Электронный научно-технический журнал Октябрь 2013
http://www.bru.mogilev.by

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ АНАЛИЗА ВРЕМЕННЫХ РЯДОВ МЕТОДОМ СИНГУЛЯРНОГО СПЕКТРАЛЬНОГО АНАЛИЗА[1]

Н.М. Чапаров, **Е.А. Якимов, А.И. Якимов**

Аннотация. В статье представлен программный модуль для сингулярного спектрального анализа числовых последовательностей данных. Дано описание программной реализации этапов вложения, сингулярного разложения, группировки, диагонального усреднения. Приведены технические характеристики программного обеспечения.

Ключевые слова: программный модуль, сингулярный спектральный анализ, трендовая составляющая, гармоническая составляющая, шумовая составляющая.

1. Введение

Сингулярный спектральный анализ (ССА) [1] является полезным инструментом, который может использоваться для того, чтобы решить следующие задачи: обнаружение тенденций в различных областях; сглаживание; извлечение (обнаружение) компонентов сезонности; одновременное извлечение (распознание) циклов с маленькими и большими периодами; извлечение периодичности с переменной амплитудой; одновременное извлечение (распознание) сложных тенденций и периодичностей; обнаружение структуры в коротких временных последовательностях; определение точки изменения тенденции. Решение всех этих задач соответствует основным возможностям ССА. Чтобы достигнуть вышеупомянутых возможностей ССА, нет необходимости знать параметрическую модель рассматриваемых временных рядов.

Одной из конечных целей разработки программного обеспечения (ПО) является выделение трендовых, периодических и шумовых составляющих временного ряда. Входные данные должны храниться в текстовом файле или XML-файле, выходные данные – трендовая, периодическая и шумовая составляющие в виде рядов в текстовом файле или XML-файле с возможностью сохранения их графиков.

2. Содержательное описание сингулярного спектрального анализа

Математической основой ССА является сингулярное разложение. Для успешного применения ССА следует последовательно пройти несколько шагов.

Вложение. На этом шаге выбирается ширина окна L, от выбора которой зависят результаты применения ССА. Из-за того, что нет общих рекомендаций по выбору ширины окна, параметр L зависит от решаемой задачи и предварительной информации, известной о временном ряде. Например, для выделения тренда рекомендуется выбирать ширину окна не слишком большой. С другой стороны, для выделения гармонических колебаний рекомендуется большая ширина окна, можно выполнять два прохода с разной шириной окна. После выбора ширины окна в соответствии с L строится траекторная матрица A ряда, которая будет являться по условию ее построения ганкелевой.

Сингулярное разложение. Для матрицы $S = A \cdot A^{\mathrm{T}}$ находятся собственные числа μ и ортонормированные собственные векторы $\mathbf{U}$. Упорядоченные по убыванию собствен-

[1] Работа выполнена при финансовой поддержке ГПНИ по заданию «Информатика и космос 1.3.02»

ные числа, которые больше нуля, часто называются сингулярными числами, а соответствующие им собственные векторы – левыми сингулярными векторами **U**. После этого вычисляются векторы **V**, которые называются правыми сингулярными векторами, и находятся элементарные матрицы, на сумму которых раскладывается первоначальная траекторная матрица.

Группировка. На данном этапе элементарные матрицы группируются по принципу принадлежности к тренду, гармоническим колебаниям или к шуму. Этот этап является наиболее сложным при применении ССА. Для нахождения тренда на диаграммах собственных векторов (по оси абсцисс откладывается порядковый номер координаты собственного вектора, а по оси ординат – значение координаты собственного вектора) выделяют медленно меняющиеся векторы. Сумма элементарных матриц, соответствующих этим векторам, будет являться траекторной матрицей тренда ряда. После этого восстанавливают гармонические колебания ряда. Для отделения шума можно воспользоваться несколькими замечаниями: нерегулярное поведение сингулярных векторов может говорить о принадлежности их к набору, порожденному шумовой компонентой; также об этом может свидетельствовать медленное, практически без скачков, убывание собственных чисел с некоторого номера.

Диагональное усреднение. Если полученные сгруппированные матрицы ганкелевы, то они являются траекторными матрицами некоторого временного ряда, который может быть легко по ним восстановлен. Однако обычно сгруппированные матрицы редко получаются ганкелевыми, поэтому для восстановления временного ряда прибегают к диагональному усреднению. В соответствии с этим этапом каждый член восстановленного временного ряда будет являться средним арифметическим соответствующей ему побочной диагонали траекторной матрицы.

В результате проделанных шагов получается несколько временных рядов: один описывает тренд первоначального (исходного) временного ряда, другой – гармонические колебания, а третий – шумовые составляющие.

3 Программная реализация сингулярного спектрального анализа

Для исследования временных рядов методом ССА применяется программный модуль BelSim2#.SSA [2], который предназначен для исследовании числовых последовательностей данных и как специализированный инструмент для проведения исследований ССА [3, 4]. До разработки BelSim2#.SSA исследования проводились на основе комплекса информационных технологий, представленных табличным процессором MS Excel, математическим пакетом Mathcad и пакетом статистической обработки данных Statistica [5]. Такая методика является достаточно трудоемкой из-за значительного числа ручных операций.

Этап вложения. Для экспериментальных исследований исходный ряд $G = (g_0, g_1, ..., g_{N\text{-}1})$ может быть задан по известным функциям либо определен из входного файла. По известным функциям ряд формируется в программном коде *C#*:

```
Random R = new Random();
for (int i = 0; i < n * 2 - 1; i++)
{
    x[i] = i+1;
   G[i] = x[i] + A*Math.Sin(x[i]);
}
```

В главной форме на вкладке «Исходный ряд» отображается график сформированного ряда и затем в программном коде формируется матрица *A*, которая по правилам построения является ганкелевой. Процедура вложения является преобразованием исходного одномерного ряда $G = (g_0, g_1, ..., g_{N\text{-}1})$ в последовательность L - мерных векторов, число которых равно $K = N - L + 1$:

$$\mathbf{A}_i = (g_{i-1}, \ldots, g_{i+L-2})^T \text{ , } 1 \le i \le K \text{ .} \tag{1}$$

Блок кода *C#* формирования матрицы *A*:

```
for (int i = 0; i < n; i++)          // n - количество строк матрицы A
  {
    for (int j = 0; j < m; j++)      // m - количество столбцов матрицы A
     {
           A[i, j] = y[i + j];
      }
  }
```

Эти вектора образуют траекторную матрицу $A = [\mathbf{A}_1; \ldots; \mathbf{A}_K]$ ряда G, в которой $a_{ij} = g_{i+j-2}$, т. е. матрица A имеет одинаковые элементы на диагонали $i + j = const$.

Этап сингулярного разложения. Обозначим $S = A \cdot A^T \in R^{L \times L}$. Матрица $A \cdot A^T$ симметричная и неотрицательно определенная, а значит ее собственные числа $\{\mu_k\}_{k=1}^{L}$ вещественны и неотрицательны. Представленные в виде $\mu_1 \ge ... \ge \mu_L \ge 0$ собственные числа называют сингулярными значениями матрицы *S*. Пусть $\mathbf{U}_1, \ldots, \mathbf{U}_L$ – соответствующие им ортонормированные собственные вектора. Будем называть $p = \max\{k \mid \mu_k > 0\}$ порядком сингулярного разложения. Обозначим

$$\mathbf{V}_k = \frac{1}{\sqrt{\mu_k}} A^T \mathbf{U}_k, \; k = 1, ..., p. \tag{2}$$

Тогда сингулярным разложением матрицы *A* называется ее представление в виде суммы элементарных матриц

$$A = A_1 + A_2 + ... + A_p, \quad A_k = \sqrt{\mu_k}\, \mathbf{U}_k \mathbf{V}_k^T. \tag{3}$$

Для проведения сингулярного разложения использован алгебраический класс *alglib* [6]:

alglib.svd.rmatrixsvd(A, n, m, uneeded, vtneeded, additionalmem, ref d, ref u, ref vt);

Подпрограмма *rmatrixsvd* осуществляет *SVD*-разложение прямоугольной матрицы размером *m*x*n*. На выходе подпрограмма возвращает массив сингулярных значений, упорядоченных по убыванию, и, по требованию, матрицы *U* и *VT*, причем возможно как возвращение только левых и правых сингулярных векторов, так и полных матриц размером mxn и *n*x*n* (в зависимости от параметров *uneeded* и *vtneeded*). В переменной *VT* алгоритм возвращает не матрицу *V*, а транспонированную матрицу *VT*.

Каждая из матриц A_k имеет ранг, равный единице. Поэтому их можно назвать элементарными матрицами. Вектор $\mathbf{U}_k$ называют *k*-м левым сингулярным вектором или просто *k*-м собственным вектором, вектор $\mathbf{V}_k$ – правым сингулярным вектором. Набор $< \sqrt{\mu_k}, \mathbf{U}_k, \mathbf{V}_k >$ называют *k*-ой собственной тройкой.

Этап группировки. Вид левых и правых сингулярных векторов, трактуемых в ССА как временные ряды, является очень важным для следующего шага метода – группировки. При этом для одномерного ССА левые и правые сингулярные вектора обладают определенной симметрией, так как в этих случаях сингулярные разложения траекторных матриц с длиной окна L и $K = N - L + 1$ эквивалентны.

Процедура группировки формально одинакова для всех разновидностей ССА. На основе разложения (3) процедура группировки делит все множество индексов $\{1, \ldots, p\}$ на m непересекающихся подмножеств $I_1, ..., I_m$.

Пусть $I = \{i_1, \ldots, i_p\}$, тогда результирующая матрица A_I, соответствующая группе I, определяется как $A_I = A_{i_1} + \ldots + A_{i_p}$. Такие матрицы вычисляются для $I = I_1, ..., I_m$, тем самым разложение (3) может быть записано в сгруппированном виде:

$$A = A_{I_1} + \ldots + A_{I_m}. \quad (4)$$

Процедура выбора множеств $I = I_1, \ldots, I_m$ и называется группировкой собственных троек. Для определения $I = I_1, \ldots, I_m$ в BelSim2#.SSA используются сингулярные вектора и лепестковые диаграммы (вкладка «Лепестковые диаграммы», окно для просмотра и распознавания собственных троек, рисунок 1), которые являются аналогом графика в полярной системе координат, отображая распределение значений относительно начала координат.

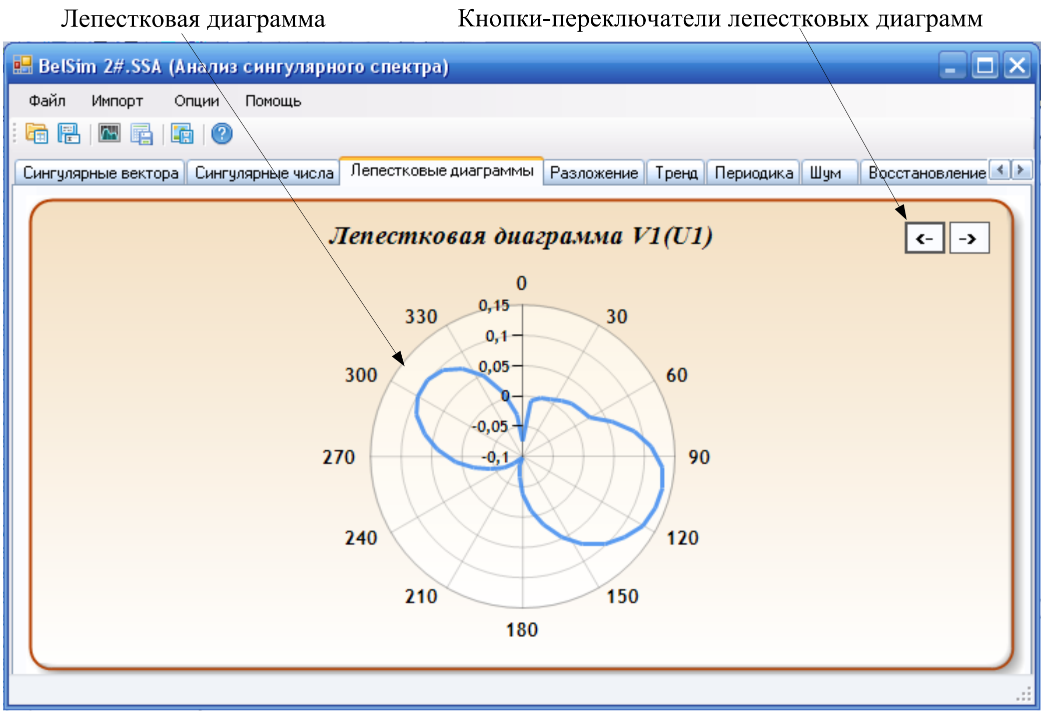

Рис. 1. Вкладка «Лепестковые диаграммы»

По особенностям представления сингулярных векторов на лепестковой диаграмме принимается решение о принадлежности их одной группе.

В программном коде для этого процесса написана специальная процедура:

double[,] razlozheniye(double[,] A_T, double[,] u, int n, int index, ref double[] T).

После группировки составляющих в окне «Сингулярные вектора» в зависимости от *index (0, 1, 2)* процедура возвращает тренд, гармонику и шум. Ниже представлен вызов этой процедуры для нахождения составляющих ряда:

X1 = razlozheniye(A_T, u, n, 0, reftrend); //выделение тренда
X2 = razlozheniye(A_T, u, n, 1, refsinus); //выделение гармоники
X3 = razlozheniye(A_T, u, n, 2, refnoise); //выделение шума

Этап диагонального усреднения. На последнем шаге базового алгоритма каждая матрица сгруппированного разложения переводится в новый ряд длины N. Для произвольной матрицы X процедуру приведения ее к ганкелевому виду и последующему преобразованию в ряд (обозначим его как $G^{\text{в}}$) выразим следующим образом. Пусть X – матрица размера $L \times K$ с элементами x_{ij}, $1 \le i \le L$, $1 \le j \le K$. Положим $L^* = \min(L, K)$, $K^* = \max(L, K)$ и $N = L + K - 1$. Пусть $z_{ij} = x_{ij}$, если $L < K$ и $z_{ij} = x_{ji}$ в остальных случаях. Тогда диагональное усреднение переводит матрицу X в ряд $(g_0^{\text{в}}, \ldots, g_{N-1}^{\text{в}})$ по формуле

$$g_k^{\text{в}} = \begin{cases} \frac{1}{k+1} \sum\limits_{j=1}^{k+1} z_{j,\, k-j+2} \mid 0 \le k \le L^* - 1; \\ \frac{1}{L^*} \sum\limits_{j=1}^{L^*} z_{j,\, k-j+2} \mid L^* - 1 \le k \le K^*; \\ \frac{1}{N-k} \sum\limits_{j=k-K^*+2}^{N-K^*+1} z_{j,\, k-j+2} \mid K^* \le k \le N. \end{cases} \qquad (5)$$

Выражение (5) соответствует усреднению элементов матрицы вдоль побочных диагоналей $i + j = k + 2$: выбор $k = 0$ дает $g_0^{\text{в}} = x_{11}$, для $k = 1$ получаем $g_1^{\text{в}} = (x_{12} + x_{21})/2$ и т. д. Применив диагональное усреднение к матрицам, полученным на этапе группировки, приходим к разложению исходного ряда в сумму m рядов.

4. Заключение

Программный модуль разработан на основе программной платформы *.NET Framework* 4.0 в среде программирования *Microsoft Visual Studio* 2008 с применением объектно-ориентированного языка программирования *C#*.

Файл с исходным временным рядом представляет собой текстовый файл, содержащий последовательность вещественных чисел. Выходной файл, создаваемый программой, представляет собой текстовый файл и графические файлы. Выходной текстовый файл хранит в себе следующую информацию: исходный временной ряд; численные значения трендовой, периодической и шумовой составляющей. Выходные графические файлы содержат графики исходного ряда; сингулярных векторов; сингулярных чисел; лепестковые диаграммы левых и правых сингулярных векторов; трендовой, периодической и шумовой составляющей.

Технические требования BelSim 2#.SSA: процессор Pentium 4x2 ГГц и выше; оперативная память 512 Мбайт и больше; пространство на жестком диске 1 Мбайт; операционная система Windows 2000/XP/Vista/7; дополнительное программное обеспечение: набор библиотек .NET Framework 4.0 Client Profile (41 Мбайт) [7].

Литература

1. **Golyandina, N.** Analysis of Time Series Structure: SSA and Related Techniques / N. Golyandina, V. Nekrutkin, A. Zhigljavsky. – Boca Raton: Chapman & Hall/CRC, 2001. – 310 p.
2. **Якимов, Е. А.** Программный модуль анализа сингулярного спектра числовых последовательностей данных «BelSim2#.SSA» : свидетельство о регистрации компьютерной программы № 518 / Е. А. Якимов, Д. М. Албкеират, Н. М. Чапаров, А. И. Якимов. – Минск: НЦИС, 2013. – Заявка № С20130042. – Дата подачи: 23.05.2013.
3. **Чапаров, Н. М.** Исследование SSA-метода при восстановлении трендовой составляющей временного ряда / Н. М. Чапаров, Я. А. Процкая; науч. рук.: А. И. Якимов // 48-я студенческая научно-техническая конференция Белорусско-Российского университета: материалы конф., редкол.: И. С. Сазонов (гл. ред.) [и др.]; 23-24 мая 2012 г. – Могилев: Белорус.-Рос. ун-т, 2012. – С. 214.
4. **Якимов, Е. А.** О краевом эффекте при сингулярном спектральном анализе числовых последовательностей данных / Е. А. Якимов, Н. М. Чапаров, Д. М. Албкеират // Математическое и имитационное моделирование систем. МОДС 2012 : тез. докл. седьмой междунар. науч.-практич. конф., 25–28 июня 2012 г. – Чернигов-Жукин: ИПММС НАН Украины, 2012. – С. 384–386.
5. **Якимов, Е. А.** Исследование SSA-метода на основе комплексного применения информационных технологий / Е. А. Якимов // Доклады БГУИР. – 2010. – № 2(48). – С. 77–83.
6. **Бочканов, С.** Библиотека алгоритмов / С. Бочканов : [Электрон. ресурс]. Numerical Analysis Library, 2012. // Режим доступа : http://www.alglib.net/ – Дата доступа 23.11.2012.
7. **Чапаров, Н. М.** Разработка программного обеспечения для исследования временных рядов методом сингулярного спектрального анализа / Н. М. Чапаров; науч. рук.: А. И. Якимов, Е. А. Якимов // 49-я студенческая научно-техническая конференция Белорусско-Российского университета: материалы конф., редкол.: И. С. Сазонов (гл. ред.) [и др.]; 16-17 мая 2013 г. – Могилев: Белорус.-Рос. ун-т, 2013. – С. 224.
8.

Чапаров Назар Мурадович
Выпускник кафедры Автоматизированные системы управления
Белорусско-Российский университет, г. Могилев
Тел.: +375(222) 25-24-47
E-mail: scof_91@mail.ru
Якимов Евгений Анатольевич
Ассистент кафедры Программное обеспечение информационных технологий
Белорусско-Российский университет, г. Могилев
Тел.: +375 (29) 312-41-94
E-mail: e-soft@bk.ru
Якимов Анатолий Иванович
Доцент кафедры Автоматизированные системы управления
Белорусско-Российский университет, г. Могилев
Тел.: +375(222) 25-24-47
E-mail: ykm@tut.by