Modified Classification Method of Multivariate Time Series Based on Shapelets

Article *in* Herald of the Bauman Moscow State Technical University Series Instrument Engineering · April 2017

DOI: 10.18698/0236-3933-2017-2-46-65

2 authors:

Peter Sotnikov
Bauman Moscow State Technical University
7 PUBLICATIONS **18** CITATIONS
SEE PROFILE

Anatoly Pavlovich Karpenko
Bauman Moscow State Technical University
127 PUBLICATIONS **428** CITATIONS
SEE PROFILE

УДК 519.6 DOI: 10.18698/0236-3933-2017-2-46-65

МОДИФИЦИРОВАННЫЙ МЕТОД КЛАССИФИКАЦИИ МНОГОМЕРНЫХ ВРЕМЕННЫХ РЯДОВ С ИСПОЛЬЗОВАНИЕМ ШЕЙПЛЕТОВ

А.П. Карпенко[1] apkarpenko@bmstu.ru
П.И. Сотников[1, 2] sotnikoffp@gmail.com

[1] МГТУ им. Н.Э. Баумана, Москва, Российская Федерация
[2] ЗАО «Информтехника и Связь», Москва, Российская Федерация

Аннотация

Рассмотрена классификация многомерных временных рядов с помощью метода шейплетов. Вместо полного перебора фрагментов исходных временных рядов для поиска шейплетов предложено использовать генетический алгоритм. Выполнена оценка качества шейплетов путем определения точности классификации, достижимой на множестве векторов расстояний от кандидата до исходных временных рядов. Эффективность предложенных модификаций метода шейплетов исследована путем анализа известных электроэнцефалограмм, полученных при работе пользователей с интерфейсом мозг–компьютер на основе волны P300. Результаты исследования показали, что применение указанных модификаций позволяет сократить почти на 99 % множество перебора при поиске шейплетов без потери точности классификации

Ключевые слова

Многомерный временной ряд, классификация, шейплет, генетический алгоритм, интерфейс мозг–компьютер

Поступила в редакцию 16.06.2016

Введение. Во многих областях науки и техники возникают задачи классификации многомерных временных рядов. Например, такие задачи возникают в процессе машинного распознавания речи, при анализе сейсмологических и метеорологических данных, биомедицинских данных, в том числе сигналов электроэнцефалограмм в нейрокомпьютерном интерфейсе. Одним из сравнительно новых методов классификации временных рядов является метод шейплетов (*англ. shapelets*), основанный на выделении таких фрагментов временного ряда, которые наилучшим образом отражают свойства одного или нескольких классов исследуемых временных рядов. К преимуществам шейплетов относят их способность «подмечать» локальные различия временных рядов и возможность удобной визуализации выявленных различий.

В исходном варианте метод шейплетов предложен в целях разделения одномерных временных рядов на два класса [1]. Для каждого отрезка временного ряда (кандидата в шейплеты) выполняют оценку его качества путем определения так называемого информационного дохода (*англ. Information gain*). В качестве шейплета выбирают кандидата с лучшим значением этой оценки. Решение о принадлежности временного ряда к одному из классов принимают после вы-

числения «расстояния» от найденного шейплета до рассматриваемого ряда. Если оно не превышает некоторого порогового значения, то ряд относят к первому классу, в противном случае — ко второму. В работе [1] также рассмотрено расширение метода шейплетов на случай трех и более классов. Для решения такой задачи предложено использовать дерево решений, в котором каждый узел дерева представляет собой бинарный шейплет–классификатор.

В работе [2] предложен метод шейплет-преобразования (*англ. Shapelet transform*), который является дальнейшим развитием исходного варианта метода шейплетов. Идея метода состоит в том, что в качестве характерных признаков временного ряда используют расстояния от этого ряда до набора из k лучших шейплетов. Преимуществом данного подхода является возможность применения к указанным признакам различных методов классификации, не ограничиваясь деревом решений. В той же работе предложено оценивать качество шейплетов с помощью критерия Фишера (*англ. Fisher'sratio*), показывающего отношение межклассовой дисперсии к внутриклассовой.

Существенным недостатком, ограничивающим применение метода шейплетов, является его высокая вычислительная сложность, связанная с необходимостью в предельном случае полного перебора всех фрагментов исходных временных рядов. В публикациях [1, 3] предложены способы сокращения множества перебираемых вариантов путем отбрасывания кандидатов после предварительной оценки их качества или оценки их близости к уже рассмотренным фрагментам временных рядов.

В настоящей работе для поиска шейплетов вместо полного перебора фрагментов исходных временных рядов предложено использовать генетический алгоритм. Задачу поиска шейплетов рассматриваем как задачу оптимизации, в которой роль целевой функции играет оценка качества шейплета. Варьируемыми выступают такие параметры шейплета, как индекс исходного временного ряда, фрагментом которого является шейплет, сдвиг относительно начала временного ряда и длина шейплета. Также в работе предложен новый способ оценки качества шейплетов, который заключается в определении точности классификации, достижимой на векторах расстояний от шейплета до исходных временных рядов. Эффективность предложенных модификаций метода шейплетов исследована при анализе электроэнцефалограмм (ЭЭГ), которые получены при работе пользователей с интерфейсом мозг–компьютер (ИМК) [4]. Выбор этих данных обусловлен тем, что возможность применения метода шейплетов для анализа данных в ИМК является в настоящее время малоизученной. Таким образом, результаты исследования представляют самостоятельную ценность.

Постановка задачи классификации многомерных временных рядов с помощью шейплетов. *Временной ряд* — это упорядоченная последовательность отсчетов $\mathbf{X} = \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, взятых в моменты времени $t = (0, 1, ..., (N-1))\,\Delta t$, где N — общее число отсчетов; Δt — интервал дискретизации. Многомерный временной ряд, содержащий совокупность дискретных отсчетов, полученных по

нескольким измерительным каналам, обозначаем как $\mathbf{X} \in \mathbb{R}^{M \times N}$, где M — общее число каналов.

Множество $\Omega = \{\mathbf{X}_i, k_i\}_{i=1}^{P}$, где $\mathbf{X}_i$ — i-й многомерный временной ряд; $k_i \in \{1, 2, \ldots, K\}$ — метка класса, соответствующая данному ряду, примем за исходные данные. Фрагментом S одномерного временного ряда называем набор последовательных отсчетов этого ряда. Фрагмент S ряда X, имеющий длину l и начинающийся с позиции j, записываем как $S = x_j, x_{j+1}, \ldots, x_{j+l-1}$. Общее число таких фрагментов ряда X равно, очевидно, $N - l + 1$. Фрагмент многомерного ряда $\mathbf{X}$ обозначаем как

$$\mathbf{S} = \begin{pmatrix} x_{1,j}; x_{1,j+1}; \ldots; x_{1,j+l-1} \\ x_{2,j}; x_{2,j+1}; \ldots; x_{2,j+l-1} \\ \vdots \\ x_{M,j}; x_{M,j+1}; \ldots; x_{M,j+l-1} \end{pmatrix}.$$

Если S_1, S_2 — фрагменты временного ряда X, имеющие длины l, то расстояние между этими фрагментами определяется евклидовой метрикой

$$d(S_1, S_2) = \sqrt[2]{\sum_{j=1}^{l} (x_{1,j} - x_{2,j})^2}.$$

Для устранения влияния сдвига и масштабирования данных на конечный результат перед вычислением расстояния фрагменты S_1, S_2 временного ряда должны быть нормализованы.

Расстоянием между временным рядом X и фрагментом S длины l называем минимальное расстояние между этим фрагментом и всеми возможными фрагментами этого ряда длины l:

$$d(S, X) = \min_{S_X \in D_S} d(S, S_X),$$

$$D_S = \{x_j, x_{j+1}, \ldots, x_{j+l-1}\}_{j=1}^{N-l+1}.$$

Степень отличия ряда $\mathbf{X}$ от фрагмента $\mathbf{S} \in \mathbb{R}^{M \times l}$ характеризуем с помощью M-мерного вектора расстояний

$$D(\mathbf{S}, \mathbf{X}) = \begin{pmatrix} d(S_1, X_1) \\ d(S_2, X_2) \\ \vdots \\ d(S_M, X_M) \end{pmatrix}, \qquad (1)$$

компоненты которого есть расстояния между соответствующими измерениями фрагмента и временного ряда. Множеству данных Ω ставим в соответствие совокупность векторов расстояний $\Psi(\mathbf{S}) = \{D(\mathbf{S}, \mathbf{X}_i),\ k_i\}_{i=1}^{P}$.

Пусть $\varphi(\mathbf{S})$ — скалярная функция, являющаяся некоторой оценкой качества разделения классов на множестве $\Psi(\mathbf{S})$. Полагаем, что если фрагмент $\mathbf{S}_i$ временного ряда позволяет получить лучшее разделение классов, чем фрагмент $\mathbf{S}_j$, то справедливо соотношение $\varphi(\mathbf{S}_i) > \varphi(\mathbf{S}_j)$.

Шейплетом $\mathbf{S}_{\text{opt}}$ набора данных Ω называется такой фрагмент многомерного временного ряда, для которого оценка качества разделения классов принимает наибольшее значение:

$$\varphi(\mathbf{S}_{\text{opt}}) = \max_{\mathbf{S} \in D_S} \varphi(\mathbf{S}). \tag{2}$$

Определим фрагмент временного ряда тройкой чисел (i, j, l), где i — номер временного ряда во множестве Ω; j — смещение фрагмента относительно начала ряда; l — длина фрагмента; $i, j, l \in \mathbb{Z}$. Кандидата в шейплеты, определенного с использованием тройки (i, j, l), обозначим как $\mathbf{S}(i, j, l)$. Задачу поиска шейплетов (2) формулируем в виде

$$\max_{i,j,l \in D} \varphi(\mathbf{S}(i, j, l)) = \varphi(\mathbf{S}(i^*,\ j^*,\ l^*)) = \varphi(\mathbf{S}_{\text{opt}}); \tag{3}$$

$$D = \begin{cases} 1 \le i \le P; \\ 1 \le j \le N - l + 1; \\ l_{\min} \le l \le l_{\max}, \end{cases}$$

где $l_{\min}, l_{\max}$ — константы, определяющие границы диапазона длин кандидатов.

Базовый вариант метода шейплетов и его улучшения. Идея метода шейплет-преобразования (метода шейплетов) заключается в формировании вектора характерных признаков (ВХП) временного ряда путем расчета расстояний от этого ряда до набора из нескольких шейплетов. В отличие от оригинального метода поиска шейплетов [1] такой подход позволяет перейти от использования дерева решений для классификации временных рядов к любым другим методам классификации.

Обозначим $\mathbb{S}$ совокупность из k шейплетов $\mathbf{S}_1, \ldots, \mathbf{S}_k$. Тогда ВХП многомерного временного ряда $\mathbf{X}$, полученный методом шейплет-преобразования, можно записать в виде

$$V(\mathbb{S}) = \left(D^{\text{т}}(\mathbf{S}_1, \mathbf{X}), \ldots, D^{\text{т}}(\mathbf{S}_k, \mathbf{X})\right)^{\text{т}}, V(\mathbb{S}) \in \mathbb{R}^{M \times k},$$

где $D(\mathbf{S}_i, \mathbf{X})$, $i = 1, \ldots, k$ — векторы расстояний (1).

В исходном варианте метода шейплет-преобразования в набор $\mathbb{S}$ включают k лучших кандидатов, найденных путем полного перебора всех фрагментов

временных рядов из исходного набора данных [2], т. е. для поиска кандидатов используют так называемый алгоритм «грубой силы» (*англ. Brute-Force Algorithm*). Псевдокод данного алгоритма представлен далее.

$\mathbb{S} \leftarrow \varnothing$

$l \leftarrow l_{\min}$

While $l \le l_{\max}$

$shapelets \leftarrow \varnothing$

$candidates \leftarrow GenerateCandidates(\Omega, l)$

for each S **in** *candidates*

$quality \leftarrow CheckCandidate(\Omega,\ \mathrm{S})$

$shapelets.add(S, quality)$

end for

$l \leftarrow l+1$

$sortByQuality(shapelets)$

$\mathbb{S} \leftarrow merge(k, \mathbb{S}, shapelets)$

end while

return $\mathbb{S}$

Входными данными алгоритма являются множество Ω и границы $l_{\min}, l_{\max}$. Процедура *GenerateCandidates* генерирует все возможные фрагменты длины l из множества Ω и сохраняет их в неупорядоченном списке кандидатов *candidates*. В список *shapelets* включаются все сгенерированные кандидаты вместе с полученными для них оценками качества разделения классов *quality*. Далее с помощью процедуры *sortByQuality* производится сортировка элементов списка в порядке убывания значений для оценки качества разделения классов. Функция *merge* выполняет слияние списков $\mathbb{S}$ и *shapelets* таким образом, что при этом выбираются только k лучших кандидатов.

В представленном алгоритме число рассматриваемых кандидатов из одного временного ряда длины N равно $O(N^2)$, а общее число кандидатов для всего набора данных Ω объема P равно $O(PN^2)$. Поскольку время вычисления расстояния от кандидата до любого временного ряда из набора Ω можно оценить величиной $O(PN^2)$, общая сложность алгоритма составляет $O(P^2N^4)$, т. е. очень быстро растет с ростом величин P, N.

Известен ряд подходов, обеспечивающих сокращение времени поиска шейплетов:

1) кэширование вычислений при расчете расстояний от кандидата до временных рядов [1];

2) отбрасывание кандидатов на основе предварительной оценки их качества [1];

3) отбрасывание кандидатов после оценки их близости к уже рассмотренным фрагментам временных рядов [3];

4) применение специального символьного алфавита для кодирования временных рядов [5].

Используем алгоритм, реализующий третий из указанных подходов. В работе [3] показано, что данный подход позволяет значительно сократить перебор без существенной потери точности разделения классов. Идея метода основывается на предположении, что близкие друг к другу кандидаты дают близкие значения оценки качества разделения классов.

Введем в рассмотрение порог ϵ, имеющий смысл перцентили распределения расстояний между случайно выбранными парами кандидатов. Значение p-й перцентили указывает, что p % значений, представленных в распределении, лежат ниже этого уровня. Таким образом, вводя порог ϵ, мы можем исключить из рассмотрения до p % кандидатов, лежащих в окрестности рассматриваемого фрагмента временного ряда.

Алгоритм определения порога для отбрасывания кандидатов (*ComputeThresholdAlgorithm)* имеет следующий вид:

$Z \leftarrow \varnothing$

$$q \leftarrow 1$$

while $q \le Q$

$$i \leftarrow random(P)$$
$$j \leftarrow random(N - l)$$
$$S \leftarrow getCandidate(i, j, l)$$
$$i' \leftarrow random(P)$$
$$j' \leftarrow random(N - l)$$
$$S' \leftarrow getCandidate(i', j', l)$$
$$Z \leftarrow Z \cup dist(S, S')$$
$$q \leftarrow q + 1$$

end while

$$Z \leftarrow sort(Z)$$
$$\epsilon \leftarrow Z_{\frac{p}{100}Q}$$

return ϵ

На итерациях алгоритма между случайно выбранными фрагментами S, S' вычисляем расстояние $dist(S, S')$, и это расстояние добавляем в список Z. По результатам работы алгоритма порог ϵ определяем как значение p-й перцентили в отсортированном по возрастанию списке расстояний Z.

Для фрагментов $\mathbf{S}, \mathbf{S}'$ многомерных временных рядов алгоритм *ComputeThreshold* использует скалярную свертку:

$$dist\left(\mathbf{S},\mathbf{S}'\right)=\max_{i=[1,\ldots,M]}d\left(S_i,S_i'\right), \tag{4}$$

где $d\left(S_i,S_i'\right)$ — расстояния между соответствующими измерениями фрагментов $\mathbf{S}$, $\mathbf{S}'$.

С учетом алгоритма *ComputeThreshold* процедура *GenerateCandidates* базового алгоритма *Brute-Force* приобретает следующий вид:

$pool \leftarrow \varnothing$

$p \leftarrow const$

$Q \leftarrow const$

$\epsilon \leftarrow ComputeThreshold\left(\Omega, l, p, Q\right)$

$i \leftarrow 1$

while $i \le P$

$j \leftarrow 1$

while $j < N - l$

$S \leftarrow getCandidate\left(i, j, l\right)$

$D \leftarrow \varnothing$

for each S' **in** $pool$

$D \leftarrow D \cup dist\left(S, S'\right)$

end for

if $\forall d \in D \,|\, d \rangle \epsilon$

$pool \leftarrow pool \cup S$

end if

$j \leftarrow j + 1$

end while

$i \leftarrow i + 1$

end while

return $pool$

В процессе формирования списка кандидатов *pool* из исходного набора данных Ω выделяем все возможные фрагменты длины l. Для каждого такого фрагмента S оцениваем расстояние от этого фрагмента до всех кандидатов, уже вошедших в список *pool*. Если рассматриваемый фрагмент S удален от всех кандидатов на расстояние, превышающее значение порога ϵ, то включаем его в список *pool*. В противном случае, фрагмент S отбрасываем.

В методе шейплет-преобразования выбор числа шейплетов представляет собой самостоятельную задачу. Использование малого числа шейплетов в наборе может не дать достаточной информации для точного разделения классов, а применение большого числа шейплетов может привести к переобучению алгоритма классификации и ослабить влияние важных признаков рассматриваемых временных рядов на результат классификации. В оригинальном методе

шейплет-преобразования в набор включают $N/2$ лучших кандидатов, полученных за один проход алгоритма поиска шейплетов. Поскольку похожие друг на друга шейплеты, как правило, отражают свойства одного и того же класса, включение в набор нескольких похожих шейплетов не увеличивает информативность ВХП. Поэтому рекомендуется оставлять в наборе только шейплеты, значимо отличающиеся друг от друга. Для этих целей можно использовать алгоритм иерархической кластеризации [6].

Предлагаемые модификации метода шейплетов. *Использование генетического алгоритма*. Задача поиска шейплетов (3) представляет собой задачу целочисленной оптимизации. В работе [7] рассмотрена аналогичная постановка задачи и для её решения предложено использовать известный метод градиентного спуска (после сведения задачи к непрерывной). Поскольку нет никаких оснований полагать, что задача (3) является одноэкстремальной, этот метод может отыскать лишь ее локальное решение. Генетический же алгоритм может локализовать глобальный экстремум задачи (3) и тем самым повысить точность разделения многомерных временных рядов на классы.

Особей *s* популяции генетического алгоритма определят набор

$$s = F,\ H,\ \varphi(F),$$

где вектор F — фенотип особи; H — генотип особи; $\varphi(F)$ — приспособленность. Фенотип особи задаем тройкой чисел (i, j, l), определяющих фрагмент временного ряда, т. е. полагаем $F=(i,\ j,\ l)$. Для кодирования значений варьируемых параметров (i, j, l) используем представление генотипа в виде монохромосомы $H=(h_i,\ h_j,\ h_l)$, где h_i, h_j, h_l — бинарные гены, отвечающие за кодирование чисел i, j, l соотвественно.

Для каждой из особей *s* значение функции приспособленности $\varphi(F)$ вычисляем по следующей схеме.

1. На основе индексов i, j, l выбираем из множества Ω кандидата в шейплеты $\mathbf{S}=\mathbf{S}(i,\ j,\ l)$.

2. Рассчитываем расстояния от кандидата $\mathbf{S}$ до всех объектов из множества Ω. Формируем новое множество $\Psi(\mathbf{S})=\{D(\mathbf{S},\mathbf{X}_i),k_i\}_{i=1}^{P}$.

3. На множестве $\Psi(\mathbf{S})$ выполняем оценку точности разделения классов. Полученную оценку принимаем в качестве значения функции приспособленности $\varphi(F)$.

В генетическом алгоритме отбираются родительские пары по методу «рулетки». С равной вероятностью используем одноточечный или двухточечный операторы скрещивания (оператор кроссинговера) и для каждой пары родителей формируем два потомка. В качестве оператора мутации применяем оператор равномерной генной мутации. Методом смертельных штрафов [8] учитываем ограничения на значения параметров (i, j, l). По общему принципу эволюционных

алгоритмов в качестве решения задачи используем значения варьируемых параметров $\left(i^*, j^*, l^*\right)$, соответствующие лучшей особи (имеющей наибольшее значение функции приспособленности φ^*). Из последнего поколения генетического алгоритма выбираем k лучших шейплетов.

Для того чтобы повысить вероятность локализации глобального решения, используем метод мультистарта [8].

Новая оценка качества разделения классов. В исходном варианте метода шейплетов качество кандидатов оценивают с помощью информационного дохода. Позже в ряде публикаций для этой цели было предложено использовать статистические критерии, позволяющие судить о степени разделения классов: критерий Фишера, критерий Краскела — Уоллиса (*англ. Kruskal — Wallistest*), медианный критерий и др. [9].

Мы предлагаем в качестве меры $\varphi(\mathbf{S})$ качества разделения классов использовать точность классификации, достижимую на векторах расстояний от кандидата до объектов исходного набора данных. Точность классификации предлагаем оценивать с помощью простого классификатора на основе метода k ближайших соседей. Для сокращения объема вычислений используем модификацию этого метода, в которой решение о принадлежности объекта классу принимается после вычисления расстояний от этого объекта до центров классов C (вместо расчета расстояний до всех соседей).

Схема предлагаемого алгоритма для определения качества разделения классов имеет следующий вид.

1. Формируем множество $\Psi(\mathbf{S}) = \left\{D\left(\mathbf{S}, \mathbf{X}_i\right), k_i\right\}_{i=1}^{P}$.

2. Разбиваем множество $\Psi(\mathbf{S})$ на пять групп одинакового размера для организации перекрестной проверки (*англ. cross-validation*) качества разделения.

3. На раундах (*англ. folds*) перекрестной проверки четыре группы используем для обучения классификатора, а одну группу — для его тестирования.

4. По результатам пяти раундов вычисляем среднюю точность классификации, которую и принимаем в качестве значения функции $\varphi(\mathbf{S})$ для данного кандидата.

Введение процедуры кросс-проверки необходимо для получения несмещенной оценки точности разделения классов.

Программная реализация и вычислительный эксперимент. *Программная реализация.* Программная реализация метода шейплетов выполнена в среде графического программирования NILabVIEW 2012. Особенностью среды разработки является то, что функции и операторы в ней представляют в виде виртуальных приборов (инструментов), а программный код — в виде блок-диаграммы, на которой отображаются связи между виртуальными приборами.

При реализации вычислительных алгоритмов использованы следующие библиотечные функции:

– функции библиотеки *XMLParser* для чтения входных данных;

– инструмент *IMAQTrainNearestNeighborVI* из библиотеки *ClassifierEngines* для построения классификатора на основе *k* ближайших соседей;

– инструмент *IMAQCrossValidationVI* из библиотеки *Classification* для организации процедуры кросс-проверки;

– инструмент *IMAQTrainSVMVI* из библиотеки *ClassifierEngines* для построения классификатора на основе метода опорных векторов;

– функции библиотек *QueueOperations*, *NotifierOperations* для организации взаимодействия между потоками.

Разработанная утилита «*MultichannelShapeletsSearch.vi*» реализует поиск шейплетов с помощью алгоритма полного перебора и алгоритма поиска с отбрасыванием кандидатов. Утилита «*OptimalShapeletsSearch.vi*» реализует поиск шейплетов с помощью генетического алгоритма. Приложения доступны на вебсайте http://ru-bci.org/ в разделе «Загрузки».

Тестовые данные. В качестве тестовых данных использованы записи ЭЭГ, полученные исследовательской группой Кристофа Гугера (*ChristophGuger*) компании *g.tecMedicalEngineeringGmbH*. Запись ЭЭГ проводилась для 100 испытуемых при работе с интерфейсом мозг–компьютер на основе волны P300. На экране отображалась матрица из 36 символов (*A*, *B*, …, *Z*; 0, 1, …, 9). В первом варианте эксперимента на короткий интервал времени (100 мс) поочередно подсвечивались строки и столбцы этой матрицы. Во втором варианте подсвечивался отдельно каждый символ. В обоих вариантах с помощью анализа ЭЭГ и выделения реакции в виде волны P300 определялся тот символ, к которому привлечено внимание испытуемого. Каждый испытуемый принял участие в двух сессиях. Во время обучающей сессии испытуемым было предложено набрать слово WATER, фокусируя внимание поочередно на каждой букве, а во время тестовой сессии — слово LUCAS. Запись ЭЭГ велась с помощью восьми электродов. Частота дискретизации составляла 256 Гц, разрешающая способность — 24 бита.

В работе использованы данные ЭЭГ, зарегистрированные для восьми испытуемых (*s*3–*s*10), участвовавших в первом эксперименте (http://bnci-horizon-2020.eu/, раздел *Database*, набор данных *VisualP300speller* (003-2015)). Для всех испытуемых *s*3–*s*10 полученные во время обучающей и тестовой сессий записи ЭЭГ содержат по 900 эпох, из которых 750 эпох соответствуют незначимым стимулам (подсветка строк и столбцов матрицы, которые не содержат набираемого символа), а 150 эпох — значимым стимулам.

Рассмотрим задачу бинарной классификации указанных записей ЭЭГ, цель которой — определить, содержит ли данная запись реакцию в виде волны P300 на предъявляемый стимул или нет.

Предобработка данных и параметры используемых алгоритмов классификации. На первом шаге предвыборки сигнал ЭЭГ разбиваем на эпохи (отрезки) продолжительностью 800 мс, начиная с момента предъявления стимула (подсветки столбца или матрицы виртуальной клавиатуры). На втором шаге выпол-

няем фильтрацию сигнала в диапазоне частот 0,2…30 Гц и понижение частоты выборок с 256 Гц до 64 Гц.

Для восьми испытуемых *s*3–*s*10 поиск шейплетов многомерного сигнала ЭЭГ осуществляем среди записей обучающей сессии. Записи ЭЭГ, полученные во время тестовой сессии, используем для оценки точности классификации.

Для алгоритма полного перебора (алгоритма A_1) границы, определяющие диапазон длин кандидатов, принимаем равными $l_{\min} = 0{,}2\,N$, $l_{\max} = N$, где $N \approx 5$ — длина отрезка сигнала ЭЭГ. Для алгоритма поиска с отбрасыванием кандидатов (алгоритма A_2) используем те же границы диапазона кандидатов. Значение перцентили p принимаем 10 %. Для генетического алгоритма (алгоритма A_3) используем следующие значения свободных параметров: кодирование генов 10 разрядными двоичными числами (код Грея); размер популяции — 1000 особей; число поколений — 10; вероятность мутации — 0,1. Для испытуемых *s*3–*s*10 было выполнено по пять запусков генетического алгоритма и по пять запусков алгоритма поиска с отбрасыванием кандидатов (метод мультистарта).

Таблица 1

Значения оценок качества шейплетов

Испытуемый	Алгоритм		
	A_1	A_2	A_3
*s*3	0,64	0,63	0,63
s4	0,73	0,68	0,72
*s*5	0,71	0,68	0,70
s6	0,62	0,60	0,62
*s*7	0,64	0,63	0,63
*s*8	0,66	0,64	0,64
*s*9	0,65	0,63	0,63
*s*10	0,70	0,68	0,68

Сравнение эффективности алгоритмов классификации. В табл. 1 представлены значения оценок качества разделения классов $\varphi(S_{\text{opt}})$ для шейплетов, найденных с помощью алгоритмов A_1, A_2, A_3. Для алгоритмов A_2, A_3 приведены средние арифметические значения, рассчитанные по результатам мультистарта (рис. 1).

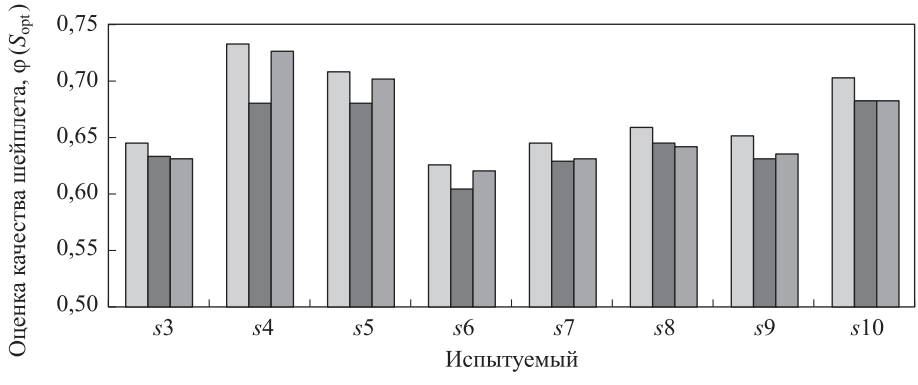


Рис. 1. Оценки качества шейплетов, найденных различными алгоритмами поиска (— алгоритм A_1, — алгоритм A_2, — алгоритм A_3)

В табл. 2 для алгоритма A_2 представлено, сколько в среднем (по результатам мультистарта) кандидатов было оставлено в рассмотрении.

Таблица 2

Среднее число кандидатов в шейплеты, оставленных после отбрасывания

Испытуемый	*s*3	*s*4	*s*5	*s*6	*s*7	*s*8	*s*9	*s*10
Число кандидатов	7 682	4 472	3 737	4 682	4 986	4 512	4 380	4 670

Видно, что общее число всех возможных фрагментов временных рядов длиной $l \in \left[l_{\min}; l_{\max} \right]$ составляет

$$N_S = P \sum_{l=l_{\min}}^{l_{\max}} (N - l + 1).$$

С учетом того, что каждый отрезок ЭЭГ содержит 51 отсчет, а поиск шейплетов проводится среди 900 отрезков, полученных во время обучающей сессии, общее число фрагментов временных рядов, рассматриваемых в алгоритме A_1, равно 812 700. Таким образом, из табл. 2 следует, что алгоритм A_2 позволяет исключить из рассмотрения в среднем до 99,4 % кандидатов, а алгоритм A_3 с размером популяции 1000 особей и числом поколений, равным 10 — почти 99 % кандидатов.

Для того чтобы определить, какой из алгоритмов A_2, A_3 дает лучшее приближение к глобальному оптимуму, по результатам мультистарта этих алгоритмов вычислены величины $\Delta\overline{\varphi}(\mathbf{S}_{\text{opt}})$ — средние отклонения оценок качества разделения классов от их оптимальных значений (табл. 3, рис. 2). В качестве последних приняты значения указанных оценок, полученные с помощью алгоритма полного перебора A_1 (см. табл. 2).

Таблица 3

Средние отклонения оценок качества разделения классов $\Delta\overline{\varphi}(\mathbf{S}_{\text{opt}})$

Испытуемый	Алгоритм	
	A_2	A_3
*s*3	0,013	0,014
*s*4	0,052	0,007
*s*5	0,029	0,007
*s*6	0,020	0,004
*s*7	0,015	0,013
*s*8	0,015	0,017
*s*9	0,022	0,018
*s*10	0,021	0,021
Общее среднее	0,023	0,013

Из табл. 3 и рис. 2 следует, что генетический алгоритм A_3 в среднем обеспечивает лучшее приближение к оптимальному значению. Для проверки этой

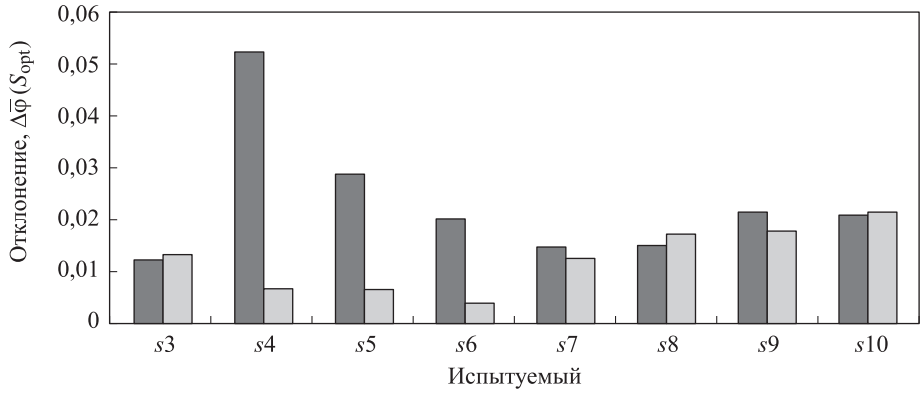


Рис. 2. Отклонения $\Delta\bar{\varphi}(\mathbf{S}_{opt})$ оценок качества разделения классов от их оптимальных значений (— алгоритм A_1, — алгоритм A_3)

гипотезы выполнена оценка статистической значимости полученных результатов с помощью однофакторного дисперсионного анализа. Уровень значимости, при котором отвергается нулевая гипотеза о равенстве средних, принят равным $\alpha = 0,05$. Рассчитанное значение критерия Фишера составляет $F_{эмп} = 4,704$ и превышает критическое значение $F_{крит} = 4,6$, определенное для уровня значимости $\alpha = 0,05$. Следовательно, предположение о том, что генетический алгоритм имеет большую эффективность, является верным.

Оценка обобщающих свойств найденных шейплетов. Обобщающие свойства шейплетов, найденных с помощью алгоритмов A_1, A_2, A_3, оценены на записях ЭЭГ, полученных во время тестовой сессии. Для каждого из алгоритмов в итоговый набор шейплетов было включено 32 лучших кандидата, найденных за один проход алгоритма. Для генетического алгоритма выборка кандидатов осуществлена из последнего поколения.

Для записей ЭЭГ, входящих в обучающую и тестовую выборки, выполнено шейплет-преобразование данных. Для векторов характерных признаков, сформированных для обучающей и тестовой выборок, построен классификатор и выполнено его тестирование. Использован комитет *ν-SVM* классификаторов, каждый из которых реализует модификацию метода опорных векторов (*англ. SupportVectorMachine*), в которой допускаются ошибки на обучающей выборке [10]. Варьируемый параметр $\nu \in [0;1]$ задает верхнюю границу доли ошибок обучения и нижнюю границу числа опорных векторов. В составе комитета использованы классификаторы, обученные со значениями $\nu = 0,1;\ 0,15;\ \ldots;\ 0,95$. Решение о принадлежности объекта к классу принималось после рейтингового голосования членов комитета. Матрица рейтингов оценивалась на основе точности прогнозирования классов членами комитета [11].

Использованные тестовые данные являются сильно несбалансированными (содержат большое число эпох ЭЭГ, соответствующих незначимым стимулам). В связи с этим оценка общей точности классификации может давать некор-

ректный результат. Так, высокая общая точность классификации может наблюдаться, если все объекты классификатор относит к одному классу, представителей которого в выборке больше, и при этом ни один объект из других классов не распознан верно. Поэтому для оценки точности классификации использована специальная мера G_{mean} [12, 13], которая представляет собой среднее геометрическое чувствительности и специфичности классификатора:

$$G_{\text{mean}} = = \sqrt{\frac{TP}{TP+FN}\,\frac{TN}{TN+FP}}\,.$$

Здесь TP — число истинно положительных результатов (запись ЭЭГ в действительности содержит волну Р300); TN — число истинно отрицательных результатов (запись ЭЭГ в действительности не содержит волны Р300); FN, FP — числа ложно отрицательных и ложно положительных результатов. Значение меры G_{mean} растет, если растут одновременно и чувствительность $\left(TP/\left(TP+FN\right)\right)$ и специфичность $\left(\left(TN/\left(TN+FP\right)\right)\right)$ классификатора, и стремится к нулю, если либо чувствительность, либо специфичность стремятся к нулю.

Значения меры G_{mean}, вычисленные для алгоритмов A_1, A_2, A_3, приведены в табл. 4 и на рис. 3. В столбце «Децимация сигнала» для сравнения приведены значения G_{mean}, полученные на векторах характерных признаков, сформированных путем понижения частоты выборок до 16 Гц в каждом канале. Такой прием позволяет отследить изменение низкочастотных составляющих ЭЭГ сигнала и часто применяется для формирования ВХП в работах, относящихся к ИМК на основе волны Р300 [14, 15].

Таблица 4

Значения меры G_{mean} для алгоритмов A_1, A_2, A_3

Испытуемый	Децимация сигнала	Алгоритм A_1	Алгоритм A_2	Алгоритм A_3
*s*3	0,74	0,72	0,70	0,68
*s*4	0,70	0,75	0,76	0,72
*s*5	0,69	0,74	0,73	0,73
*s*6	0,59	0,61	0,54	0,52
*s*7	0,68	0,66	0,67	0,69
*s*8	0,79	0,77	0,77	0,71
*s*9	0,79	0,73	0,72	0,71
*s*10	0,79	0,78	0,78	0,79
Среднее	0,72	0,72	0,71	0,69

Для данных, представленных в табл. 4, проверка гипотезы о равенстве средних также выполнена с помощью однофакторного дисперсионного анализа. Рассчитанное значение критерия Фишера составляет $F_{\text{эмп}} = 0,289$, что меньше критического значения $F_{\text{крит}} = 2,947$, определенного для уровня значимости $\alpha = 0,05$.

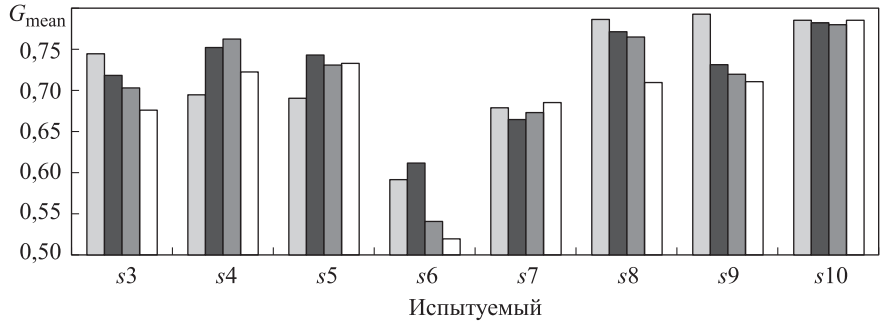


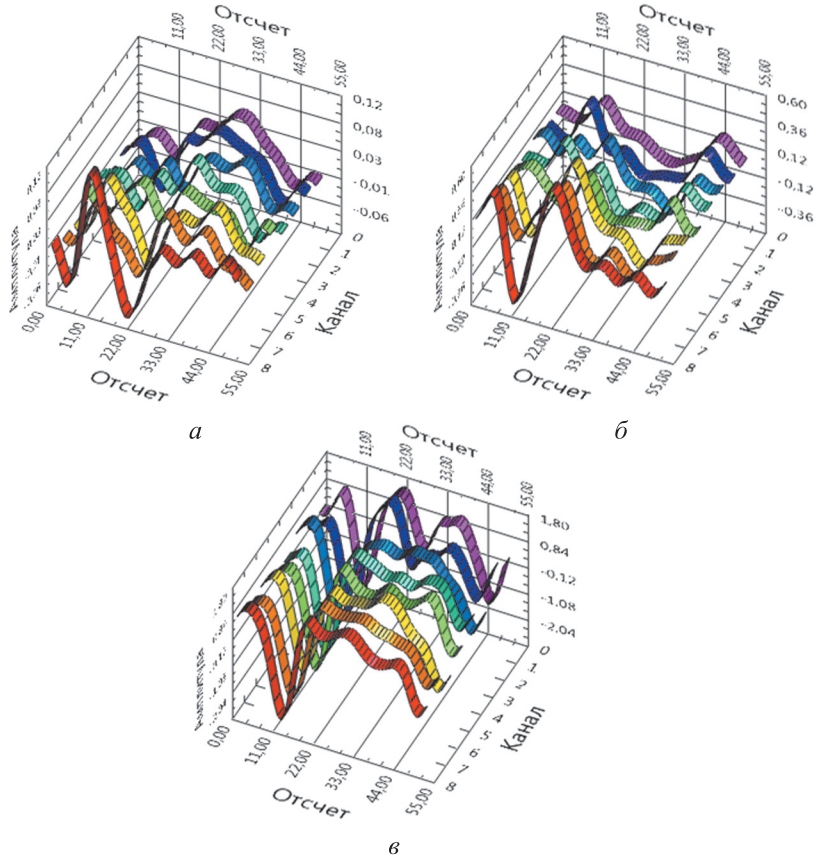
Рис. 3. Значения меры G_{mean} для алгоритмов A_1, A_2, A_3 (— децимация сигнала, — алгоритм A_1, — алгоритм A_2, — алгоритм A_3)

Приведенные результаты исследования показывают, что наборы шейплетов, которые были найдены алгоритмами A_1, A_2, A_3 и использованы для формирования характерных признаков сигнала ЭЭГ, обладают одинаковыми обобщающими свойствами. Наблюдаемые различия между алгоритмами не являются статистически значимыми.

Отметим следующее обстоятельство. Поскольку реакция на значимый стимул возникает в ЭЭГ спустя определенное время (равное примерно 300 мс), в ИМК на основе волны P300 важна не только информация о форме сигнала, которая «подмечается» шейплетами, но также информация о фазе, которую метод шейплетов не учитывает. Тем не менее, представленные результаты показывают, что метод шейплетов обладает эффективностью, сопоставимой с эффективностью классического метода выделения характерных признаков (основанного на понижении частоты выборок).

Для испытуемого *s*10 в качестве примера на рис. 4, *а* показана форма сигнала ЭЭГ после усреднения по эпохам, содержащим реакцию на незначимые стимулы; на рис. 4, *б* — после усреднения по эпохам, содержащим реакцию на значимые стимулы. На рис. 4, *в* приведена форма шейплета, найденного с помощью алгоритма полного перебора A_1. Видно, что шейплет в данном случае в большей степени отражает свойства класса, соответствующего реакции на незначимые стимулы.

Оценка влияния числа шейплетов в наборе на итоговую точность классификации. Рассмотрим, как исключение из набора похожих шейплетов влияет на точность классификации. Для каждого алгоритма классификации A_1, A_2, A_3 было сформировано пять новых наборов, состоящих из 2, 4, 8, 16, 24 шейплетов. Наборы сформированы с помощью алгоритма иерархической кластеризации [6]. Для каждого набора шейплетов вычислена оценка достижимой точности классификации путем расчета значений меры G_{mean} на тестовых данных. На рис. 5 приведены зависимости значений G_{mean}, усредненных по всем испытуемым, от числа шейплетов в наборе.



а *б* *в*

Рис. 4. Формы сигнала ЭЭГ и шейплета:

а и *б* — усредненные формы сигнала ЭЭГ, соответствующие реакции на незначимые и значимые стимулы; *в* — шейплет, обеспечивающий лучшее разделение классов

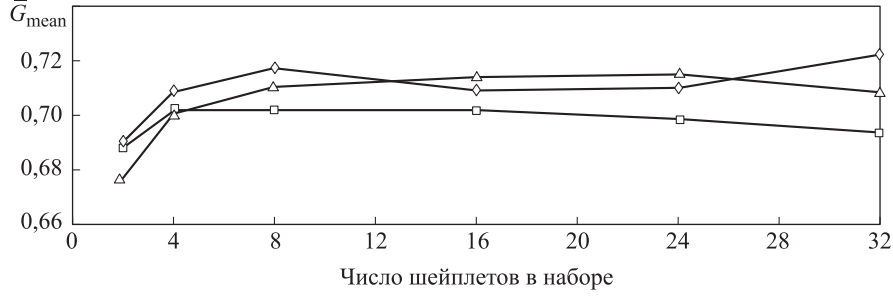


Рис. 5. Зависимость точности классификации $\bar{G}_{\text{mean}}$ от числа шейплетов в наборе (—◇— алгоритм A_1, —△— алгоритм A_2, —□— алгоритм A_3)

На рис. 5 видно некоторое повышение точности классификации алгоритмов A_2 и A_3 при уменьшении числа шейплетов в наборе. Эффект обусловлен тем, что алгоритм кластеризации, отбрасывая близкие шейплеты, позволяет исключить избыточные признаки, не несущие дополнительной информации о свойствах классов.

Вместе с тем малое число (менее четырех) шейплетов в наборе приводит к снижению точности классификации для всех алгоритмов классификации.

Заключение. В результате исследований выявлена возможность применения генетического алгоритма для повышения эффективности классификации многомерных временных рядов методом шейплетов. На известных тестовых данных, представляющих собой записи ЭЭГ, полученные при экспериментах с интерфейсом мозг–компьютер, установлено, что генетический алгоритм в среднем позволяет получить лучшее приближение к оптимальному значению, чем алгоритм поиска с отбрасыванием кандидатов. При этом генетический алгоритм обеспечивает сокращение множества перебора почти на 99 %.

Показана возможность применения предложенной модификации метода шейплетов для выделения характерных признаков сигнала ЭЭГ в приложениях интерфейса мозг–компьютер.

Для повышения точности классификации, достижимой на векторах-признаков, полученных с помощью шейплет-метода, имеет смысл рассмотреть другие стратегии включения шейплетов в набор: вместо выбора нескольких лучших кандидатов, целесообразно решать задачу поиска оптимального сочетания шейплетов. Такой подход составляет предмет для дальнейших исследований.

ЛИТЕРАТУРА

1. *Ye L., Keogh E.* Time series shapelets: a new primitive for data mining // Proc. 15th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. 2009. P. 947–956.

2. *Lines J., Davis L.M., Hills J., Bagnall A.* A shapelet transform for time series classification // Proc. 18th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. 2012. P. 289–297.

3. *Grabocka J., et al.* Scalable discovery of time-series shapelets. 2015. Cornell University, Technical Report arXiv:1503.03238. URL: https://arxiv.org/pdf/1503.03238.pdf (дата обращения: 25.01.2017).

4. *How* many people are able to control a P300-based brain–computer interface (BCI)? / C. Guger, S. Daban, E. Sellers, C. Holzner, G. Krausz // Neuroscience Letters. Vol. 462. No. 1. 2009. P. 94–98. DOI: 10.1016/j.neulet.2009.06.045
URL: http://www.sciencedirect.com/science/article/pii/S0304394009008192

5. *Rakthanmanon T., Keogh E.* Shapelets: a scalable algorithm for discovering time series shapelets // Proc. 13th SIAM Int. Conf. on Data Mining. 2013. P. 668–676.

6. *Classification* of time series by shapelet transformation / J. Hills, J. Lines, E. Baranauskas, J. Mapp, A. Bagnall // Data Mining and Knowledge Discovery. 2014. Vol. 28. No. 4. P. 851–881. DOI: 10.1007/s10618-013-0322-1
URL: http://link.springer.com/article/10.1007%2Fs10618-013-0322-1

7. *Grabocka J., Schilling N., Wistuba M., Schmidt-Thieme L.* Learning time-series shapelets // Proc. 20th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. 2014. P. 392–401.

8. *Карпенко А.П.* Современные алгоритмы поисковой оптимизации. М.: Изд-во МГТУ им. Н.Э. Баумана, 2014. 446 с.

9. *Lines J., Bagnall A.* Alternative quality measures for time series shapelets // Intelligent Data Engineering and Automated Learning — IDEAL 2012. 2012. Vol. 7435. P. 475–483.

10. *Chen P.H., Lin C.J., Scholkopf B.* A tutorial on ν-support vector machines // Applied Stochastic Models in Business and Industry. 2005. Vol. 21. No. 2. P. 111–136.
DOI: 10.1002/asmb.537 URL: http://onlinelibrary.wiley.com/doi/10.1002/asmb.537/abstract

11. *Нессонова М.Н.* Метод рейтингового голосования комитета алгоритмов в задачах классификации с учителем // Запорожский медицинский журнал. 2013. № 1. С. 101–102.
DOI: 10.14739/2310-1210.2013.1.15533 URL: http://zmj.zsmu.edu.ua/article/view/15533

12. *Kubat M., Holte R., Matwin S.* Learning when negative examples abound // Proc. 9th European Conf. on Machine Learning. LNCS. 1997. Vol. 1224. P. 146–153.

13. *Anand A., Pugalenthi G., Fogel G.B., Suganthan P.* An approach for classification of highly imbalanced data using weighting and undersampling // Amino Acids. 2010. Vol. 39. No. 5. P. 1385–1391. DOI: 10.1007/s00726-010-0595-2
URL: http://link.springer.com/article/10.1007/s00726-010-0595-2

14. *Hoffmann U., Vesin J., Diserens K., Ebrahimi T.* An efficient P300-based brain-computer interface for disabled subjects // Journal of Neuroscience Methods. 2008. Vol. 167. No. 1. P. 115–125. DOI: 10.1016/j.jneumeth.2007.03.005
URL: http://www.sciencedirect.com/science/article/pii/S0165027007001094

15. *Riccio A., Schettini F., Pizzimenti A.* Attention and P300-based BCI performance in people with amyotrophic lateral sclerosis // Frontiers in Human Neuroscience. 2013. Vol. 7. Article no. 732. DOI: 10.3389/fnhum.2013.00732
URL: http://journal.frontiersin.org/article/10.3389/fnhum.2013.00732/full

Карпенко Анатолий Павлович — д-р физ.-мат. наук, доцент, зав. кафедрой «Системы автоматизированного проектирования» МГТУ им. Н.Э. Баумана (Российская Федерация, 105005, Москва, 2-я Бауманская ул., д. 5).

Сотников Пётр Иванович — аспирант кафедры «Системы автоматизированного проектирования» МГТУ им. Н.Э. Баумана (Российская Федерация, 105005, Москва, 2-я Бауманская ул., д. 5), руководитель проекта ЗАО «Информтехника и Связь» (Российская Федерация, 107140, Москва, Верхняя Красносельская ул., д. 2/1, стр. 1).

Просьба ссылаться на эту статью следующим образом:

Карпенко А.П., Сотников П.И. Модифицированный метод классификации многомерных временных рядов с использованием шейплетов // Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение. 2017. № 2. С. 46–65. DOI: 10.18698/0236-3933-2017-2-46-65

MODIFIED CLASSIFICATION METHOD OF MULTIVARIATE TIME SERIES BASED ON SHAPELETS

A.P. Karpenko[1] apkarpenko@bmstu.ru
P.I. Sotnikov[1, 2] sotnikoffp@gmail.com

[1] Bauman Moscow State Technical University, Moscow, Russian Federation
[2] ZAO "Informtekhnika and Svyaz", Moscow, Russian Federation

Abstract

We consider the classification of multivariate time series using a paradigm, called shapelets. Instead of exhaustive search among all subsequences of the original time series, we suggest using a genetic algorithm for shapelets discovering. The problem of shapelets discovering is considered as a one-criterion optimization task. The quality of candidates acts as an objective function. Variable parameters are candidate attributes that define their position in the original dataset. We also propose measuring the quality of shapelets by assessing the classification accuracy. The assessment is made on a new dataset, where each object represents the distance vector from a shapelet to original time series. We evaluate efficiency of the proposed method modifications on the known electroencephalogram (EEG) recordings obtained for subjects performing a spelling task with P300-based brain-computer interface (BCI). The results show that these modifications can reduce the search space by nearly 99% with no loss of classification accuracy

Keywords

Time series, classification, shapelets, genetic algorithm, brain-computer interface

REFERENCES

[1] Ye L., Keogh E. Time series shapelets: a new primitive for data mining. *Proc. 15th ACM SIGKDD Int. Conf. on Knowledge discovery and data mining*. 2009, pp. 947–956.

[2] Lines J., Davis L.M., Hills J., Bagnall A. A shapelet transform for time series classification. *Proc. 18th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. 2012, pp. 289–297.

[3] Grabocka J., et al. Scalable discovery of time-series shapelets. 2015. Cornell University, Technical Report arXiv:1503.03238. Available at: https://arxiv.org/pdf/1503.03238.pdf (accessed 25.01.2017).

[4] Guger C., Daban S., Sellers E., Holzner C., Krausz G. How many people are able to control a P300-based brain–computer interface (BCI)? *Neuroscience Letters*, 2009, vol. 462, no. 1, pp. 94–98. DOI: 10.1016/j.neulet.2009.06.045
Available at: http://www.sciencedirect.com/science/article/pii/S0304394009008192

[5] Rakthanmanon T., Keogh E. Shapelets: a scalable algorithm for discovering time series shapelets. *Proc. 13th SIAM Int. Conf. on Data Mining*. 2013, pp. 668–676.

[6] Hills J., Lines J., Baranauskas E., Mapp J., Bagnall A. Classification of time series by shapelet transformation. *Data Mining and Knowledge Discovery*, 2014, vol. 28, no. 4, pp. 851–881.
DOI: 10.1007/s10618-013-0322-1
Available at: http://link.springer.com/article/10.1007%2Fs10618-013-0322-1

[7] Grabocka J., Schilling N., Wistuba M., Schmidt–Thieme L. Learning time-series shapelets. *Proc. 20th ACM SIGKDD Int. Conf. on Knowledge discovery and data mining*. 2014, pp. 392–401.

[8] Karpenko A.P. Sovremennye algoritmy poiskovoy optimizatsii [Modern search optimization algorithms]. Moscow, Bauman MSTU Publ., 2014. 446 p.

[9] Lines J., Bagnall A. Alternative quality measures for time series shapelets. *Intelligent Data Engineering and Automated Learning — IDEAL 2012*. 2012, vol. 7435, pp. 475–483.

[10] Chen P.H., Lin C.J., Scholkopf B. A tutorial on ν-support vector machines. *Applied Stochastic Models in Business and Industry*, 2005, vol. 21, no. 2, pp. 111–136.
DOI: 10.1002/asmb.537
Available at: http://onlinelibrary.wiley.com/doi/10.1002/asmb.537/abstract

[11] Nessonova M.N. Method of rating voting of algorithms committee in classification tasks with teacher. *Zaporozhskiy meditsinskiy zhurnal*, 2013, no. 1, pp. 101–102 (in Russ.).
DOI: 10.14739/2310-1210.2013.1.15533
Available at: http://zmj.zsmu.edu.ua/article/view/15533

[12] Kubat M., Holte R., Matwin S. Learning when negative examples abound. *Proc. 9th European Conf. on Machine Learning. LNCS*. 1997, vol. 1224, pp. 146–153.

[13] Anand A., Pugalenthi G., Fogel G.B., Suganthan P. An approach for classification of highly imbalanced data using weighting and undersampling. *Amino Acids*, 2010, vol. 39, no. 5, pp. 1385–1391. DOI: 10.1007/s00726-010-0595-2
Available at: http://link.springer.com/article/10.1007/s00726-010-0595-2

[14] Hoffmann U., Vesin J., Diserens K., Ebrahimi T. An efficient P300-based brain-computer interface for disabled subjects. *Journal of Neuroscience Methods*, 2008, vol. 167, no. 1, pp. 115–125. DOI: 10.1016/j.jneumeth.2007.03.005
Available at: http://www.sciencedirect.com/science/article/pii/S0165027007001094

[15] Riccio A., Schettini F., Pizzimenti A. Attention and P300-based BCI performance in people with amyotrophic lateral sclerosis. *Frontiers in Human Neuroscience*, 2013, vol. 7, article no. 732. DOI: 10.3389/fnhum.2013.00732
Available at: http://journal.frontiersin.org/article/10.3389/fnhum.2013.00732/full

Karpenko A.P. — Dr. Sc. (Phys.-Math.), Assoc. Professor, Head of Computer Aided Design Systems Department, Bauman Moscow State Technical University (2-ya Baumanskaya ul. 5, Moscow, 105005 Russian Federation).

Sotnikov P.I. — post-graduate student of Computer Aided Design Systems Department, Bauman Moscow State Technical University (2-ya Baumanskaya ul. 5, Moscow, 105005 Russian Federation), project manager at ZAO "Informtekhnika and Svyaz" (Verkhnyaya Krasnosel'skaya ul. 2/1, str. 1, Moscow, 107140 Russian Federation).

Please cite this article in English as:

Karpenko A.P., Sotnikov P.I. Modified Classification Method of Multivariate Time Series Based on Shapelets. *Vestn. Mosk. Gos. Tekh. Univ. im. N.E. Baumana, Priborostr.* [Herald of the Bauman Moscow State Tech. Univ., Instrum. Eng.], 2017, no. 2, pp. 46–65.
DOI: 10.18698/0236-3933-2017-2-46-65