

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

1	Введение	3
2	Базовый метод SSA	5
2.1	Алгоритм метода SSA	5
2.2	Свойства SSA	6
3	Метод Circulant singular spectrum analysis (CiSSA)	10
3.1	Алгоритм метода CiSSA	10
3.2	Свойства	14
4	Метод Generalized singular spectrum analysis (GSSA)	16
4.1	Алгоритм метода GSSA	16
4.2	Свойства GSSA	17
5	Сравнение алгоритмов разложения SSA, GSSA, Фурье и CiSSA	19
5.1	Различия SSA и GSSA	19
5.2	Преимущества и недостатки методов SSA, Фурье и CiSSA	21
5.3	Собственные пространства	24
5.4	Точная разделимость	24
5.5	Асимптотическая разделимость	25
5.6	Отделение сигнала от шума	27
5.7	Автоматическая группировка и проверка на реальных данных	27
5.8	Выводы	30
6	Заключение	31
7	Список литературы	

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 Θ , 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. $\hat{\mathbf{X}}^{(2)}$ является оценкой для $\mathbf{X}^{(2)}$. При этом, 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.

Анализ сингулярного спектра (**SSA** [3]) — метод, целью которого является разложение оригинального ряда на сумму небольшого числа интерпретируемых компонентов, таких как медленно изменяющаяся тенденция (тренд), колебательные компоненты (сезонность) и шум. 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 адаптивного базиса позволяет подстраиваться под любой входной ряд.

В данном исследовании рассматриваются модификации **SSA**, предложенные другими авторами, а именно, **CiSSA** [1] и **GSSA** [5].

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

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**.

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 следующем разделе 3 представлен метод **CiSSA**, также с описанием его основных характеристик. В секции 4 показан **GSSA**. Раздел 5 посвящён сравнению методов **SSA**, разложения Фурье и **CiSSA** на модельных и реальных примерах. В заключительной секции 6 подведены основные итоги исследования.

2 Базовый метод SSA

Рассмотрим базовый метод сингулярного спектрального анализа [3].

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 $\mathbf{S} = \mathbf{X}\mathbf{X}^T$, $\lambda_1, \dots, \lambda_L$ — 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_0, \dots, g_{N-1} :

$$g_k = \begin{cases} \frac{1}{k+1} \sum_{m=1}^{k+1} y_{m,k-m+2}^* & \text{для } 0 \leq k < L^* - 1, \\ \frac{1}{L^*} \sum_{m=1}^{L^*} y_{m,k-m+2}^* & \text{для } L^* - 1 \leq k < K^*, \\ \frac{1}{N-k} \sum_{m=k-K^*+2}^{N-K^*+1} y_{m,k-m+2}^* & \text{для } K^* \leq k < N. \end{cases}$$

Применяя данную операцию к матрицам $\mathbf{X}_{I_1}, \dots, \mathbf{X}_{I_m}$, получают 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, порождаемые строками $\mathbf{X}^{(1)}$ и $\mathbf{X}^{(2)}$ соответственно, ортогональны. То же самое должно выполняться для столбцов [3].

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 $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ [3].

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 [3].

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 $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ [3]. Это будет означать, что для разложения

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

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 [3].

Таблица 1: Точная разделимость

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

Стоит отметить, что $+$ в таблице 1 для $\mathbf{X}_{\cos_1} = A_1 \cos(2\pi\omega_1 n + \varphi_1)$ и $\mathbf{X}_{\cos_2} = A_2 \cos(2\pi\omega_2 n + \varphi_2)$ достигается, если $L\omega_1 \in \mathbb{N}$, $K\omega_1 \in \mathbb{N}$ или $L\omega_2 \in \mathbb{N}$, $K\omega_2 \in \mathbb{N}$, $\omega_1 \neq \omega_2$ [3].

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:

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

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

$$\rho^{(L,K)} \stackrel{\text{def}}{=} \max \left(\max_{1 \leq i,j \leq K} |\rho_{i,j}^{(L)}|, \max_{1 \leq i,j \leq L} |\rho_{i,j}^{(K)}| \right) < \varepsilon [3].$$

Определение 5. 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 [3].

Как можно заметить по таблице 2, для гораздо большего класса функций асимптотическая разделимость имеет место [3].

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. Они позволяют более точно отделять временные подряды друг от друга. В данной работе будут использоваться методы EOSSA и FOSSA. Подробнее про них можно почитать в [2].

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 $X = (\dots, x_{-1}, x_0, x_1, \dots)$. Линейный конечный фильтр — это оператор Φ , который преобразует временной ряд 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 Φ :

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

Определение 8. Амплитудно-частотная характеристика (АЧХ) линейного фильтра Φ :

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

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

Определение 9. Фазово-частотная характеристика (ФЧХ) линейного фильтра Φ :

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

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

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

Теперь рассмотрим алгоритм **SSA** с точки зрения линейных фильтров [4]. 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 [4] (неплохо бы расписать). □

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 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 [3].

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

$$R_X^{(N)}(k, l) \stackrel{\text{def}}{=} \frac{1}{N} \sum_{m=1}^N x_{k+m} x_{l+m} \xrightarrow{N \rightarrow \infty} R_X(k - l). \quad (4)$$

Если (4) выполняется, тогда R_X называется ковариационной функцией стационарного ряда X .

Теорема 2. Пусть R_X — ковариационная функция стационарного ряда X . Тогда существует конечная мера m_X , определенная на борелевских подмножествах $(-1/2, 1/2]$, такая, что

$$R_X(k) = \int_{(-\frac{1}{2}, \frac{1}{2}]} e^{i2\pi k\omega} m_X(d\omega).$$

Мера m_X называется спектральной мерой ряда X .

Доказательство. Доказательство в [3]. □

3.1 Алгоритм метода CiSSA

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

Зафиксируем стационарный временной ряд X состоящий из N элементов и выберем длину окна L .

3.1.1 Вложение

Такой же, как и в **SSA**. Считаем матрицу X , заданную в (1).

3.1.2 Разложение

Для каждого $k = 1 : L$ вычисляются 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 .

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:

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

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

$$\hat{\gamma}_L = \begin{pmatrix} \hat{\gamma}_1 & \hat{\gamma}_2 & \dots & \hat{\gamma}_L \\ \hat{\gamma}_2 & \hat{\gamma}_1 & \dots & \hat{\gamma}_{L-1} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\gamma}_L & \hat{\gamma}_{L-1} & \dots & \hat{\gamma}_1 \end{pmatrix}. \quad (5)$$

Данная матрица $L \times L$ называется Тенлицевой и используется в методе Toeplitz SSA (подробнее про данный метод можно прочитать в книге [3]). На ее основе составим циркулярную матрицу для алгоритма Circulant SSA [1]:

$$\hat{C}_L = \begin{pmatrix} \hat{c}_1 & \hat{c}_2 & \dots & \hat{c}_L \\ \hat{c}_2 & \hat{c}_1 & \dots & \hat{c}_{L-1} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{c}_L & \hat{c}_{L-1} & \dots & \hat{c}_1 \end{pmatrix}, \quad (6)$$

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 (6) задаются по формуле:

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

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

Элементарное разложение

Для каждой частоты $w_k = \frac{k-1}{L}$, $k = 2 : \lfloor \frac{L+1}{2} \rfloor$, 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:

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

3.1.3 Группировка

Такой же шаг, как и в базовом SSA. Однако группировка будет производиться на непересекающиеся подгруппы по частотам от 0 до 0.5, поскольку частоты выше 0.5 представляют собой зеркальное отражение частот ниже 0.5. Именно поэтому объединяются матрицы $\mathbf{X}_{B_k} = \mathbf{X}_k + \mathbf{X}_{L+2-k}$. Разложение $\mathbf{X}_{B_k} = \mathbf{X}_k + \mathbf{X}_{L+2-k} = U_k U_k^H \mathbf{X} + U_{L+2-k} U_{L+2-k}^H \mathbf{X}$, где U^H — это комплексное сопряжение и транспонирование вектора U .

3.1.4 Диагональное усреднение

Такой же шаг, как и в базовом **SSA**.

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 $i = 1$.

$$\begin{aligned} U_k &= L^{-\frac{1}{2}} \left(1, e^{-i2\pi \frac{k-1}{L}}, e^{-i2\pi 2 \frac{k-1}{L}}, \dots, e^{-i2\pi(L-1) \frac{k-1}{L}} \right)^T, \\ U_k^H &= L^{\frac{1}{2}} \left(1, e^{i2\pi \frac{k-1}{L}}, e^{i2\pi 2 \frac{k-1}{L}}, \dots, e^{i2\pi(L-1) \frac{k-1}{L}} \right), \\ L^{-\frac{1}{2}} c_k &= U_k^H X_1 = x_1 + e^{i2\pi \frac{k-1}{L}} x_2 + e^{i2\pi 2 \frac{k-1}{L}} x_3 + \dots + e^{i2\pi(L-1) \frac{k-1}{L}} x_L. \\ X_1^k &= c_k U_k = \left(c_k, c_k e^{-i2\pi \frac{k-1}{L}}, c_k e^{-i2\pi 2 \frac{k-1}{L}}, \dots, c_k e^{-i2\pi(L-1) \frac{k-1}{L}} \right)^T. \end{aligned}$$

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 3.2.1 рассмотрена связь между матрицей \mathbf{X}_{B_k} и разложениями Фурье для векторов вложения.

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 предсказать значения временного ряда за его пределами (экстраполяция) как в правом, так и в левом направлениях на заданное число шагов H . Таким образом, трендовая (нелинейная) компонента ряда будет выделяться заметно лучше. В ходе работы алгоритм выполняет следующие шаги:

1. **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;
2. **Построение дифференцированного ряда:** Временной ряд \mathbf{X} сначала преобразуется в дифференцированный ряд $d\mathbf{X}$, чтобы удалить трендовые компоненты;
3. **Построение AR-модели:** После этого для дифференцированного ряда вычисляются коэффициенты авторегрессионной модели A с использованием метода Юла-Уокера, основываясь на определенном ранее порядке p ;
4. **Правое расширение ряда:** С помощью AR-модели ряд $d\mathbf{X}$ прогнозируется на H шагов вправо. Затем возвращается к своему изначальному состоянию путем интегрирования $d\mathbf{X}$. Получается расширение исходного ряда \mathbf{X} на H шагов вправо;

5. **Левое расширение ряда:** Аналогично предыдущему пункту, ряд прогнозируется на H шагов влево;
6. **Возвращение расширенного ряда:** В конце метод возвращает расширенный временной ряд X_{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.



Рис. 1: 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.

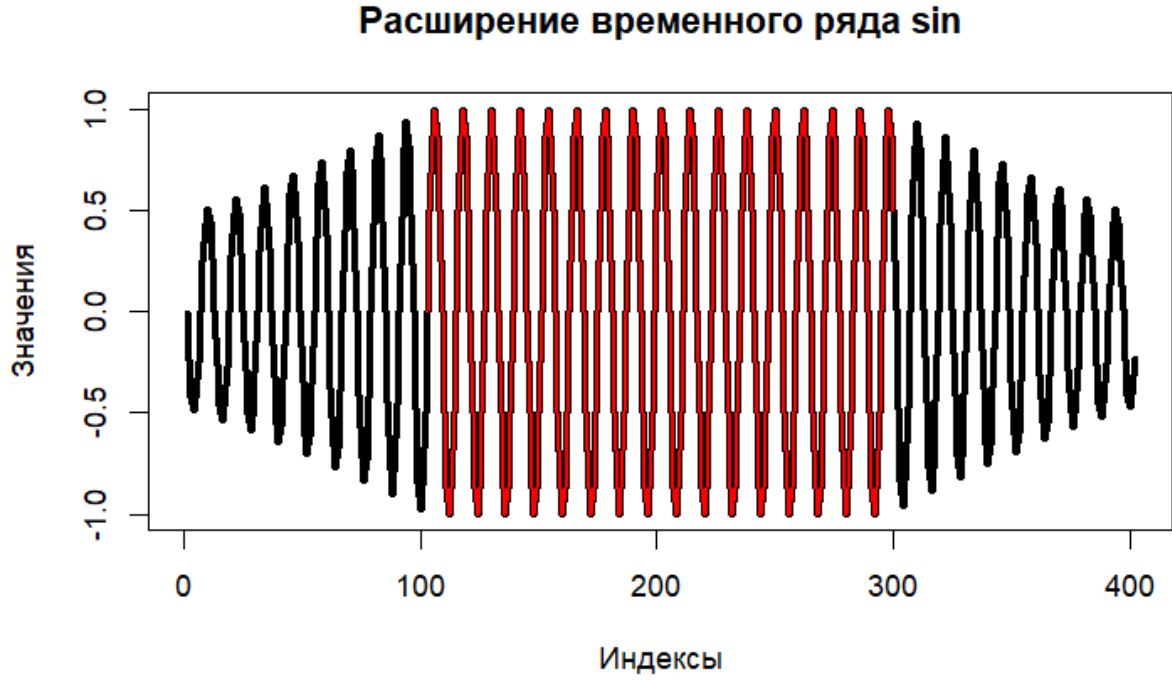


Рис. 2: Расширение временного ряда синуса. Красным показан настоящий ряд, черным — его расширение

На рисунке 2 видно, что синус расширился неправильно, от концов настоящего ряда до концов расширенного значения постепенно уменьшались. Как будет показано в секции 5, это повлияет на значения ошибки.

3.2 Свойства

3.2.1 Связь CiSSA с разложением Фурье

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

Определение 11. *Разложение*

$$x_n = c_0 + \sum_{k=1}^{\lfloor \frac{N+1}{2} \rfloor} (c_k \cos(2\pi nk/N) + s_k \sin(2\pi nk/N)), \quad (7)$$

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 .

3.2.2 Точная разделимость

Поскольку данный метод является аналогом разложения Фурье, то в смысле сильной разделимости можно точно разделить ряд, в котором одной из компонент является $\cos(2\pi\omega + \varphi)$ с частотой ω 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 .

3.2.3 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.

4 Метод Generalized singular spectrum analysis (GSSA)

В этом разделе описана модификация **SSA** на основе добавления весов к строкам L -траекторная матрица \mathbf{X} [5]. Авторы метода называют его обобщенным, поскольку базовый **SSA** является частным случаем **GSSA** с параметром $\alpha = 0$.

4.1 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:

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

4.1.1 Вложение

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

$$\mathbf{X}^{(\alpha)} = \begin{pmatrix} w_1 x_1 & w_1 x_2 & w_1 x_3 & \dots & w_1 x_K \\ w_2 x_2 & w_2 x_3 & w_2 x_4 & \dots & w_2 x_{K+1} \\ w_3 x_3 & w_3 x_4 & w_3 x_5 & \dots & w_3 x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_L x_L & w_L x_{L+1} & w_L x_{L+2} & \dots & w_L x_N \end{pmatrix}. \quad (8)$$

4.1.2 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: $(\sqrt{\lambda^{(\alpha)}}, U^{(\alpha)}, V^{(\alpha)})$.

4.1.3 Группировка

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

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

4.1.4 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_0, \dots, g_{N-1} :

$$g_k = \begin{cases} \frac{1}{\sum_{n=1}^k w_n} \sum_{m=1}^{k+1} y_{m,k-m+2}^* & \text{для } 0 \leq k < L-1, \\ \frac{1}{\sum_{n=1}^L w_n} \sum_{m=1}^L y_{m,k-m+2}^* & \text{для } L-1 \leq k < K, \\ \frac{1}{\sum_{n=k-K+1}^L w_n} \sum_{m=k-K+2}^{N-K+1} y_{m,k-m+2}^* & \text{для } K \leq k < N. \end{cases}$$

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 $\mathbf{X}_1^{(\alpha)} + \dots + \mathbf{X}_m^{(\alpha)} = \mathbf{X}^{(\alpha)}$.

4.2 Свойства GSSA

4.2.1 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 (8) $\mathbf{X}^{(\alpha)}$ можно получить, что $\mathbf{X}^{(\alpha)} = \text{diag}(w_1, w_2, \dots, w_L) \mathbf{X} = \text{diag}(\mathbf{w}^{(\alpha)}) \mathbf{X}$. Поскольку матрица $\text{diag}(\mathbf{w}^{(\alpha)})$ имеет ранг равный L , она диагональна, то и $\text{rank } \mathbf{X}^{(\alpha)} = \text{rank } \text{diag}(\mathbf{w}^{(\alpha)}) \mathbf{X} = \text{rank } \mathbf{X}$.

4.2.2 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 $K \times N$:

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

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

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

Теорема 3. Компонента временного ряда $\tilde{\mathbf{X}}$, восстановленная с использованием собственной тройки $(\sqrt{\lambda^{(\alpha)}}, U^{(\alpha)}, V^{(\alpha)})$, имеет вид:

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

Доказательство. Доказательство проводится аналогично доказательству теоремы 1. \square

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

$$\tilde{x}_s = \sum_{j=-(L-1)}^{L-1} \left(\sum_{k=1}^{L-|j|} u_k^{(\alpha)} u_{k+|j|}^{(\alpha)} w_k / \sum_{i=1}^L w_i \right) x_{s-j}, \quad L \leq s \leq K. \quad (9)$$

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

5 Сравнение алгоритмов разложения SSA, GSSA, Фурье и CiSSA

Все вычисления, а также код методов **CiSSA** и **GSSA** можно найти в github репозитории [6].

5.1 Различия SSA и GSSA

В данном разделе сравниваются алгоритмы базового **SSA** и **GSSA** с параметром $\alpha \neq 0$. Чтобы понять их принципиальное отличие, рассмотрим методы с точки зрения линейных фильтров: по представлениям (3) и (9) 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 α равных 0 (базовый **SSA**), $\frac{1}{2}$, 1, 2:

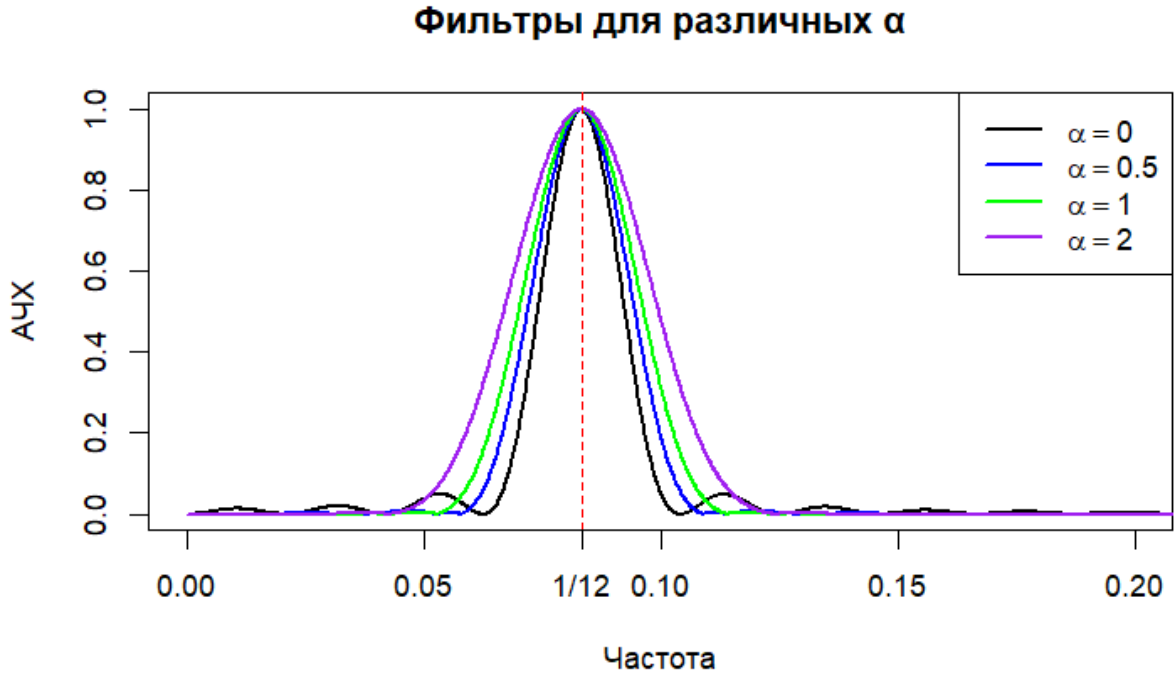


Рис. 3: АЧХ фильтров, отвечающих за $X = \sin\left(\frac{2\pi}{12}x\right)$, при разных α

На рисунке 3 показано, как фильтры ведут себя для различных значений параметра α . Для всех рассмотренных значений α фильтры подавляют частоты, значительно отличающиеся от частоты синуса $\omega = \frac{1}{12}$. При малых значениях α , таких как $\alpha = 0$, наблюдается волнообразное поведение фильтра, что указывает на частичное захватывание соседних частот, хотя и не близких к частоте синуса. С увеличением α 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, $N = 96 \cdot 2 - 1$, $L = 48$.

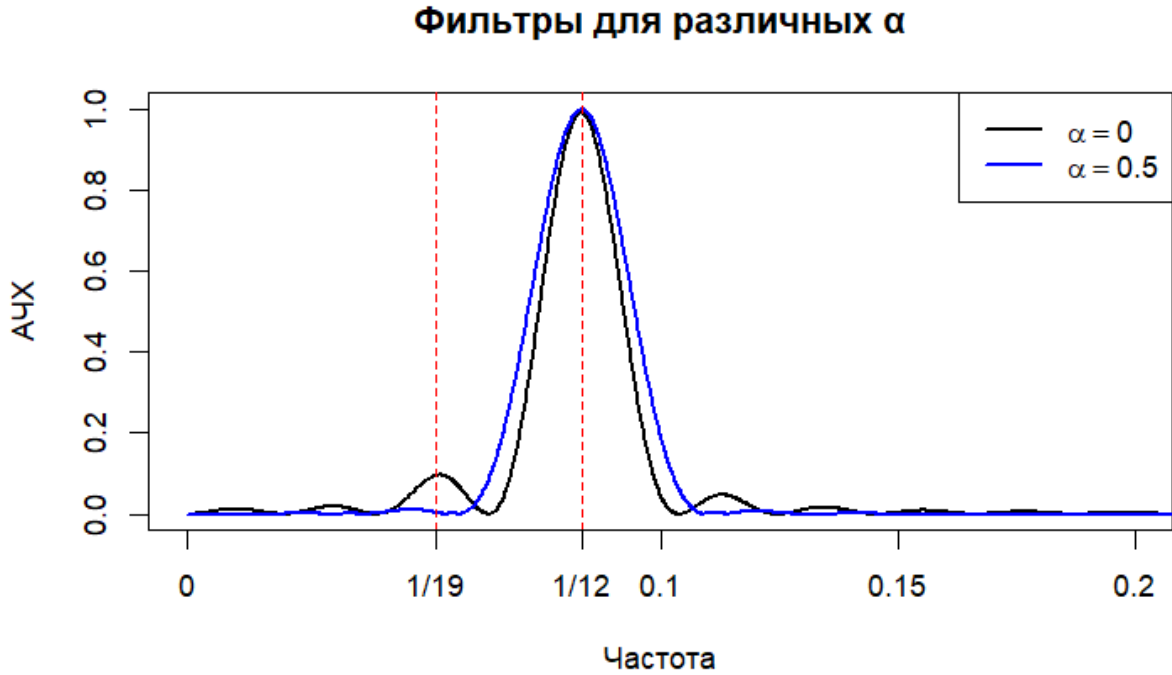


Рис. 4: Ряд $X = X_{\sin} + X_{\cos}$. АЧХ фильтров, отвечающих за $X_{\sin} = \sin\left(\frac{2\pi}{12}n\right)$, при разных α

По рисунку 4 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:

Метод/Ошибка	X_{\sin}	X_{\cos}	X
SSA	5.15e-03	5.15e-03	6.01e-30
GSSA, $\alpha = \frac{1}{2}$	3.68e-04	3.68e-04	9.53e-30

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

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

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

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

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.

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

5.2 Преимущества и недостатки методов SSA, Фурье и CiSSA

В данной секции проводится сравнение различных методов: базовый **SSA**, **SSA** с использованием EOSSA для улучшения разделимости, разложения Фурье и Фурье с расширением ряда, базового **CiSSA** и **CiSSA** с расширением ряда. Для наглядного отображения преимуществ каждого из этих методов составлена таблица 5, где строки соответствуют методам, а столбцы — условиям (особым видам компонент ряда). На пересечении строк и столбцов указан знак, показывающий, достигается ли разделение компоненты: плюс (+) обозначает точное выполнение, знак стремления указывает на асимптотическое выполнение, а минус (−) — на отсутствие разделимости. Для разложения Фурье подразумевается, что $L = N$.

Обозначения:

- \cos — в ряде присутствуют только периодические компоненты вида $\cos(2\pi\omega x + \varphi)$;
- 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.

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

Таблица 5: Преимущества и недостатки методов

На основе таблицы 5 были выбраны примеры, следующие ниже.

Данные методы разложения временного ряда должны совпадать, если ряд состоит только из периодических компонент. Например, пусть $X = X_{\sin} + X_{\cos} = \sin \frac{2\pi}{12}x + \frac{1}{2} \cos \frac{2\pi}{3}x$, $L = 96$,

$N = 96 \cdot 2$ для разложения Фурье и $N = 96 \cdot 2 - 1$ для остальных, чтобы выполнялись условия выполнения разделимости частот. Сравним результаты по среднеквадратичной ошибке:

Метод/Компонента	X_{\sin}	X_{\cos}	X
SSA	6.8e-30	1.5e-29	1.8e-29
SSA EOSSA	1.5e-29	7.5e-30	2.0e-29
Fourier	1.7e-28	3.5e-28	5.1e-28
Fourier extended	6.2e-04	2.6e-03	3.2e-03
CiSSA	1.9e-29	5.3e-30	2.1e-29
CiSSA extended	2.0e-04	8.6e-04	1.1e-03

Таблица 6: MSE разложений ряда $X = X_{\sin} + X_{\cos}$

Таблица 6 показывает, что разложения без расширений ряда сделали правильное (с точностью до вычислений с помощью компьютера) разделение компонент ряда. Однако расширение в методах **CiSSA** и Фурье ухудшило разделимость периодических частей.

Теперь добавим к этому ряду шум: $X = X_{\sin} + X_{\cos} + X_{\text{noise}} = \sin \frac{2\pi}{12}x + \frac{1}{2} \cos \frac{2\pi}{3}x + \varepsilon_n$, где $\varepsilon_n \sim N(0, 0.1)$, $L = 96$, $N = 96 \cdot 2$ для разложения Фурье и $N = 96 \cdot 2 - 1$ для остальных. Результаты должны ухудшиться. Проводилось 100 тестов, в таблице 7 указаны средние значения ошибки для одних и тех же реализаций шума.

Метод/Компонента	X_{\sin}	X_{\cos}	X
SSA	2.9e-04	3.1e-04	5.9e-04
SSA EOSSA	2.9e-04	3.1e-04	5.9e-04
Fourier	1.0e-04	1.1e-04	2.2e-04
Fourier extended	1.2e-03	3.9e-03	5.1e-03
CiSSA	1.6e-04	1.8e-04	3.4e-04
CiSSA extended	6.6e-04	1.9e-03	2.5e-03

Таблица 7: MSE разложений ряда $X = X_{\sin} + X_{\cos} + X_{\text{noise}}$

По таблице 7 видно, что зашумление ряда дало негативный эффект на ошибку. 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.

Попробуем добавить к ряду непериодическую компоненту. $X = X_{\sin} + X_{\cos} + X_c + X_e = \sin \frac{2\pi}{12}x + \frac{1}{2} \cos \frac{2\pi}{3}x + 1 + e^{\frac{x}{100}}$, $L = 96$, $N = 96 \cdot 2$ для разложения Фурье и $N = 96 \cdot 2 - 1$. Непериодические компоненты будут отвечать низким частотам. Проблема лишь в том, что с помощью методов разложения Фурье **CiSSA** невозможно различить между собой две непериодические компоненты, поскольку группировка работает по частотам, элементы разложения неизбежно смешаются между собой. Будем искать экспоненту и константу по низким частотам, назовем это трендовой составляющей ряда. По таблице 5 лучше всех должен справиться **SSA** с улучшением разделимости EOSSA. Хуже всех — разложение Фурье, поскольку он никаким образом не сможет вычленить из ряда экспоненту.

Метод/Компонента	$X_c + X_e$	X_{\sin}	X_{\cos}	X
SSA	5.0e-03	8.9e-07	5.2e-05	4.4e-03
SSA EOSSA	1.7e-28	1.6e-29	8.7e-30	1.6e-28
Fourier	1.1e-01	6.1e-04	6.8e-03	1.1e-01
Fourier extended	1.4e-03	1.3e-03	8.4e-03	9.6e-03
CiSSA	5.3e-02	1.6e-05	4.9e-04	4.4e-02
CiSSA extended	5.0e-04	2.1e-04	1.1e-03	6.0e-04

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

Результаты таблицы 8 повторяют вышеизложенные рассуждения. Также заметно, что периодические компоненты лучше выделились с помощью **CiSSA** без процедуры расширения ряда в сравнении с **CiSSA** с расширением.

Теперь добавим шум в предыдущий пример. Результаты всех разложений должны ухудшиться. $X = X_{\sin} + X_{\cos} + X_c + X_e + X_{\text{noise}} = \sin \frac{2\pi}{12}x + \frac{1}{2} \cos \frac{2\pi}{3}x + 1 + e^{\frac{x}{100}} + N(0, 0.1)$, $L = 96$, $N = 96 \cdot 2$ для разложения Фурье и $N = 96 \cdot 2 - 1$. Было проведено 100 тестов, в таблице 9 указаны средние значения ошибки.

Метод/Компонента	$X_c + X_e$	X_{\sin}	X_{\cos}	X
SSA	5.2e-03	2.9e-04	3.6e-04	5.2e-03
SSA EOSSA	9.5e-04	2.9e-04	3.1e-04	1.5e-03
Fourier	1.2e-01	6.9e-04	7.2e-03	1.1e-01
Fourier extended	3.0e-03	1.9e-03	9.6e-03	1.2e-02
CiSSA	5.5e-02	1.7e-04	7.0e-04	4.6e-02
CiSSA extended	2.7e-03	6.8e-04	2.1e-03	3.1e-03

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

Метод	exp_err	sin_err	cos_err	all_err
SSA	5.2e-03	2.9e-04	3.6e-04	5.2e-03
SSA EOSSA	9.5e-04	2.9e-04	3.1e-04	1.5e-03
Fourier	1.2e-01	6.9e-04	7.2e-03	1.1e-01
CiSSA	5.5e-02	1.7e-04	7.0e-04	4.6e-02
CiSSA extended	2.7e-03	6.8e-04	2.1e-03	3.1e-03
Fourier extended	3.0e-03	1.9e-03	9.6e-03	1.2e-02

Таблица 10: Example Table

Как видно из таблицы 9, разделения ухудшились, однако **SSA** с улучшением разделимости EOSSA отработал лучше всех. Также был проведен 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.

По результатам данных примеров и таблицы 5, можно понять, что **CiSSA** работает лучше, чем разложение Фурье, как при расширении ряда, так и без него. Однако это не удивительно, ведь разложение Фурье это частный случай **CiSSA** при $L = N$. А **SSA** с улучшением разделимости EOSSA показал себя лучше базового **SSA**. Таким образом, далее не будем рассматривать разложение Фурье и базовый **SSA**, остановимся на **SSA** с EOSSA, **CiSSA** с расширением и без него. Кроме того, по умолчанию будет использоваться **CiSSA** с расширением, если есть непериодичность, и обычный **CiSSA**, если все компоненты периодичны. Также при написании **SSA** будет подразумеваться использование **SSA** с EOSSA, если нет конкретных указаний.

5.3 Собственные пространства

Каждый алгоритм после группировки порождает построенными матрицами собственные подпространства. В случае базового **SSA** алгоритма базис подпространств является адаптивным, то есть зависящим от X, L, N . Таким образом, **SSA** может отличить, например, произведение полиномов, экспонент и косинусов друг от друга.

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

5.4 Точная разделимость

Как удалось выяснить, классов точной разделимости больше в базовом алгоритме **SSA**, однако в случае разделения \cos , условия менее жесткие при использовании **CiSSA**.

Проверим на примерах. Возьмем временной ряд, с разложением которого оба алгоритма должны справиться: $X = X_C + X_{\cos} = 1 + \cos(\frac{2\pi}{12}x)$, $L = 96 \mid 12$, $N = 96 \cdot 2 - 1$, $K = 96 \mid 12$. Будем считать MSE между настоящими компонентами ряда и вычисленными.

Метод/Компонента	X_C	X_{\cos}
SSA	2.1e-30	4.9e-30
CiSSA	3.6e-31	5.2e-30

Таблица 11: MSE разложений ряда $X = X_C + X_{\cos}$, $\omega K \in \mathbb{N}$.

Ошибки таблицы 11 можно посчитать за погрешность вычислений на компьютере.

Теперь возьмем временной ряд, при котором **SSA** должен отработать хуже **CiSSA**: $X = X_C + X_{\cos} = 1 + \cos(\frac{2\pi}{12}x)$, $L = 96 \mid 12$, $N = 96 \cdot 2 + 5$, $K = 102 \nmid 12$. Поскольку K не делится на частоту косинуса, условия точной разделимости в **SSA** не выполняются. Будем считать MSE между настоящими компонентами ряда и вычисленными.

Метод/Компонента	X_C	X_{\cos}
SSA	9.5e-5	9.6e-5
CiSSA	3.2e-31	5.1e-30

Таблица 12: MSE разложений ряда $X = X_C + X_{\cos}$, $\omega K \notin \mathbb{N}$.

Таким образом, с разделением косинуса от константы лучше справился алгоритм **CiSSA**, поскольку в нем требуется меньше условий на параметры алгоритма.

5.5 Асимптотическая разделимость

Как было сказано, асимптотически разделимы в методе **SSA** полиномы, гармонические функции (косинус, косинус помноженный на экспоненту, экспонента) [3]. 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 отделен косинус, но постоянно его увеличивать, в конечном счете получится снизить ошибку выделения нужной компоненты косинуса, если брать соседние частоты с частотой компоненты. Однако в этом случае нужно выбирать диапазон частот, которые стоит объединить.

Непериодические компоненты повлияют на ошибку разложений всего временного ряда, они смешаются и их уже никак не получится отделить методом **CiSSA**. Рассмотрим более детально пример с экспонентой: $X = X_c + X_e + X_{\cos} + X_{\sin} = 1 + e^{\frac{x}{100}} + \cos(\frac{2\pi}{12}x) + \sin(\frac{2\pi}{24}x)$, $N = 96 \cdot 2 - 1$, $L = 96$, можно получить следующие результаты:

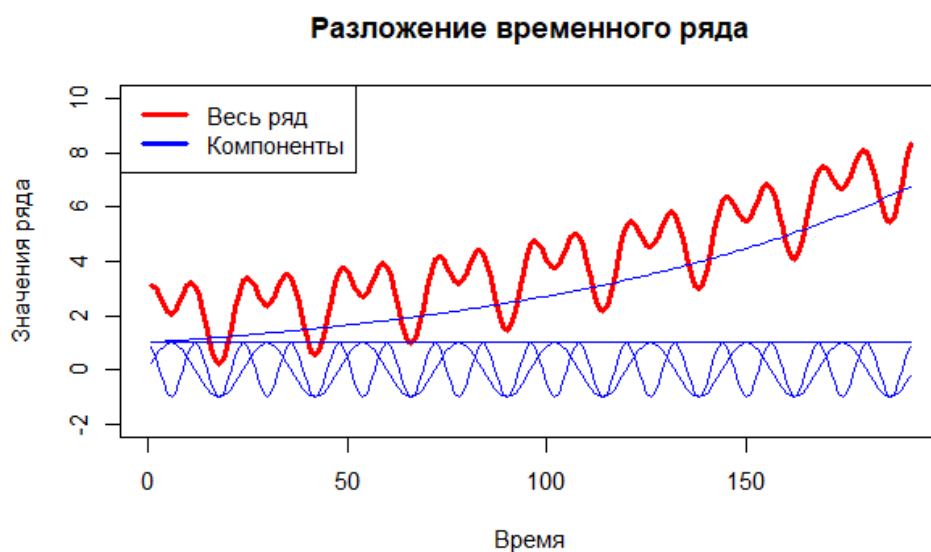


Рис. 5: Правильное разложение ряда $X = X_c + X_e + X_{\cos} + X_{\sin}$

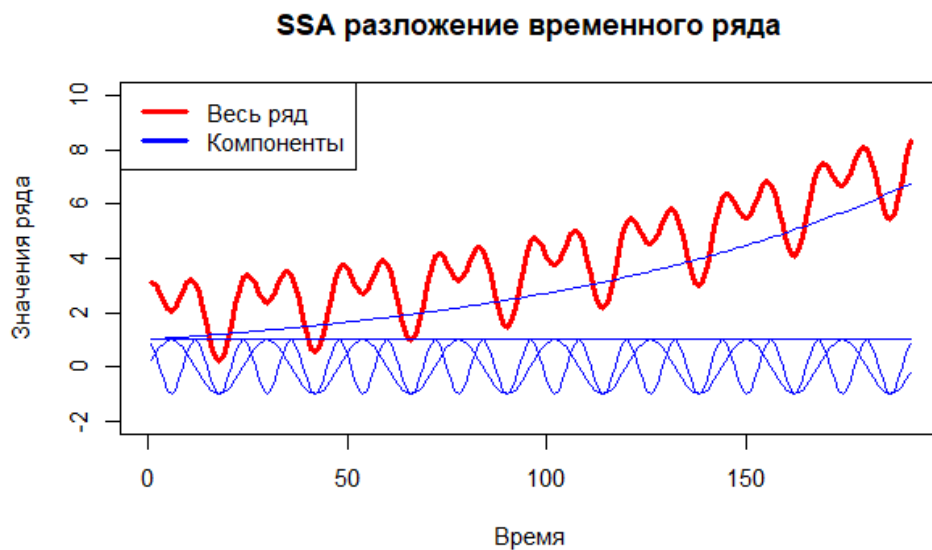


Рис. 6: Разложение ряда $X = X_c + X_e + X_{\cos} + X_{\sin}$ методом **SSA**

Метод **SSA** разделил правильно все компоненты друг от друга.

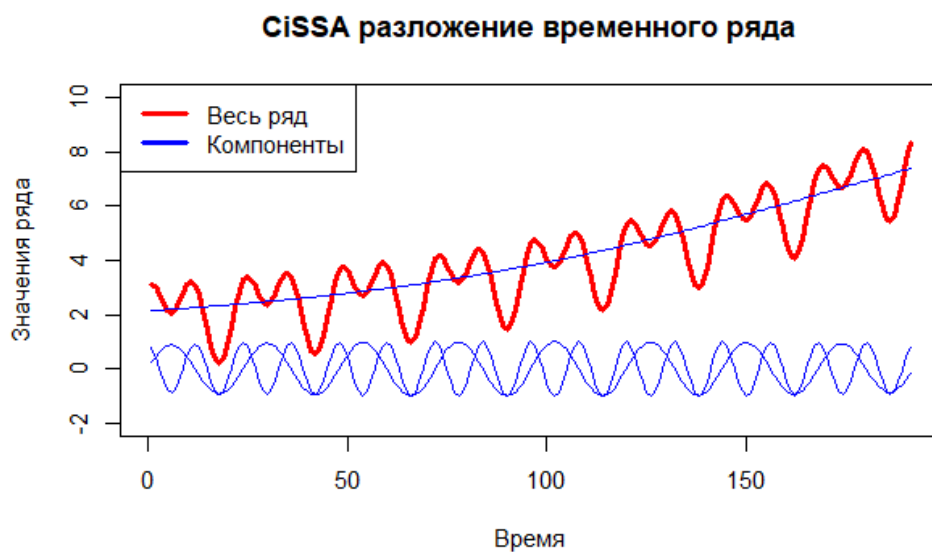


Рис. 7: Разложение ряда $X = X_c + X_e + X_{\cos} + X_{\sin}$ методом **CiSSA**

В случае **CiSSA** получилось так, что экспонента и константа смешались в одну компоненту, одни и те же низкие частоты отвечают за них одновременно, поскольку они не являются периодиками.

Метод/Компонента	X_e	X_c	$X_c + X_e$	X_{\sin}	X_{\cos}
SSA	2.2e-25	2.2e-25	4.2e-28	3.8e-29	1.6e-29
CiSSA	none	none	3.5e-02	1.4e-04	1.9e-03

Таблица 13: MSE разложений ряда $X = X_c + X_e + X_{\cos} + X_{\sin}$ методов **SSA** и **CiSSA**

Таблица 13 и рисунки 6, 7 показывают, что метод **SSA** справился лучше в сравнении с **CiSSA**, причем как по разделимости, так и по ошибке. В алгоритме **CiSSA** трендовая составляющая также смешалась с сезонной, поэтому увеличилась ошибка при косинусе. Стоит отметить, что в данном примере использовался алгоритм улучшения разделимости EOSSA [2] для метода **SSA**. Без него не получились бы такие результаты.

Или же, если заменить X_e на $X_{e-\cos}$, то есть теперь ряд $X = X_c + X_{e-\cos} + X_{\cos} + X_{\sin} = 1 + e^{\frac{x}{100}} \cos(\frac{2\pi}{48}x) + \cos(\frac{2\pi}{12}x) + \sin(\frac{2\pi}{24}x)$, то получится следующая таблица ошибок:

Метод/Компонента	$X_{e-\cos}$	X_{\sin}	X_{\cos}
SSA	4.7e-29	1.1e-29	8.4e-30
CiSSA	3.2e-02	2.6e-04	5.8e-03

Таблица 14: MSE разложений ряда $X = X_c + X_{e-\cos} + X_{\cos} + X_{\sin}$ методов **SSA** и **CiSSA**

Таким образом, таблица 14 показывает тот же недостаток у метода **CiSSA**, что и таблица 13.

5.6 Отделение сигнала от шума

Рассматривая ряд из предыдущего пункта, добавим к нему гауссовский шум с стандартным отклонением 0.1: $X = X_c + X_e + X_{\cos} + X_{\sin} + X_{\text{noise}} = 1 + e^{\frac{x}{100}} + \cos(\frac{2\pi}{12}x) + \sin(\frac{2\pi}{24}x) + N(0, 0.1^2)$, $N = 96 \cdot 2 - 1$, $L = 96$. Сделав такой тест 10000 раз, получим следующий результат по ошибке MSE между настоящим сигналом и его оценкой:

Метод/Статистики	min	median	mean	max	sd
SSA	5.8e-04	2.0e-03	2.1e-03	4.9e-03	6.2e-04
CiSSA	2.5e-02	3.4e-02	3.4e-02	4.9e-02	3.7e-03

Таблица 15: Данные по распределению ошибки восстановления сигнала разложений методов **SSA** и **CiSSA**

По таблице 15 можно увидеть что метод **SSA** отработал лучше **CiSSA**.

5.7 Автоматическая группировка и проверка на реальных данных

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**. При этом, прежде чем применять его, стоит выполнить процедуру улучшения разделимости. В данной работе будут использоваться методы EOSSA и FOSSA [2].

Сравним работы этих алгоритмов сначала на модельных примерах, затем на реальных данных.

Используем ряд из секции 5.5 про асимптотическую разделимость: $X = X_c + X_e + X_{\cos} + X_{\sin} = 1 + e^{\frac{x}{100}} + \cos(\frac{2\pi}{12}x) + \sin(\frac{2\pi}{24}x)$, $N = 96 \cdot 2 - 1$, $L = 96$. Применяем алгоритм EOSSA [2] для лучшей разделимости, количество компонент для улучшения в EOSSA $r = 10$ и выбираем в качестве интересующих частот диапазоны $(\frac{1}{24} - \varepsilon, \frac{1}{24} + \varepsilon)$, $(\frac{1}{12} - \varepsilon, \frac{1}{12} + \varepsilon)$, $\varepsilon = \frac{1}{97}$. Результаты остаются теми же, как и в таблице 13 и рисунках 6, 7, однако теперь группировка ряда произошла по интересующим частотам.

Теперь рассмотрим реальные данные — месячные ряды промышленного производства (Industrial Production, IP), index 2010 = 100, в США. Данные промышленного производства полезны, поскольку 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**, так и **SSA** с автоматическим определением частот и улучшением разделимости 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 $L = 192$.

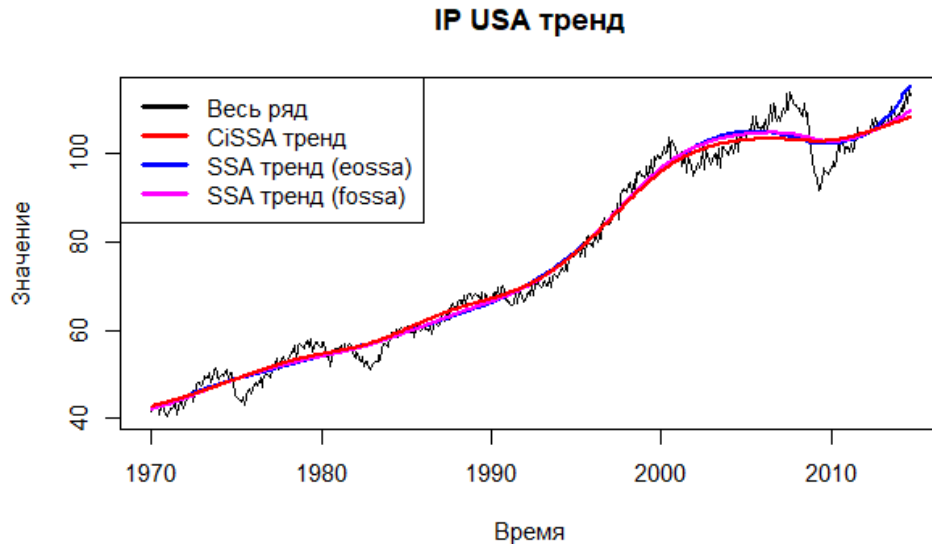


Рис. 8: Трендовая составляющая данных IP USA

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.

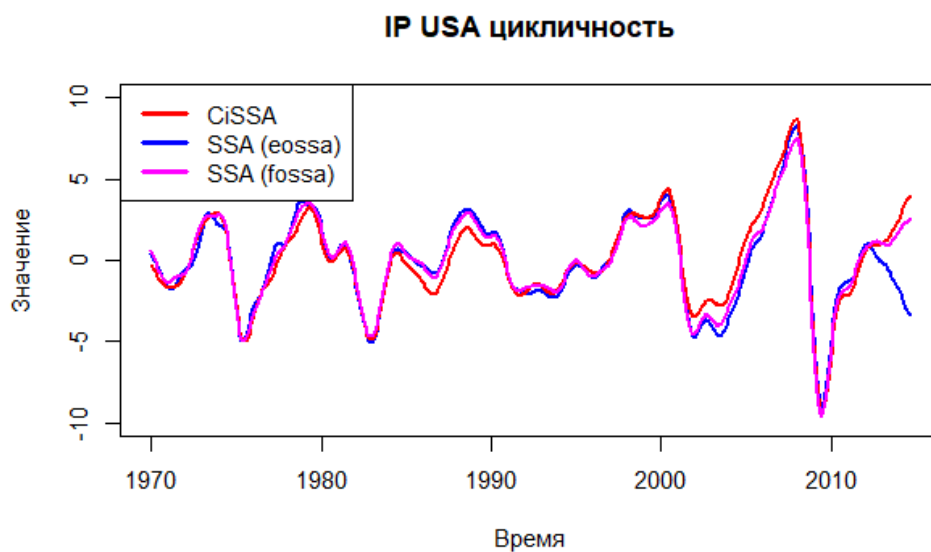


Рис. 9: Циклическая составляющая данных IP USA

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

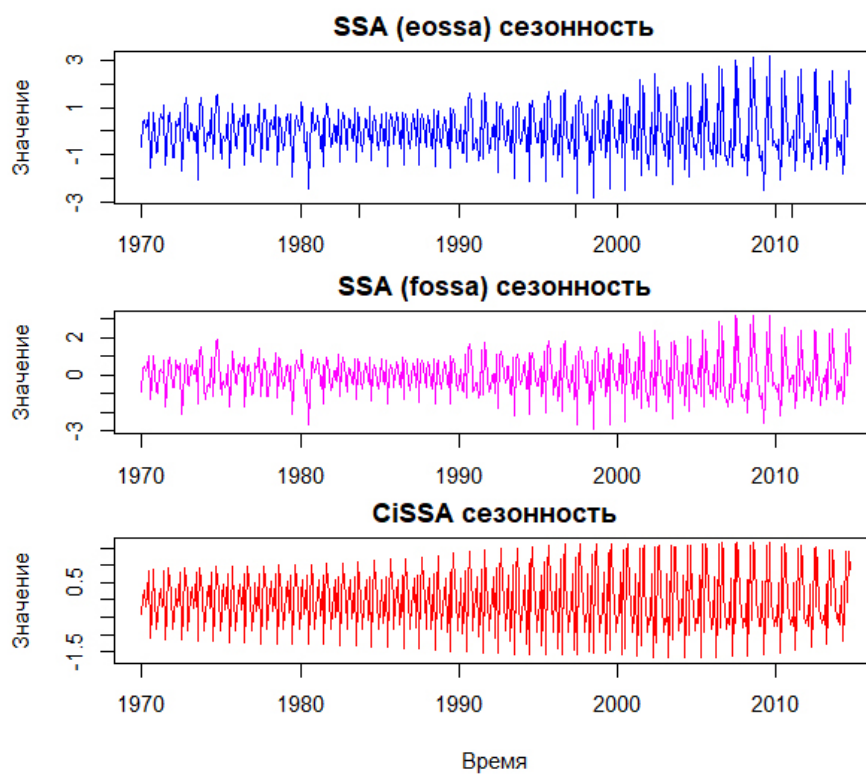


Рис. 10: Сезонная составляющая данных IP USA

Поскольку в базовом **SSA** адаптивный базис, сезонность является менее систематичной, разброс значений выше по сравнению с **CiSSA**.

Шум же является нормальным во всех случаях.

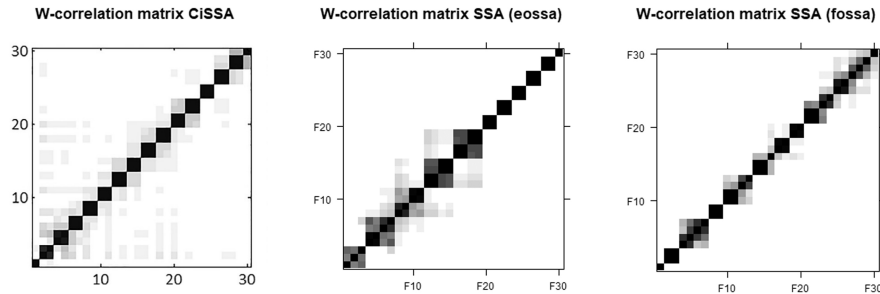


Рис. 11: Матрицы корреляций IP USA

По матрицам корреляции заметно, что при использовании **SSA** с улучшением разделимости EOSSA, сильно 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 в силу неадаптивного базиса более строго выглядит для метода **CiSSA**.

5.8 Выводы

По полученным результатам, можно следующие выводы:

1. Алгоритм **CiSSA** работает лучше разложения Фурье;
2. Если понятно, что ряд состоит только из периодических компонент, стоит использовать **CiSSA** без процедуры расширения, поскольку она делает ошибки разделений периодики больше. И напротив, если есть непериодичность, лучше расширять ряд;
3. Если данные зашумлены или имеется непериодичность, алгоритм **SSA** с улучшением разделимости справляется в среднеквадратичном лучше **CiSSA** с расширением ряда или без.

6 Заключение

В данной работе исследован алгоритм **CiSSA**, сравнены методы **CiSSA** и **SSA**, и полученные знания были проверены на реальных и смоделированных примерах с помощью языка R. Оба алгоритма справляются с поставленными задачами, существенным различием является то, что 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] Nina Golyandina, Pavel Dudnik, and Alex Shlemov. Intelligent identification of trend components in singular spectrum analysis. *Algorithms*, 16(7):353, 2023.
- [3] Nina Golyandina, Vladimir Nekrutkin, and Anatoly Zhigljavsky. *Analysis of Time Series Structure: SSA and Related Techniques*. Chapman and Hall/CRC, 2001.
- [4] Nina Golyandina and Anatoly Zhigljavsky. *Singular Spectrum Analysis for Time Series*. SpringerBriefs in Statistics. Springer Berlin Heidelberg, 2 edition, 2020.
- [5] 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.
- [6] Nikolay Pogrebnikov. SPbSU SSA coursework: Time series analysis. https://github.com/xSICHx/spbu_ssa_methods_coursework/tree/main, 2024.