

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ (государственный университет)
Факультет управления и прикладной математики
Кафедра «Интеллектуальные системы»
при Вычислительном центре им. А. А. Дородницына РАН

Моргачев Глеб Игоревич

**Построение метрического пространства
многоиндексных временных рядов с эффективно
вычисляемой функцией метрики**

010900 — Прикладные математика и физика

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Научный руководитель:
Доктор ф.-м. наук
Стрижов Вадим Викторович

Москва
2021 г.

Содержание

1	Введение	4
2	Постановка задачи	6
2.1	Общая задача построения метрического пространства	6
2.2	Задача классификации временных рядов	7
2.3	Задача классификации в пространстве представлений	7
3	Выравнивание временных рядов	8
3.1	Алгоритм динамического выравнивания DTW	8
3.2	Построение центроида	9
4	Построение функции расстояния	10
4.1	Алгоритм выравнивания	10
4.2	Алгоритм масштабирования	11
5	Аппроксимация выравнивающего пути	11
5.1	Выравнивающий путь из класса полиномов	11
5.1.1	Генетический алгоритм	12
5.2	Выравнивающий путь из класса монотонной нейросети	14
6	Вычислительный эксперимент	15
7	Результаты	19

Аннотация

В данной работе рассматривается задача построения метрического пространства временных рядов. Рассматриваемая функция расстояния инвариантна относительно монотонных преобразований осей времени. Стандартный подход к построению подобных функций расстояния предполагает использования метода динамического выравнивания, имеющего большую вычислительную сложность. В работе предлагается подход к уменьшению вычислительной сложности выравнивания временных осей, путем наложения ограничений на класс моделей, которому принадлежит выравнивающий путь.

Рассматривается поиск выравнивающего пути среди класса непрерывных параметризованных моделей: монотонных полиномов степеней 1 и 4, а также монотонных нейронных сетей. Применимость данного подхода исследуется на задачах многоклассовой классификации временных рядов методов ближайших соседей, а также на задаче построения шаблонов классов с помощью модифицированного метода DBA.

Эксперименты проведены на показаниях акселерометра: распознавание активностей человека по данным с носимых устройств и жестов с контроллера Wii, а также данные об использовании электроэнергии в домах при использовании различных устройств.

Ключевые слова: временные ряды, метрическая классификация временных рядов, динамическое выравнивание временных рядов, DBA.

1 Введение

Актуальность работы. В настоящий момент временные ряды являются распространённым способом представления данных в различных задачах: определения активности человека по данным ЭЭГ [11] или акселерометрам носимых устройств [10],

Выравнивание временных осей [9] является часто используемым инструментом для задания метрики в пространстве рядов, активно применяется для решения задач классификации [7], поиска подпоследовательностей [6]. Помимо того, что данный подход показывает высокие результаты в большом спектре прикладных задач, он остаётся хорошо интерпретируемым.

Недостатком выравнивания временных осей является его высокая вычислительная сложность. В работе предлагается метод, позволяющий снизить вычислительную сложность выравнивания за счёт ограничения на класс моделей, которому принадлежит выравнивающий путь. При этом, предлагаемый подход остаётся инвариантным относительно монотонных преобразований временной оси.

Применимость данного подхода проверяется на двух актуальных прикладных задачах, встречающихся в обработке временных рядов: многоклассовой классификации и построении наиболее репрезентативного представителя для класса объектов [14].

Цель работы. Целью работы является построение метрического пространства с функцией расстояния, инвариантной относительно монотонных преобразований оси времени. Данная функция расстояния должна быть вычислима за асимптотически эффективное время (должна быть эффективнее алгоритма динамического выравнивания) и хорошо подходить для решения задач классификации и построения шаблона.

Научная новизна. Предложен новый алгоритм вычисления пути выравнивания между парой временных рядов, решающий проблему высокой вычислительной сложности классического алгоритма динамического выравнивания. Предложенный подход сохраняет свойство инвариантности функции расстояния относительно сдвигов, растяжений и сжатий временной оси.

Практическая значимость Рассмотренный алгоритм имеет асимптотически

более низкую вычислительную сложность, а значит позволяет производить вычисления выравнивания между длинными временными рядами используя меньше ресурсов.

Использование вычислительных ресурсов является важным вопросом анализа временных рядов, так как многие задачи решаются на маломощных носимых устройствах.

Положения, выносимые на защиту:

- 1) Предложена функция расстояния для временными рядами;
- 2) Изучены различные модели аппроксимации выравнивающего пути между временными рядами;
- 3) Проведено сравнение предложенных алгоритмов выравнивания в задачах классификации и получения шаблонов классов.

Обзор литературы

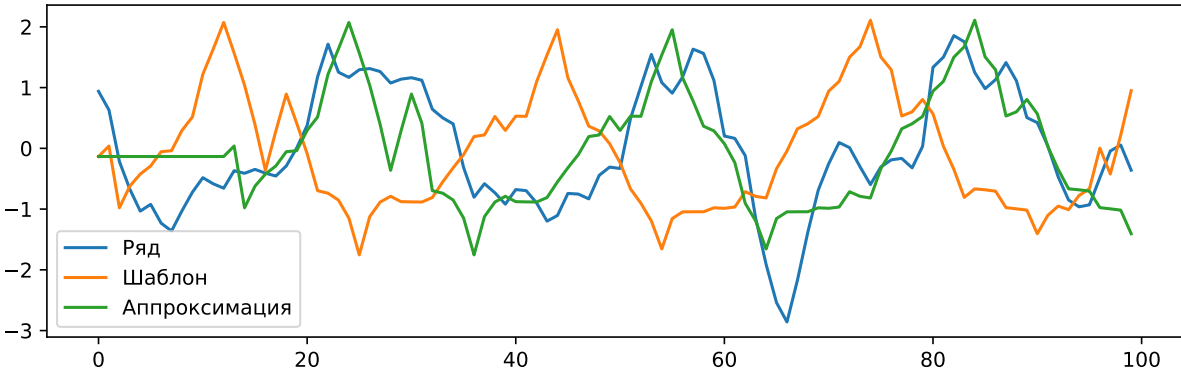
Определения функции расстояния между временными рядами является важной задачей анализа временных рядов. Их применяют для решения задач классификации [7, 20], кластеризации [1, 12], поиска подпоследовательностей [17], усреднения набора временных рядов [1, 14].

Наиболее простым способом задания расстояния является евклидово расстояние [1], однако данный подход неустойчив к деформациям оси времени. Применяются методы на основе выравнивания оси времени [], нахождении наибольшей общей подпоследовательности [18], нахождении наибольшей взвешенной подпоследовательности [13].

В некоторых работах предлагается использовать подход на основе обучения функции расстояния [16, 2].

В работе [17] исследуются оптимизации алгоритма динамического выравнивания и показывается, что они оказывают минимальное негативное влияние на качество работы.

Рис. 1: Пример аппроксимации временного ряда линейный шаблоном.



2 Постановка задачи

Определение 2.1. Функцией выравнивания π^f для пары временных $\mathbf{x}_1, \mathbf{x}_2, \in \mathbb{R}^n$, называется функция, отображающую временную ось ряда \mathbf{x}_i во временную ось ряда \mathbf{x}_j , т.е.:

$$\pi^f : \{1, \dots, n\} \rightarrow \{1, \dots, n\}.$$

Путь должен удовлетворять следующим условиям:

- 1) *Граничные условия:* $\pi^f(1) = 1, \pi^f(n) = n$.
- 2) *Монотонность:* $\pi^f(k + 1) - \pi^f(k) \geq 0$.

2.1 Общая задача построения метрического пространства

Задана выборка $\mathcal{D} = \{(\mathbf{x}_i, y_i), i = 1, \dots, N\}$, состоящая из N пар объект–метка класса. Где объекты \mathbf{x}_i являются временными рядами длины n , $\mathbf{x}_i \in \mathbb{R}^n$.

Ответ y_i - метка класса, $y_i \in \{1, \dots, k\}$.

Проводится разбиение выборки на обучающую и контрольную части.

Задано множество функций расстояния между парами временных рядов $\mathbf{R} = \{\mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}_+\}$ выравнивающих оси времени рядов друг относительно друга, т.е. соответствующая своей функции выравнивания.

Каждая функция выравнивания должна принадлежать одному из *классу допустимых функций* \mathbf{C}_j .

Рассматривается задача нахождения функции расстояния между временными рядами, вычислимой за асимптотически-эффективное время, которая является оп-

тимальной относительно внешних критериев качества Q_1, Q_2 :

$$\rho_i = \arg \max_{\rho \in \mathbf{R}} Q_i.$$

Здесь критерии качества Q_1, Q_2 являются критериями качества для задач классификации временных рядов и выделения репрезентативного представителя для класса соответственно.

2.2 Задача классификации временных рядов

Для выборки \mathfrak{D} заданной ранее и функции расстояния ρ , заданной на объектах, решается задача метрической многоклассовой классификации методом k ближайших соседей [4]. Другими словами строится отображение:

$$f_\rho : \mathbb{R}^n \rightarrow \{1, \dots, k\}.$$

В качестве критерия качества рассматривается точность:

$$Q_1 = \frac{1}{N} \sum_{i=1}^N [f_\rho(\mathbf{x}_i) = y_i].$$

2.3 Задача классификации в пространстве представлений

Для оценки качества шаблонов, получаемых с помощью рассматриваемых функций расстояния, рассматривается вспомогательная задача классификации исходной выборки \mathfrak{D} в пространстве *представлений* $\mathbf{H} \subset \mathbb{R}^k$, получаемом как расстояния до каждого из шаблонов $\mathbf{p}_i, i \in \{1, \dots, k\}$:

$$\mathbf{h}_i = [\rho(\mathbf{x}_i, \mathbf{p}_1), \dots, \rho(\mathbf{x}_i, \mathbf{p}_k)].$$

$$\mathbf{h}_i \in \mathbf{H}$$

Решается задача многоклассовой классификации в пространстве представлений \mathbf{H} с евклидовой функцией расстояния. Моделью классификации здесь выступает *метод опорных векторов*:

$$f : \mathbf{H} \rightarrow \{1, \dots, k\}.$$

В качестве критерия качества также рассматривается точность:

$$Q_2 = \frac{1}{N} \sum_{i=1}^N [f_\rho(\mathbf{x}_i) = y_i].$$

В роли шаблонов выступают *центроиды* классов.

3 Выравнивание временных рядов

Определение 3.1. *Выравнивающим путем $\pi_{\mathbf{x}_1, \mathbf{x}_2}$ для пары временных $\mathbf{x}_1, \mathbf{x}_2, \in \mathbb{R}^n$, называется последовательность пар:*

$$\pi_{\mathbf{x}_1, \mathbf{x}_2} = \{\pi_k\} = \{(i, j)_k \mid k \in \{1, \dots, K\}, i_k, j_k \in \{1, \dots, n\}\}.$$

Путь должен удовлетворять следующим условиям:

- 1) *Граничные условия:* $\pi_1 = (1, 1), \pi_K = (n, n)$.
- 2) *Непрерывность:* $\pi_k = (i, j), \pi_{k-1} = (i', j') : i - i' \leq 1, j - j' \leq n$.
- 3) *Монотонность:* $\pi_k = (i, j), \pi_{k-1} = (i', j') : i - i' \geq 0, j - j' \geq 0$.

Определение 3.2. *Стоимостью пути π между парой временных рядов $\mathbf{x}_1, \mathbf{x}_2, \in \mathbb{R}^n$ называется:*

$$cost(\pi) = \sqrt{\sum_{(i,j) \in \pi} (\mathbf{x}_1^i - \mathbf{x}_2^j)^2}.$$

Где \mathbf{x}^i – значение временного ряда \mathbf{x} в момент времени i .

Определение 3.3. *Оптимальным выравнивающим путём между парой временных рядов называется путь наименьшей стоимости.*

$$\pi^* = \arg \min_{\pi} cost(\pi)$$

3.1 Алгоритм динамического выравнивания DTW

В качестве расстояния между парой временных рядов используется стоимость пути наименьшей стоимости.

Задано два временных ряда, $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$. Для вычисления пути требуется построить матрицу размера $n \times n$ с элементами $D_{ij} = |\mathbf{x}_1^i - \mathbf{x}_2^j|$. Между рядами строится выравнивающий путь $\pi_{\mathbf{x}_1, \mathbf{x}_2}$, который минимизирует расстояние между ними.

Построение оптимального выравнивающего пути методом DTW проводится с помощью рекуррентной процедуры:

$$\gamma(1, j) = D_{1j}, \quad \gamma(i, 1) = D_{i1};$$

$$\gamma(i, j) = D_{ij} + \min(\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)).$$

Здесь $\gamma(i, j)$ суммарное расстояние, D_{ij} – расстояние в текущей клетке.

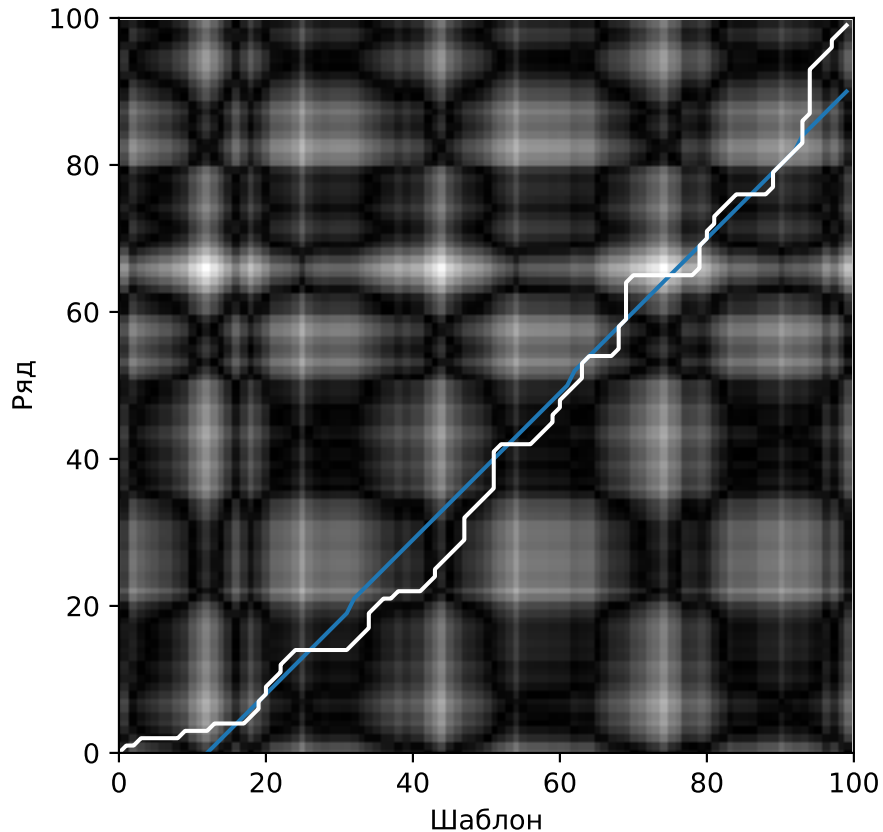
Элемент γ_{ij} заданной таким образом матрицы γ равен стоимости выравнивающего пути между первыми i и j значениями \mathbf{x} и \mathbf{y} соответственно.

Данная процедура позволяет найти путь наименьшей стоимости между парой рядов. Он соответствует аргументам γ , выбираемым на каждом шаге.

Функция расстояния, порождаемые данным алгоритмом выравнивания, соответствует стоимости оптимального пути, т.е.:

$$\rho_{DTW}(\mathbf{x}_1, \mathbf{x}_2) = \gamma(n, n)$$

Рис. 2: Аппроксимация выравнивающего пути линейной моделью.



3.2 Построение центраида

Определение 3.4. Для множества временных рядов $\mathbf{X} \subset \mathbb{R}^n$ центроидом по расстоянию ρ называется вектор $\mathbf{c} \in \mathbb{R}^n$, минимизирующая расстояние до всех объектов множества:

$$\mathbf{c} = \arg \min_{\mathbf{c} \in \mathbb{R}^n} \sum_{\mathbf{x} \in \mathbf{X}} \rho(\mathbf{c}, \mathbf{x}).$$

В случае DBA функция расстояния ρ порождена алгоритмами выравнивания.

Алгоритма построения центра [14] для выравнивающих функций расстояния итеративен и каждая итерация состоит из двух этапов:

- 1) Расчёт расстояния между всеми рядами класса и текущим приближением центра.
- 2) Перерасчет центра.

Сложность первой итерации алгоритма $\mathcal{O}(mn^2)$, где m – количество представителей класса. Сложность квадратична по n , так как рассматривается вычисления расстояния с помощью алгоритма динамического выравнивания.

Сложность второй итерации – $\mathcal{O}(mn)$. Таким образом, сложность алгоритма вычисления центра – $\mathcal{O}(Imn^2)$, где I – количество итераций.

Необходимо заметить, что основная сложность описанного выше алгоритма в вычислении выравнивающего пути с помощью DTW.

4 Построение функции расстояния

Предлагается функция ρ_{approx} расстояния между парами временных $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$:

$$\rho_{approx} : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}_+.$$

В процессе вычисления функции расстояния, предлагается рассматривать один из временных рядов как шаблон, которым будет аппроксимироваться второй временной ряд. Будем рассматривать \mathbf{x}_2 как шаблон. Аппроксимация будет происходить в два этапа:

- 1) Выравнивание временных осей.
- 2) Масштабирование.

4.1 Алгоритм выравнивания

Выравнивание производится с помощью функция пути π^f . В качестве допустимых классов выравнивания $\mathbf{C}_1, \mathbf{C}_2$, рассматриваются:

- 1) Монотонные полиномы степеней 1 и 4. Выравнивающий путь в данном подходе находится с помощью *генетического алгоритма*, описание которого приведено в 5.1.
- 2) Полносвязной монотонной нейронной сети [19]. Описание алгоритма и анализ работы приведены в 5.2.

Оба данных подхода предполагают, что функция выравнивания является некоторой параметризованной функцией $\pi^f(\mathbf{w})$. Задачей выравнивания является определение оптимальных параметров:

$$\mathbf{w}^* = \arg \min_{\mathbf{w}} \|\mathbf{x}_1 - \mathbf{x}_2[\pi^f(\mathbf{w})]\|_2.$$

$\mathbf{x}_2^{aligned} = \mathbf{x}_2[\pi^f(\mathbf{w})]$ обозначает, что i -ая компонента ряда \mathbf{x}_2 заменяется на $\pi^f(\mathbf{w}, i)$ -ую компоненту ряда \mathbf{x}_1 . То есть, временная ось шаблона выровнена, чтобы соответствовать временной оси ряда \mathbf{x}_1 .

4.2 Алгоритм масштабирования

Масштабирование выровненных рядов, проводится с помощью алгоритма SEMOR, основанного на использовании линейной регрессии для минимизации расстояния между соответствующими значениями. Таким образом, шаблон приводится к масштабу ряда \mathbf{x}_1 и добавляется тот же сдвиг:

$$\hat{\mathbf{x}}_2(w_1, w_2) = w_1 + w_2 \mathbf{x}_2^{aligned},$$

$$w_{1,2}^* = \arg \min_{w_{1,2}} \|\mathbf{x}_1 - \hat{\mathbf{x}}_2(w_1, w_2)\|_2.$$

На рис. 2 приведен пример аппроксимации временного ряда шаблоном.

5 Аппроксимация выравнивающего пути

5.1 Выравнивающий путь из класса полиномов

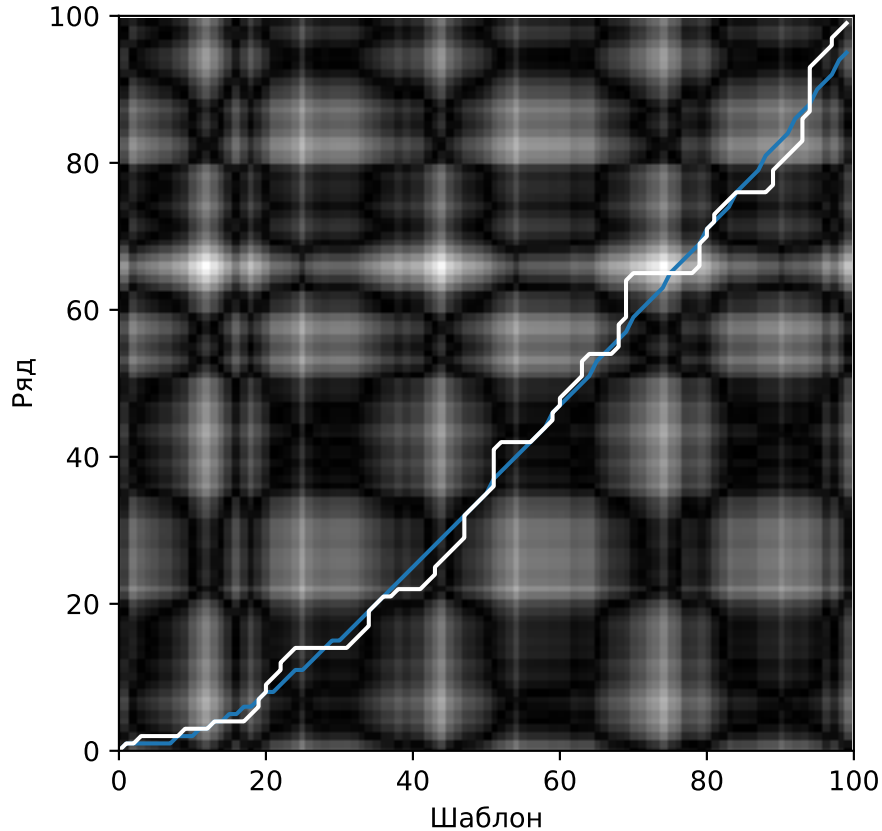
В данном подходе предлагается выбирать выравнивающий путь из класса монотонных полиномов степени k :

$$p_k(t) = \sum_{i=0}^k w_i * k^i,$$

для которых выполняется условие монотонности.

Таким образом, для полинома степени k необходимо обучить $k + 1$ параметров. Обучение будет происходить с помощью *генетического алгоритма* [5]. Он позволяет выбрать путь выравнивание без вычисления всех попарных расстояний между точками временных рядов, тем самым снижая сложность нахождения пути.

Рис. 3: Аппроксимация выравнивающего пути полиномом степени 4.



5.1.1 Генетический алгоритм

Далее подробнее описываются стадии генетического алгоритма.

- 1) **Генерация начальной популяции.** Генерируется начальное множество путей, то есть начальное множество параметров \mathbf{w} .
- 2) **Скращивание.** На данном этапе создаются новые пути на основе существующих. Для двух пар родителей $\mathbf{w}_1, \mathbf{w}_2$ новым путем будет

$$\mathbf{w}_{new} = \frac{\mathbf{w}_1 + \mathbf{w}_2}{2}$$

- 3) **Мутации.** Здесь каждый путь изменяется на случайную величину.
- 4) **Оценка.** На этапе оценки для каждого пути π^f считается сумма весов (\tilde{n}) случайных измерений $Q_{\pi}^f = \sum_i^{\tilde{n}} q_{\pi^f, i}$.

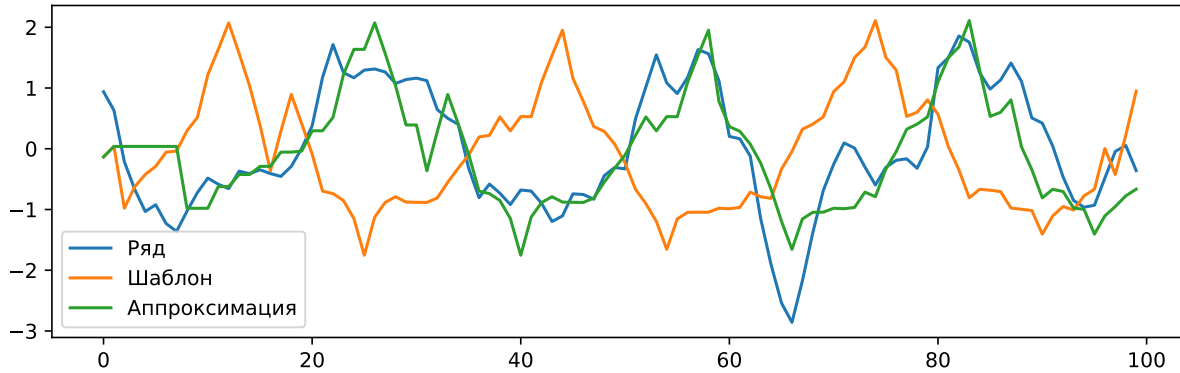
Весом измерения i называется:

$$q_{\pi^f, i} = (\mathbf{x}_1^i - \mathbf{x}_2[\pi^f(\mathbf{w})]^i)^2$$

Компоненты, по которым будет проводиться оценка, сэмплятся на каждой итерации работы алгоритма выравнивания из равномерного на $\{1, \dots, n\}$ распределения, без повторов.

- 5) **Селекция.** На каждой итерации отбираются пути с наименьшим весом Q .
- 6) **Критерий останова.** Алгоритм останавливает свою работу, когда информационная энтропия популяции (суммарный вес путей) меньше заданного порога.

Рис. 4: Пример аппроксимации временного ряда полиномом степени 4.



Утверждение 5.1. Сложность одной итерации описанного алгоритма $\mathcal{O}(p\tilde{n})$, где p - размер популяции, а \tilde{n} - количество рассматриваемых измерений при оценке веса пути Q .

Доказательство.

Необходимо вычислять расстояние между точками только \tilde{n} раз для каждого пути из популяции. Отсюда, функцию выравнивания $\pi^f(\mathbf{w})$ также необходимо вычислять только \tilde{n} раз для каждого пути из популяции.

Следовательно, сложность:

$$p * (\mathcal{O}(\tilde{n} + \tilde{n})) = \mathcal{O}(p\tilde{n}).$$

■

На рис. 1 приведен пример работы алгоритма выравнивания.

Утверждение 5.2. Сложность вычисления функции расстояния с аппроксимацией пути выравнивания полиномами $\rho_{poly} = \mathcal{O}(I p \tilde{n} + n)$, где I - размер популяции.

Если считать размер популяции и число итераций не зависимыми от n , то сложность линейна по длине временного ряда.

Доказательство.

Первая часть очевидна, так как максимально возможно I итераций. Т.е. полная сложность выравнивания $\mathcal{O}(Ip\tilde{n})$. Сложность вычисления расстояния между путем и выровненным шаблоном $\mathcal{O}(n)$.

Если I, p от n не зависят, то

$$\mathcal{O}(Ip\tilde{n} + n) = \mathcal{O}(\tilde{n} + n) \leq \mathcal{O}(n + n) = \mathcal{O}(n),$$

так как $\tilde{n} \leq n$, поскольку не имеет смысла проводить оценку более чем в n точках. ■

Утверждение 5.3. Сложность вычисления центроида с помощью алгоритма 3.2, на основе функции $\rho_{poly} = \mathcal{O}(I_c m(Ip\tilde{n} + n))$, где I_c - количество итераций алгоритма вычисления центроида.

Если считать размер популяции и число итераций не зависимыми от n , то сложность линейна по длине временного ряда.

Доказательство.

Следует из утверждения 5.2. ■

Утверждение 5.4. Для функции расстояния, описанной в данном пункте не выполняются свойства метрики, помимо неотрицательности.

5.2 Выравнивающий путь из класса монотонной нейросети

Функция выравнивания берётся из множества функций представимых с помощью монотонной нейронной сети [19].

Данная архитектура позволяет аппроксимировать некоторую функцию так, что полученная аппроксимация будет гарантирована монотонной по одному из параметров.

Соответственно, функция выравнивания π^f рассматривается как:

$$\pi^f(t) = \int_0^t \pi^{f'}(t) dt + \pi^f(0),$$

где производная функции выравнивания $\pi^{f'}$ - положительно определена.

В предлагаемом подходе $\pi^{f'}$ моделируется с помощью полносвязной нейронной сети, аргументов которой, помимо t , являются временные ряды $\mathbf{x}_1, \mathbf{x}_2$, между которыми вычисляется расстояние.

Таким образом, число аргументов нейронной сети равно $2n + 1$. Данная нейронная сеть обучаются с помощью стохастического градиентного спуска на обучающей выборке, для которой вычислены попарные расстояния ρ_{DTW} .

Таким образом, нейронная сеть явно учится предсказывать оптимальный путь выравнивания.

Утверждение 5.5. *Сложность вычисления выравнивающего пути с помощью предложенного алгоритма, для фиксированных размерностей нейронной сети – $\mathcal{O}(n)$.*

Доказательство.

Аналогично 5.2. ■

Утверждение 5.6. *Сложность вычисления центроида с помощью алгоритма 3.2, на основе функции $\rho_{nn} = \mathcal{O}(I_c m n)$, где I_c - количество итераций алгоритма вычисления центроида.*

Доказательство.

Следует из утверждения 5.5. ■

Утверждение 5.7. *Для функции расстояния, описанной в данном пункте не выполняются свойства метрики, помимо неотрицательности.*

6 Вычислительный эксперимент

В целях проверки применимости предложенных подходов была проведена серия экспериментов на следующий наборах данных:

- 1) Распознавание активностей человека по показаниям акселерометра телефона UCI HAR [15]. В данном наборе данных содержатся временные ряды, соответствующие шести активностям человека: ходьба, подъём по лестнице, спуск по лестнице, сидение, стояние, лежание. Длина временных рядов после обработки была равна 21.
- 2) Распознавание жестов по показаниям акселерометра контроллера **Wii** [8]. Временные ряды данного набора данных относились к 10 классам: взятие предмета, тряска, движение вправо, движение влево, движение вверх, движение вниз, круговое движение по часовой стрелке, круговое движение против часовой стрелки, движение за пределы экрана, движение обратно на экран.

Функция расстояния	Скорость, ит./с	Q_1
Классификация активности человека		
<i>DTW</i>	19.0	0.71 ± 0.02
<i>DTW</i> _{<i>p</i>₁}	12.8	0.53 ± 0.06
<i>DTW</i> _{<i>p</i>₄}	13.5	0.56 ± 0.04
<i>NN</i>	90.2	0.51 ± 0.06
Распознавание жестов		
<i>DTW</i>	2.2	0.46 ± 0.04
<i>DTW</i> _{<i>p</i>₁}	4.4	0.43 ± 0.06
<i>DTW</i> _{<i>p</i>₄}	4.3	0.41 ± 0.04
<i>NN</i>	–	–
Распознавание электроприборов		
<i>DTW</i>	2.4	0.71 ± 0.02
<i>DTW</i> _{<i>p</i>₁}	2.2	0.40 ± 0.06
<i>DTW</i> _{<i>p</i>₄}	2.2	0.29 ± 0.04
<i>NN</i>	–	–

Таблица 1: Классификация в пространстве представлений

Были взяты показания акселерометра по оси x . Рассматривались фрагменты временных рядов длиной в 100 измерений.

- 3) Данные об использовании электроэнергии в домах [3]. Временные ряды соответствуют потреблению энергии при различных комбинациях используемых в доме электроприборов.

Временные ряда длиной в 80 измерений, каждый записывался в течение двух минут.

Все выборки были разбиты на обучающую и тестовую части. Каждый ряд был нормализован:

$$\mathbf{x}_n = \frac{\mathbf{x} - E[\mathbf{x}]}{Var[\mathbf{x}]}.$$

Эксперимент с монотонной нейронной сетью проведен только для набора данных об электроприборах, так как данная архитектура позволила работать только с временными рядами малой длины. Данный подход показал малое время работы, по сравнению с другими подходами, однако в виду отсутствия результатов работы данного подхода на длинных временных рядах, его сравнение с остальными методами

Функция расстояния	Скорость, ит./с	Q_2
Классификация активности человека		
<i>DTW</i>	18.2	0.61 ± 0.05
<i>DTW</i> _{<i>p</i>₁}	12.2	0.48 ± 0.04
<i>DTW</i> _{<i>p</i>₄}	10.3	0.51 ± 0.02
<i>NN</i>	90.0	0.41 ± 0.06
Распознавание жестов		
<i>DTW</i>	1.8	0.38 ± 0.04
<i>DTW</i> _{<i>p</i>₁}	2.3	0.40 ± 0.03
<i>DTW</i> _{<i>p</i>₄}	2.3	0.37 ± 0.03
<i>NN</i>	–	–
Распознавание электроприборов		
<i>DTW</i>	2.2	0.44 ± 0.11
<i>DTW</i> _{<i>p</i>₁}	2.1	0.30 ± 0.02
<i>DTW</i> _{<i>p</i>₄}	2.2	0.26 ± 0.09
<i>NN</i>	–	–

Таблица 2: Классификация в пространстве представлений

было бы некорректно.

Можно заметить, что на задаче классификации временных рядов с помощью метода k ближайших соседей (таблица 1), на всех наборах данных наибольшая точность, как и ожидалось, у алгоритма динамического выравнивания.

Аппроксимации с помощью полиномов степеней 1 и 4 показывают близкие результаты на обеих задачах, что может быть объяснено сложностью обучения более сложной функции с помощью генетического алгоритма.

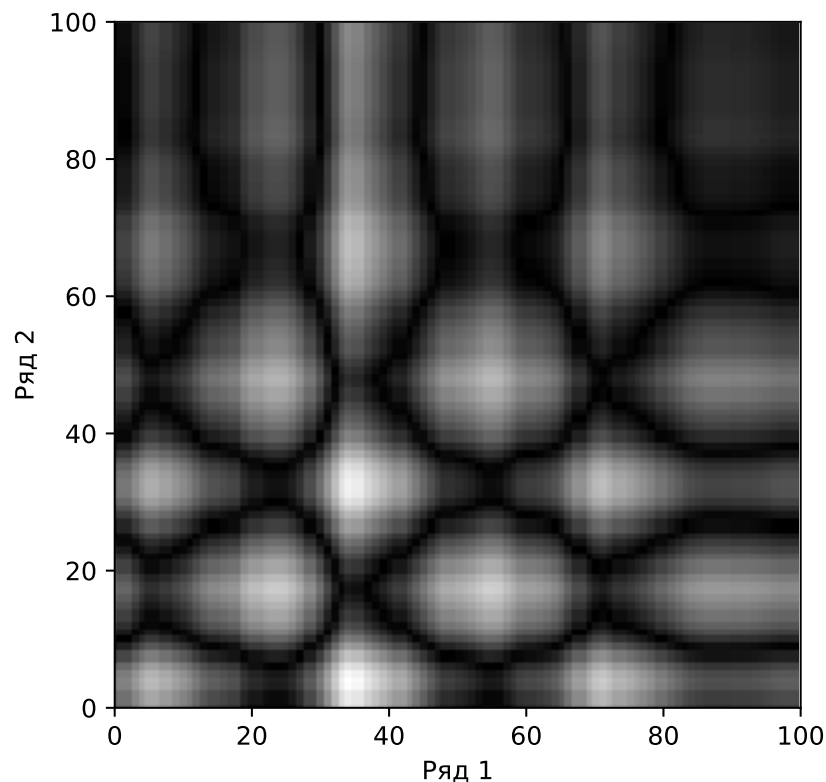
Для коротких временных рядов алгоритм динамического выравнивания показывает значительно меньшее время вычисления функции расстояния, чем подходы на основе аппроксимации полиномами (18 ит./с. против 10 ит./с.) для датасета активности человека в задаче построения шаблона.

Однако на длинных временных рядах быстрее работают алгоритмы, аппроксимирующие путь выравнивания.

Это подтверждает утверждение 5.2 об асимптотике предложенного алгоритма.

В задаче распознавания жестов, алгоритмы динамического выравнивания и аппроксимированного выравнивания показывают схожие результаты. Это может быть

Рис. 5: Матрица попарных расстояний между точками для распознавания жестов.

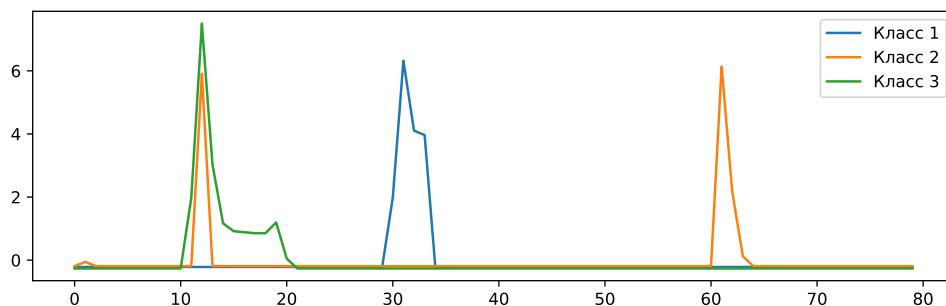


объяснено структурой данных: на рис. 5 видно, что матрица попарных расстояния между рядами одного класса для этого датасета, содержит в себе пути выравнивания простой структуры, который легко находится предлагаемой моделью аппроксимации.

Можно заметить, что на данных об используемых электроприборах, динамическое выравнивание показало значительно лучшие результаты, чем аппроксимирующие аналоги.

Причиной этого также может быть особенность структуры временных рядов: в этом датасете представители различных классов имеют схожую структуру и для их различения мало подходят приближенные алгоритмы. Пример данных рядов можно увидеть на рис. 6.

Рис. 6: Примеры объектов из датасета используемых устройств.



7 Результаты

В работе предложен подход к заданию функции расстояния между парой временных рядов, выравнивающий оси временных рядов друг относительно друга. В ходе работы алгоритм ищет выравнивающий путь среди заданного класса функций.

Были исследованы классы монотонных полиномов, а также монотонных нейронных сетей простой структуры.

Главным достоинством предложенного подхода является его эффективная работа с длинными последовательностями. Были доказаны асимптотические оценки времени работы.

Предложенный подход был проверен на задачах метрической классификации и построение центраида набора рядов. Проведено сравнение с классическим алгоритмом выравнивания и произведен анализ ошибок.

В качестве тестовых данных для экспериментальной проверки были использованы данные акселерометра телефона, акселерометра контроллера игровой приставки, а также показания потребления электроэнергии жилых домов.

Список литературы

- [1] *Anh D. T., Thanh L. H.* An efficient implementation of k-means clustering for time series data with dtw distance // *Int. J. of Business Intelligence and Data Mining.* — 2015. — 20. — Vol. 10. — Pp. 213–232.
<http://http://www.inderscience.com/link.php?id=71311>
- [2] Deep multiple metric learning for time series classification / Z. Chen, Y. Liu, J. Zhu et al. // *IEEE Access.* — 2021. — Vol. 9. — Pp. 17829–17842.
- [3] Electric devices dataset.
<http://http://www.timeseriesclassification.com/description.php?Dataset=ElectricDevices>
- [4] *Fix E., Hodges Jr J. L.* Discriminatory analysis-nonparametric discrimination: consistency properties: Tech. rep.: California Univ Berkeley, 1951.
- [5] *Goldberg D. E.* Genetic Algorithms in Search, Optimization, and Machine Learning. — New York: Addison-Wesley, 1989.
- [6] *Goncharov A.* Weighted dynamic time warping for optimal subsequence search // EAGE Publications. — Tyumen: 2019.
- [7] *Goncharov A. V., Strijov V. V.* Analysis of dissimilarity set between time series // *Computational Mathematics and Modeling.* — 2018. — Jul. — Vol. 29, no. 3. — Pp. 359–366.
<http://https://doi.org/10.1007/s10598-018-9415-4>
- [8] *Guna J., Humar I., Pogacnik M.* Intuitive gesture based user identification system. — 2012. — 07. — Pp. 629–633.
- [9] *Keogh E. J., Pazzani M. J.* Scaling up dynamic time warping to massive dataset // Proceedings of the Third European Conference on Principles of Data Mining and Knowledge Discovery. — PKDD '99. — London, UK, UK: Springer-Verlag, 1999. — Pp. 1–11.
<http://http://dl.acm.org/citation.cfm?id=645803.669511>

- [10] *Kwapisz J. R., Weiss G. M., Moore S. A.* Activity recognition using cell phone accelerometers // *SIGKDD Explor. Newsl.* — 2011.
[http://http://doi.acm.org/10.1145/1964897.1964918](http://doi.acm.org/10.1145/1964897.1964918)
- [11] *Kwessi E., Edwards L.* Analysis of eeg data using complex geometric structurization. — 2021.
- [12] *Liao T. W.* Clustering of time series data—a survey // *Pattern Recognition.* — 2005. — Vol. 38, no. 11. — Pp. 1857 – 1874.
[http://http://www.sciencedirect.com/science/article/pii/S0031320305001305](http://www.sciencedirect.com/science/article/pii/S0031320305001305)
- [13] *Morse M. D., Patel J.* An efficient and accurate method for evaluating time series similarity // *SIGMOD '07.* — 2007.
- [14] *Petitjean F., Ketterlin A., Gançarski P.* A global averaging method for dynamic time warping, with applications to clustering // *Pattern Recognition.* — 2011. — Vol. 44, no. 3. — Pp. 678–693.
- [15] A public domain dataset for real-life human activity recognition using smartphone sensors / D. Garcia-Gonzalez, D. Rivero, E. Fernandez-Blanco, M. R. Luaces // *Sensors.* — 2020. — Vol. 20, no. 8.
<http://https://www.mdpi.com/1424-8220/20/8/2200>
- [16] Routine modeling with time series metric learning / P. Compagnon, G. Lefebvre, S. Duffner, C. Garcia // *CoRR.* — 2019. — Vol. abs/1907.04666.
[http://http://arxiv.org/abs/1907.04666](http://arxiv.org/abs/1907.04666)
- [17] Searching and mining trillions of time series subsequences under dynamic time warping / T. Rakthanmanon, B. Campana, A. Mueen et al. // *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* — KDD '12. — New York, NY, USA: ACM, 2012. — Pp. 262–270.
[http://http://doi.acm.org/10.1145/2339530.2339576](http://doi.acm.org/10.1145/2339530.2339576)
- [18] *Vlachos M., Kollios G., Gunopulos D.* Discovering similar multidimensional trajectories // *Proceedings 18th International Conference on Data Engineering.* — 2002. — Pp. 673–684.

- [19] *Wehenkel A., Louppe G.* Unconstrained monotonic neural networks // *CoRR*. — 2019. — Vol. abs/1908.05164.
<http://http://arxiv.org/abs/1908.05164>
- [20] *Гончаров , Попова , Стрижов* Метрическая классификация временных рядов с выравниванием относительно центроидов классов // *Системы и средства информатики*. — 2015. — Vol. 25(4). — Pp. 52–64.
<http://http://strijov.com/papers/Goncharov2015MetricClassification.pdf>