$	5.5e-03	2.9e-04	3.7e-04	5.3e-03
SSA EOSSA, $r = 6$	$L = 96, K = 97$	9.3e-04	2.9e-04	3.1e-04	1.5e-03
Fourier	$N = 96 \cdot 2 - 1$	1.2e-01	1.9e-02	2.2e-02	1.0e-01
Fourier extended	$N = 96 \cdot 2 - 1$	4.7e-03	8.6e-04	3.0e-03	7.8e-03
CiSSA	$L = 97$	7.7e-02	4.1e-02	1.4e-02	1.1e-01
CiSSA extended	$L = 97$	2.7e-03	1.4e-02	3.3e-03	1.7e-02

Как видно из таблицы 10, разделения ухудшились, однако **SSA** с улучшением разделимости EOSSA отработал лучше всех, а хуже всех показали себя алгоритмы Фурье. Также был проведен был проведён двухвыборочный t-критерий для зависимых выборок с целью проверки гипотезы о равенстве средних значений ошибки для каждой компоненты, попарно для всех методов. В качестве нулевой гипотезы ( $H_0$ ) предполагалось, что средние значения двух сравниваемых выборок равны. Критический уровень значимости был установлен на уровне  $\alpha = 0.05$ . Результаты анализа показали, что во всех случаях, кроме сравнения синуса для базового **SSA** и **SSA** с EOSSA, а также синуса для Фурье и расширенного **CiSSA**,  $p$ -значение оказалось меньше 0.05, что позволяет отвергнуть нулевую гипотезу.

Таким образом, можно сделать вывод, что алгоритмы отделяют сигнал от шума примерно одинаково, когда ряд состоит из периодик и параметры правильно подобраны. Результаты не сильно изменяются для базового **SSA** и **SSA** с EOSSA. Для **CiSSA** и Фурье результаты ухудшаются. С добавлением трендовой составляющей ситуация меняется: для **SSA** результаты практически не ухудшаются, **CiSSA** с расширением показывает себя немного хуже **SSA**, остальные алгоритмы выдают значения ошибки на порядки большие.

## 6.7 Разделение непериодических составляющих между собой

Как удалось выяснить, все рассматриваемые алгоритмы могут выделять трендовую составляющую из ряда. Однако лишь **SSA** способен различить между собой две непериодических компоненты. Про скорость асимптотической разделимости для **SSA** можно подробнее узнать в книге [4]. Методы **CiSSA** и Фурье никаким образом не смогут отличить две непериодики между друг другом, поскольку они объединяют компоненты только по частотам. А двум непериодикам соответствуют одинаковые (низкие) наборы частот.

## 6.8 Преимущества и недостатки методов **SSA**, Фурье и **CiSSA**

Для наглядного отображения преимуществ каждого из этих методов составлены похожие по смыслу таблицы 11 и 12, где строки соответствуют методам, а столбцы — условиям (особым видам компонент ряда). Разделение на две таблицы объяснимо тем, что методам **SSA** и **CiSSA** важно, какие  $L$  и  $K$  выбраны у ряда, в то время как разложение Фурье волнует только  $N$ . На пересечении строк и столбцов указан знак, показывающий, достигается ли раз-

деление компоненты: плюс (+) обозначает точное выполнение, знак стремления указывает на асимптотическое выполнение, а минус (−) — на отсутствие разделимости.

Обозначения:

- cos — в ряде присутствуют только периодические компоненты вида  $A \cos(2\pi wx + \varphi)$ ;
- $X_{np1}$  — одна непериодическая компонента в ряде, остальные имеют период;
- $X_{np}$  — несколько непериодических компонент в ряде, остальные имеют период, интересует разделение между непериодическими компонентами;
- group — автоматическая группировка по заданным частотам.

Таблица 11: Преимущества и недостатки методов **SSA**, **CiSSA**

Метод/Условие	cos, $Lw \in \mathbb{N}$ , $Kw \in \mathbb{N}$	cos, $Lw \in \mathbb{N}$ , $Kw \notin \mathbb{N}$	cos, $Lw \notin \mathbb{N}$ , $Kw \notin \mathbb{N}$	$X_{np1}$	$X_{np}$	group
SSA	+	→	→	→	→	—
SSA EOSSA	+	→	→	→	→	+
CiSSA	+	+	→	—	—	+
CiSSA extended	+	+	→	→	—	+

Таблица 12: Преимущества и недостатки методов Fourier

Метод/Условие	cos, $Nw \in \mathbb{N}$	cos, $Nw \notin \mathbb{N}$	$X_{np1}$	$X_{np}$	group
Fourier	+	→	—	—	+
Fourier extended	+	→	→	—	+

Большинство ситуаций из таблицы 11 и 12 уже были разобраны в предыдущих разделах. Так, столбцы, связанные с cos, были разобраны в разделах 6.3 и 6.4. Ситуация с одной непериодической компонентой разобрана в 6.5, а с отделением нескольких непериодик в 6.7. Автоматическая группировка компонент по заранее заданным частотам в 6.1.

Анализ полученных результатов показывает, что **CiSSA** превосходит разложение Фурье как с расширением ряда, так и без него, во всех случаях, когда интересующие частоты совпадают с частотной решеткой алгоритмов. Если же частоты не совпадают, разложение Фурье может дать лучший результат благодаря более детальному разбиению частот. Кроме того, **SSA** с улучшением разделимости EOSSA продемонстрировал более высокую эффективность по сравнению со всеми ранее рассматриваемыми алгоритмами.

## 6.9 Проверка алгоритмов на реальных данных

Теперь рассмотрим реальные данные — месячные ряды промышленного производства (Industrial Production, IP), index 2010 = 100, в США. Данные промышленного производства полезны, по-

скольку оно указывается в определении рецессии Национальным бюро экономических исследований (NBER), как один из четырех ежемесячных рядов индикаторов, которые необходимо проверять при анализе делового цикла. Выборка охватывает период с января 1970 года по сентябрь 2014 года, поэтому размер выборки составляет  $N = 537$ . Источником данных является база данных IMF. Эти показатели демонстрируют различные тенденции, сезонность и цикличность (периодические компоненты, которые соответствуют циклам бизнеса). Данные IP также рассматривались в статье [1]. Применим как **CiSSA** с расширением ряда, так и **SSA** с автоматическим определением частот и улучшениями разделимости EOSSA и FOSSA с параметром  $r = 30$  по следующим группам:

1. Трендовой составляющей должны отвечать низкие частоты, поэтому диапазон:  $[0, \frac{1}{192}]$ ;
2. Циклы бизнеса по диапазонам:  $[\frac{2}{192}, \frac{10}{192}]$ ;
3. Сезонность по частотам  $\omega_k = 1/12, 1/6, 1/4, 1/3, 5/12, 1/2$ ;

На основе предыдущих требований взято  $L = 192$ .

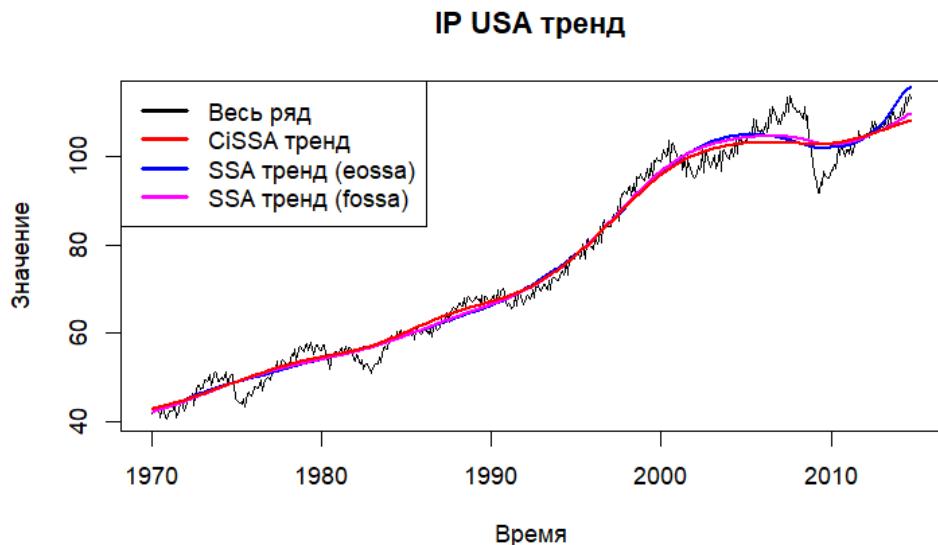


Рис. 8: Трендовая составляющая данных IP USA

При применении FOSSA улучшения разделимости алгоритм **SSA** выделяет тренд довольно похоже с **CiSSA**. Весь график **SSA** тренд EOSSA выглядит более изогнутым при визуальном сравнении с остальными.

### IP USA цикличность

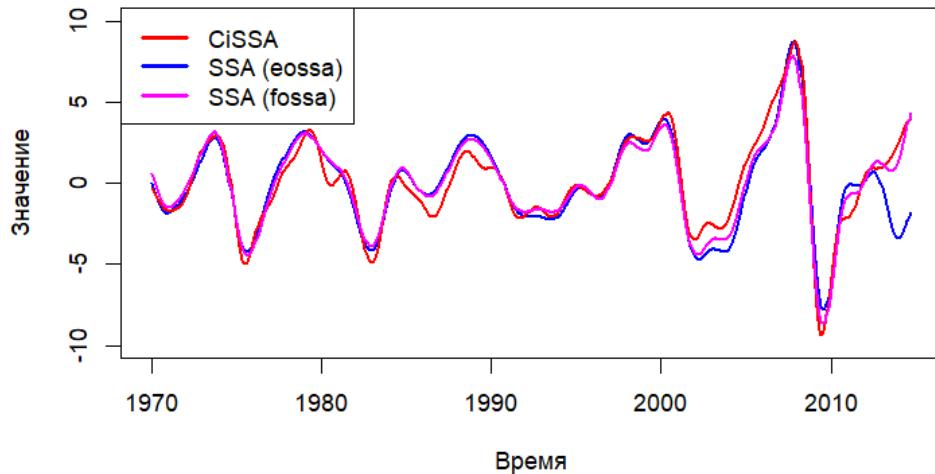


Рис. 9: Циклическая составляющая данных IP USA

Аналогичная тренду ситуация происходит с цикличностью. В случае EOSSA правый хвост (значения ряда после 2010-ого года) смешался между цикличностью и трендом.

### IP USA тренд + цикличность

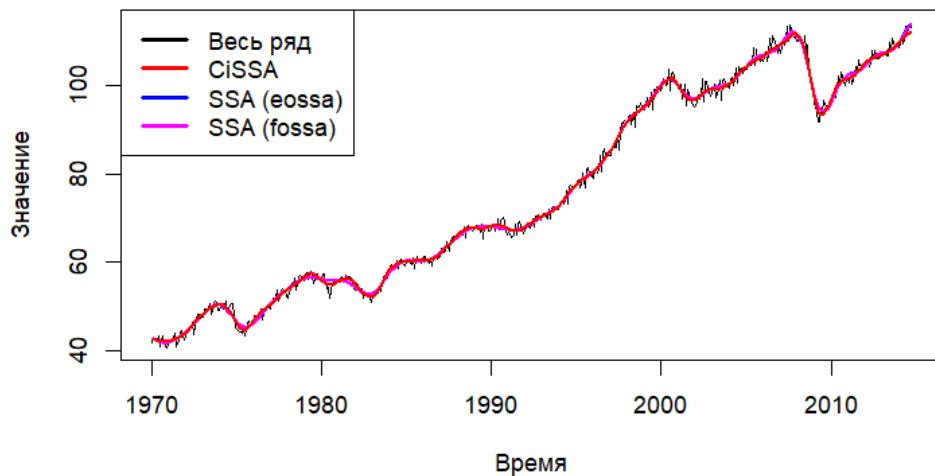


Рис. 10: Объединение тренда и цикличности IP USA

Как видно из графика 10, объединив тренд и цикличность получаем одинаковые результаты для всех рассматриваемых алгоритмов.

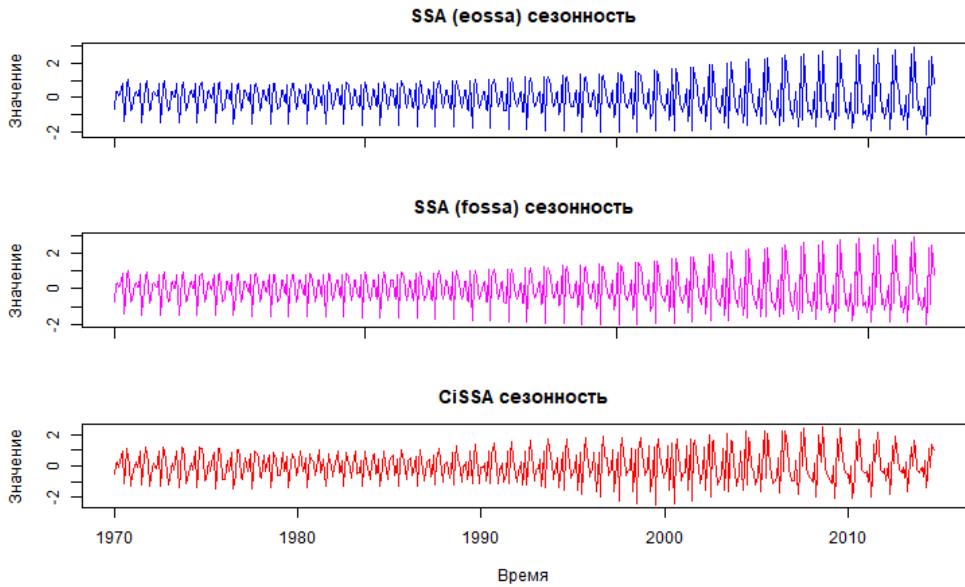


Рис. 11: Сезонная составляющая данных IP USA

Сезонность выглядит для всех алгоритмов похоже.

Шум же является нормальным во всех случаях.

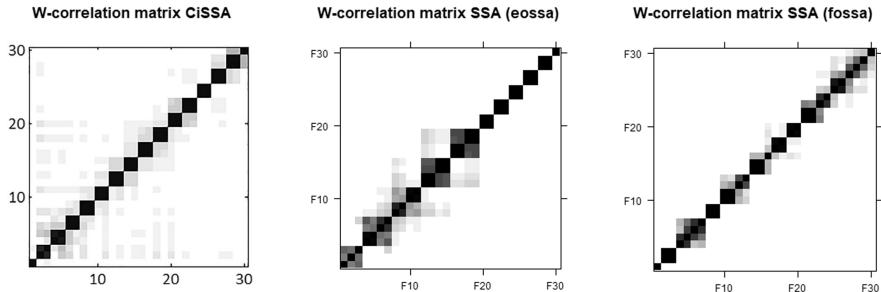


Рис. 12: Матрицы корреляций IP USA

По матрицам корреляции заметно, что при использовании **SSA** с улучшением разделимости EOSSA, смешиваются первые по значимости компоненты ряда (они и являются трендовыми и циклическими).

Таким образом, получились довольно похожие результаты в выделении тренда и цикличности при использовании **SSA** с FOSSA и **CiSSA**. Несколько иные результаты при **SSA** с EOSSA. Сезонная составляющая для всех алгоритмов выглядит схоже.

## 7 Многомерные варианты базового SSA

Временные ряды могут быть не только одномерными, но и многомерными, то есть представлять собой наборы связанных наблюдений. В данной работе рассматриваются две модификации базового **SSA**: **MSSA** и **2d-SSA**.

Под многомерным времененным рядом будем понимать систему  $s$  одномерных временных рядов  $\mathbf{X} = (\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(s)})$ , где каждый ряд  $\mathbf{X}^{(p)} = (x_1^{(p)}, \dots, x_{N_p}^{(p)})$  представляет собой последовательность числовых значений.

Под двумерным рядом будем понимать прямоугольную матрицу  $\mathbf{X} = [x_{ij}] \in \mathbb{R}^{N_x \times N_y}$ , содержащую наблюдения, упорядоченные по двум пространственным или иным измерениям (например, изображение).

### 7.1 MSSA

Рассмотрим многомерный временной ряд, то есть набор  $\{\mathbb{X}^{(p)} = (x_j^{(p)})_{j=1}^{N_p}, p = 1, \dots, s\}$  из  $s$  временных рядов длины  $N_p$ , где  $p = 1, \dots, s$ .

Обозначим  $\mathbb{X} = (\mathbb{X}^{(1)}, \dots, \mathbb{X}^{(s)})$  как исходные данные для алгоритма **MSSA**. Общая схема алгоритма базового **SSA**. Необходимо лишь определить оператор вложения  $\mathcal{J}_{\text{MSSA}}(\mathbb{X}) = \mathbf{X}$ .

#### 7.1.1 Вложение

Пусть  $L$  – длина окна,  $1 < L < \min(N_p, p = 1, \dots, s)$ . Для каждого временного ряда  $\mathbf{X}^{(p)}$  формируем  $K_p = N_p - L + 1$  векторов  $\mathbf{X}_j^{(p)} = (x_j^{(p)}, \dots, x_{j+L-1}^{(p)})^T$ , где  $1 \leq j \leq K_p$ . Обозначим  $K = \sum_{p=1}^s K_p$ . Траекторная матрица многомерного ряда  $\mathbf{X}$  – это матрица размера  $L \times K$  следующего вида:

$$\mathcal{J}_{\text{MSSA}}(\mathbb{X}) = \mathbf{X} = [\mathbf{X}_1^{(1)} : \dots : \mathbf{X}_{K_1}^{(1)} : \dots : \mathbf{X}_1^{(s)} : \dots : \mathbf{X}_{K_s}^{(s)}] = [\mathbf{X}^{(1)} : \dots : \mathbf{X}^{(s)}],$$

где  $\mathbf{X}^{(p)} = \mathcal{J}_{\text{SSA}}(\mathbf{X}^{(p)})$  – траекторная матрица одномерного ряда  $\mathbf{X}^{(p)}$ , определённая в (2.1). Таким образом, траекторная матрица системы временных рядов имеет блочно-Hankel структуру. Заметим, что

$$\mathcal{J}_{\text{MSSA}}^{-1}(\mathbf{X}) = [\mathcal{J}_{\text{SSA}}^{-1}(\mathbf{X}^{(1)}) : \dots : \mathcal{J}_{\text{SSA}}^{-1}(\mathbf{X}^{(s)})].$$

#### 7.1.2 Вложение

Аналогично базовому **SSA**.

### 7.2 Группировка

Аналогично базовому **SSA**.

#### 7.2.1 Диагональное усреднение

Так как  $\mathcal{M}_{L,K}^{(H)}$  в **MSSA** – это множество блочно-Hankel матриц, ортогональный проектор

$\Pi_{\text{stacked } \mathcal{H}}$  на  $\mathcal{M}_{L,K}^{(H)}$  имеет вид

$$\Pi_{\text{stacked } \mathcal{H}}(Y) = [\Pi_{\mathcal{H}}(Y^{(1)}) : \dots : \Pi_{\mathcal{H}}(Y^{(s)})],$$

где  $\Pi_{\mathcal{H}}$  определён в (2.2). Равенство (4.3) следует из общей формы проекции, описанной в разделе 1.1.2.6. Аналогично одномерному случаю, восстановленные ряды получаются с помощью композиции операторов  $\mathcal{T}_{\text{MSSA}}^{-1}$  и  $\Pi_{\text{stacked } \mathcal{H}}$ .

### 7.3 2d-ssa

## 8 Метод Functional singular spectrum analysis (FSSA)

## 9 Заключение

В данной работе исследованы алгоритмы **SSA**, **GSSA** и **CiSSA**. Проведено их сравнение теоретически, и полученные знания были проверены на реальных и смоделированных примерах с помощью языка R. Найдены недостатки и достоинства алгоритмов.

Алгоритм **GSSA** в сравнении с **SSA** лучше справляется с разделимостью компонент между друг другом. Однако это справедливо только тогда, когда в ряде нет шума. Метод **SSA** лучше будет справляться с задачей выделения сигнала.

При сравнении **CiSSA** и **SSA** также выяснилось, что **CiSSA** выделяет трендовую компоненту лучше, чем разложение Фурье, однако проигрывает в выделении периодик, особенно когда частота выделяемой компоненты не попадает в сетку частот методов.

Рассматривая **SSA** с улучшением разделимости и **CiSSA** на модельных примерах, видно, что по среднеквадратической ошибке **SSA** выигрывает у **CiSSA**. Кроме того, алгоритм **SSA** является более гибким: в нем адаптивный базис, есть дополнительные алгоритмы, которые довольно похоже приближают этот алгоритм к **CiSSA**, а также методы для автоматического выбора компонентов по частотам. Метод **CiSSA** является простым в использовании.

Дальнейшими действиями является рассмотрение других модификаций метода **SSA**.

## Список литературы

- [1] Juan Bogalo, Pilar Poncela, and Eva Senra. Circulant singular spectrum analysis: A new automated procedure for signal extraction. *Signal Processing*, 177, 2020.
- [2] Rahul Dey, Meloth Thamban, Chavarukonam Madhavanpillai Laluraj, Kanthanathan Mahalinganathan, Bhikaji Laxman Redkar, Sudhir Kumar, and Kenichi Matsuoka. Application of visual stratigraphy from line-scan images to constrain chronology and melt features of a firn core from coastal antarctica. *Journal of Glaciology*, 69(273):179–190, 2023.
- [3] Nina Golyandina, Pavel Dudnik, and Alex Shlemov. Intelligent identification of trend components in singular spectrum analysis. *Algorithms*, 16(7):353, 2023.
- [4] Nina Golyandina, Vladimir Nekrutkin, and Anatoly Zhigljavsky. *Analysis of Time Series Structure: SSA and Related Techniques*. Chapman and Hall/CRC, 2001.
- [5] Nina Golyandina and Anatoly Zhigljavsky. *Singular Spectrum Analysis for Time Series*. SpringerBriefs in Statistics. Springer Berlin Heidelberg, 2 edition, 2020.
- [6] Nina Golyandina and Polina Zhornikova. On automated identification in singular spectrum analysis for different types of objects, 2023.
- [7] Jialiang Gu, Kevin Hung, Bingo Wing-Kuen Ling, Daniel Hung-Kay Chow, Yang Zhou, Yaru Fu, and Sio Hang Pun. Generalized singular spectrum analysis for the decomposition and analysis of non-stationary signals. *Journal of the Franklin Institute*, Accepted/In Press, 2024.
- [8] Nikolay Pogrebniakov. SPbSU SSA coursework: Time series analysis. [https://github.com/xSICHx/spbu\\_ssa\\_methods\\_coursework/tree/main](https://github.com/xSICHx/spbu_ssa_methods_coursework/tree/main), 2024.
- [9] E.W. Weisstein. *CRC Concise Encyclopedia of Mathematics*. CRC Press, 2002.
- [10] Jammisetty Yedukondalu, Kalyani Sunkara, Vankayalapati Radhika, Sivakrishna Kondaveeti, Murali Anumothu, and Yadadavalli Krishna. Cognitive load detection through eeg lead wise feature optimization and ensemble classification. *Scientific Reports*, 15, 01 2025.