Динамическое выравнивание многомерных временных рядов*

Гончаров А. В., Моргачев Г. И., Смирнов В. , Липницкая Т.
morgachev.gi@phystech.edu, smirnov.vs@phystech.edu, tanya.lipnizky@yandex.ru
МФТИ

В данной работе исследуется кластеризация многомерных временных рядов с использованием алгоритма DTW. При использовании DTW в многомерном случае возникает проблема определения функций расстояния между элементами временных рядов. Основной целью статьи является нахождение зависимости качества кластеризации от выбора этой функции расстояния. В связи с повышением размерности возникает вопрос эффективности и применимости DTW на многомерных рядах. В качестве прикладной задачи исследуется кластеризация размеченных данных о деятельности человека полученных с акселерометра. Оценка качества кластеризации производится при сравнении с результатами кластеризации на основе авторегрессионной модели и анализу распределения классов данных в полученных кластерах.

Ключевые слова: *временные ряды, многомерные временные ряды, DTW.*

1 Введение

Для описания различных данных широко используются временные ряды. Чтобы найти их сходство вводится функция расстояния, однако стандартный поточечный подход не является информативным вследствие того, что ряды могут содержать общие паттерны, деформированные относительно временной оси: претерпевшие сдвиги либо сжатия [1]. Одним из способов решения этой проблемы является выравнивание временных рядов (DTW) [2] и его модификаций [3]. Этот подход в большом спектре задач позволяет достичь максимального качества среди его аналогов.

В работе рассматривается применения DTW для кластеризации рядов и поиска подпоследовательностей соответствующих некоторому классу событий для случая многомерных рядов. Использование DTW на подобных данных описано в [4], [5]. В работе [4] предлагается способ выравнивания многомерных рядов, основанный на нормализации исходных данных и нахождениии векторной нормы. В [5] рассматривается алгоритм, позволяющий выполнить выравнивание временных рядов между координатами. Многомерное DTW предполагает различные варианты выравнивания, такие как выравнивание относительно общей временной шкалы и между соответсвующими каналами.

В процессе работы алгоритма DTW происходит вычисление расстояний между точками сравниваемых рядов. Поскольку в многомерном случае координаты точек описываются векторами, на результат будет влиять выбор функций расстояния между ними. Исследование влияние выбора этих функций на качество кластеризации является главной особенностью этой работы. В работе используются функции расстояния порождённые L_1 и L_2 нормами.

Ещё одним стандартным подходом к нахождению сходства между рядами является сравнение представления рядов коэффициентами их регрессионных моделей. Полученная в ходе работы DTW кластеризация сравнивается кластеризацией на основе авторегрессионной модели.

---

* Работа выполнена при финансовой поддержке РФФИ, проект № 00-00-00000. Научный руководитель: Гончаров А. В. Задачу поставил: Гончаров А. В. Консультант: Гончаров А. В.

В статьях [6] [7] рассматриваются различные виды алгоритмов кластеризации временных рядов, среди которых неплохие результаты показывают варианты иерархической кластеризации. Данный вид кластеризации был выбран в качестве базового.

Данные [8] представляют собой измерения акселерометра некоторого носимого устройства, например мобильного телефона, находящегося в кармане человека, и используется для индентификации действия человека в конкретный момент времени. Данные разделены на 6 классов: ходьба, бег, подъём по лестнице, спуск по лестнице, сидение, лежание.

2 Постановка задачи

Временнным рядом называется упорядоченная последовательность измерений $S = s_1, s_2, ..., s_n$, где n - длина временного ряда.

Поскольку мы используем многомерные временные ряды, то $s_i, i \in 1 \ldots n$, представляют собой вектор размерности l. Например, при рассмотрении задачи идентификации определенного движения человека вектор является трёхмерным с координатами (x, y, z).

Пусть задано множество временных рядов $\mathbb{S} \subset \mathbb{R}^{l \times n}$. $\forall S_i \in \mathbb{S}$ задано $y_i \in \mathbb{Y}$ - множество меток классов.

Пусть есть множество функций расстояния между векторами R:

$$\mathrm{R} = \{\rho : \mathbb{R}^l \times \mathbb{R}^l \to \mathbb{R}^+\}$$

Пусть $S_i,\ S_j \in \mathbb{S}$, тогда определим некоторые метрики, которые возмём в качестве функций расстояния между векторами:

1. $L_1(v, u) = \sum\limits_{l=1}^{l} |v_i - u_i|$
2. $L_2(v, u) = \sqrt{\sum\limits_{l=1}^{l} (v_i - u_i)^2}$, евклидово расстояние, также будем обозначать его $ED(u, v)$
3. $cosine(v, u) = \dfrac{(v, u)}{\sqrt{(v, v)}\sqrt{(u, u)}}$

Данные метрики будут использоваться во время работы алгоритма DTW, для измерений расстояния между двумя точками рядов.
Каждой функции расстояния ρ между векторами соответствует функция расстояния между временными рядами:

$$DTW_\rho : \mathbb{S} \times \mathbb{S} \to \mathbb{R}^+$$

Целью эксперимента является изучения влияния выбора поточечной метрики на результат работы DTW на примере двух задач:

1. Поиск шаблонов в ряду
2. Кластеризация рядов

Кроме того, производится сравнения качества и скорости работы данного подхода с другими используемыми методами:

1. Функция расстояния на основе поточечных расстояний между рядами, при равным длинах рядов:

$$ED(S_1, S_2) = \sum_{1}^{m} \|s_{1i} - s_{2i}\|_2$$

2. Применения авторегрессионной модели для описания ряда.

2.1 Поиск паттернов

Рассматривается задача поиска заранее известных паттернов во временном ряду. Пусть есть временной ряд $\mathcal{A}$ длинны n, некоторые участки которого соответствуют событию класса P. Класс событий P - в нашем случае, набор временных рядов длины $m \ll n$.
Пусть известно $P_k \subset P$ - k экземпляров класса P, *не содержащихся в* $\mathcal{A}$.
Необходимо найти участки $\mathcal{A}$, соответствующие данному классу.
Обозначим множество начал таких участков как $T = \{t_1, \ldots, t_j\}$.

Будем считать участок найденым, если его пересечение с предполагаемым участком более 80% длины m. Поиск будет проводиться путем нахождения расстояния с помощью DTW_ρ между фрагментами ряда $\mathcal{A}$ и шаблоном, полученным из известных экземпляров класса.

Рассматриваемая функция качества:

$$Q(DTW_\rho, A, P_k, T) = \frac{\sum\limits_{i=1}^{j}[t_i - \text{найден}]}{j}$$

Тогда, решаемая задача:

$$Q(DTW_\rho, A, P_k, T) \to \max_\rho$$

2.2 Кластеризация

Исходные данные представляют собой временные ряды (длиной n)
Возьмем выборку $S \subset \mathbb{S}, \quad |S| = N$
Определим матрицу попарных расстояния:

$$D(DTW_\rho(S)) = ||D_{ij}||, \quad D_{ij} = DTW_\rho(S_i, S_j), \quad S_i, S_j \in S$$

Определим кластеризатор:

$$f : D \to Z^N$$

Где Z - множество меток кластеров.
Будем рассматривать следующие функции качества:

$$Q_1(f(D), S) = \frac{1}{|Z|}\sum_{z \in Z} \max_y \frac{N_z^y}{N_z}$$

$$Q_2(f(D), S) = \frac{1}{|Z|}\sum_{z \in Z} \max_y \frac{(N_z^y)^2}{N_z N^y}$$

Здесь:

- N_z - количество элементов в кластере с меткой z.
- N^y - количество элементов в классе y.
- N_z^y - количество элементов класса y в классе z.

Тогда, решаемая задача:

$$Q_i(D(DTW_\rho(S), S) \to \max_\rho$$

3 Описание основных методов

Для построения функции выравнивания и проверки её качества используются модель DTW (и её отптимазации).

3.1 Описание функции расстояния между объектами

В данной работе в качестве метрического расстояния между объектами предлагается использовать строимость *пути наименьшей стоимости* между объектами.

Dynamic time warping - измерение расстояния между двумя временными рядами.

Задано два временных ряда, X длины m_1 и Y длины m_1.

$$X = x_1, x_2, ..., x_i, ..., x_{m_1}$$
$$Y = y_1, y_2, ..., y_j, ..., y_{m_2}$$
$$x_i, y_j \in \mathbb{R}^n$$

Требуется построить матрицу размера $m_1 \times m_2$ с элементами $D_{ij} = d(x_i, y_j)$, где d - выбранная метрика.

Чтобы найти наибольшее соответсвие между рядами нужно найти выравнивающий путь W, который минимизирует расстояние между ними. W - набор смежных элементов матрицы D, $w_k = (i, j)_k$.

$W = w_1, w_2, ..., w_k, ..., w_K$

$max(m_1, m_2) \leqslant K \leqslant m_1 + m_2 + 1$, где K-длина выравнивающего пути

Выравнивающий путь должен удовлетворять следующим условиям:

1. $w_1 = (1, 1)$, $w_K = (m_1, m_2)$
2. $w_k = (a, b)$, $w_{k-1} = (a', b') : a - a' \leqslant 1$, $b - b' \leqslant 1$
3. $w_k = (a, b)$, $w_{k-1} = (a', b') : a - a' \geqslant 0$, $b - b' \geqslant 0$

Оптимальный выравнивающий путь должен минимизировать выравнивающую стоимость пути:

$$DTW(X, Y) = \sum_{k=1}^{K} w_k$$

Путь находится рекуррентно:
$\gamma(i, j) = d(q_i, c_j) + min(\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1))$, где $\gamma(i, j)$ суммарное расстояние, $d(q_i, c_j)$ расстояние в текущей клетке.

Кроме того, выравнивающий путь ограничивают тем, насколько он может отклоняться от диагонали. Типичным ограничением является полоса Сако-Чиба, в которой говорится, что путь искривления не может отклоняться от диагонали больше, чем на определённый процент клеток.

3.2 Описание алгоритма кластеризации

В качестве алгоритма кластеризации используется иерархическая кластеризация, который базируется на последовательном слияние ближайших кластеров. Рассматриваются различные функции расстояния между кластерами:

1. *complete:* $d(A, B) = \max_{a \in A, b \in B}(dist(a, b))$
2. *weighted:* $d(A, B) = \dfrac{(dist(S, B) + dist(T, B))}{2}$, где кластер $A = S \cup T$
3. *weighted:* $d(u, v) = \sum_{a \in A, b \in B} \dfrac{d(a, b)}{(|A| * |B|)}$

3.3 Описание авторегрессионного подхода

Авторегрессия - представляет собой подход, в котором элемент временного ряда представляется линейной комбинацией некоторого числа прошлых элементов.

Пусть есть временной ряд $X = x_1, ..., x_n,\ x \in \mathbb{R}^n$. Размер окна авторегрессионной модели l.

В модели авторегрессии

$$x_k = \sum_{i=1}^{l} k_i \cdot x_{k-i},\ k_i \in \mathbb{R}$$

$\boldsymbol{k} = \{k_i\}$ - коэффициенты, обучаемые для наилучшего описания выборки.

В дальнейшей работе с временным рядом, набор коэффициентов используется как вектор описания ряда.

В нашей работе, на множестве векторов коэффициентов производится иерархическая классификация с L_2 расстоянием между векторами.

4 Оптимизации

В основу нашей реализации легла работа [9], в которой было использовано большое количество оптимизаций, позволяющих как сократить время вычисления расстояния между рядами, так и, при возможности, вообще не вычислять его. В ходе работы мы постарались обобщить эти оптимизации на многомерный случай. Ниже приведено их краткое описание.

4.1 Использование квадрата расстояния

В DTW вычисляется квадратный корень, однако, если упустить этот шаг, относительное расстояние не изменится, поскольку обе функции монотонны и вогнуты. Это упрощает вычисления и позволяет сделать модель легкой для понимания. Таким образом, говоря о DTW, мы подразумеваем квадратные аналоги.

4.2 Использование нижней границы

Для того чтобы ускорить последовательный поиск в DTW, используется нижняя граница (Lower Bounding), чтобы отбросить неподходящие последовательности (подпоследовательности). Оптимизация ускоряет поиск ещё и потому что не требует затратных вычислений.

В эксперименте используется каскадная нижняя граница. Вначале последовательность проходит проверку на требования LB_{Kim}, которая использует расстояние между максимальными значениями рядов и минимальными значениями рядов. Однако, для того чтобы сравнить последовательности, они должны быть нормализованы, поэтому значения двух расстояний между максимальными и минимальными точками могут быть ничтожно малы. В случае если последовательность удовлетворила требованиям LB_{Kim}, происходит вторичная проверка LB_{Keogh}, использующей ED.

Также бывает полезным менять роли сравниваемых последовательностей для LB_{Keogh}, от этого будет меняться решение к какой последовательности применяется нижняя граница, причем результаты вычисления этих границ для каждой из последовательностей при их сравнении в общем случае не будут равны. Однако данный метод применяется опционально и только в том случае, если другие нижние границы не проявили себя.

4.3 Использование верхней границы

При вычислении LB_{Keogh}, мы можем заметить, что текущая сумма расстояний между каждой парой точек привысила определённое (наибольшее возможное) значение, в таком случае мы можем прекратить подсчёт, тк дальнейший действия дадут ещё более высокий результат.

Если вся LB_{Keogh} была посчитана и мы обнаружили, что должны вычислить DTW полностью, есть способ отбросить лишние вычислительные затраты на стадии подсчёта DTW. Если постепенно вычислять DTW слева направо от 1 до k и суммировать частичное накопление DTW с вкладом от LB_{Keogh} от k+1 до n. Сумма $DTW(S^i_{1:k}, S^j_{1:k}) + LB_{Keogh}(S^i_{k+1:n}, S^j_{k+1:n})$ является нижней границей для $DTW(S^i_{1:n}, S^j_{1:n})$. Если в какой-то момент такая нижняя граница превысит верхнюю, расчёт прекращается. Кроме того, расходы на расчёт такой границы незначительны.

Во многих случаях становится полезным изменить порядок поиска верхней границы. Очевидно, что разный порядок поиска приносит разное ускорение, более того существует n! выриантов упорядочивания. Чтобы найти оптимальный вариант, предлагается отсортировать индексы основанных на абсолютных значениях Z-нормализованной последовательности. Одно значение временного ряда S_i сравнивается со многими из ряда S_j, которые далее сортируются по убыванию вклада ED.

5 План эксперимента

5.1 Поиск паттернов

1. Была взята выборка, состоящая из большого числа размеченных рядов размерности 3, длиной по 182 точки. В каждом классе было выбрано случайное подмножество рядов для получения шаблона с помощью алгоритма DBA (или простого усреднения) Путем склейки остальных был получен один длинный ряд, каждый участок которого соответствовал некоторому классу.
2. С помощью алгоритма DBA был получен шаблон, для поиска наиболее близких к нему участков для каждого из классов.
3. Для каждого шаблона запускался алгоритм для поиска k наиболее близких к нему участков большого ряда с использованием ED и DTW_ρ с различными ρ в качестве функций расстояния.
4. Для каждого класса вычислялась метрика качества Q
5. Проводилось сравнение метрик и производительности подходов на основе различных методов определения расстония между рядами.

5.2 Кластеризация рядов

1. Находилась выборка ...
2. Для каждого ряда из выборки проводилось обучения авторегрессионной модели и классификация на основе полученных коэффициентов регрессии
3. Высчитывалась матрицы попарных расстояние между рядами выборки для ED и DTW с различными внутренними расстояниями.
4. Проводилось иерархическая кластеризация элементов выборки на основание полученных попарных расстояний.

5. Расситывались метрики качества качества для класстеризации на основании авторегрессионной модели, суммы попарных расстояний между точками (ED) и DTW с различными ρ.
6. Проводилось сравнений полученных результатов.

6 Эксперимент

В ходе эксперимента были использованы данные акселерометра мобильного телефона. Они представляли собой временные ряды длинной в 600 точек ускорений по осям X, Y, Z. Из них была сгенерирована выборка из 2048 рядов по 50 точек. Каждых из рядов принадлежал к одному из шести возможных классов. Данные были равномерно распределены по всем классам.

Проводилась кластеризация этих данных описаными методами. В качестве расстояния между векторами использовались L_1 и L_2 нормы. Так как в процессе получения данных были возможны различные положения телефона в кармане, кластеризация проводилась на 24, 36 и 48 кластеров.

6.1 Результаты

		Q_1			Q_2		
ρ	n clust	*complete*	*average*	*weighted*	*complete*	*average*	*weighted*
L_1	24	0.5059	0.5854	0.6384	0.2732	0.3761	0.4488
	36	0.5325	0.6196	0.6163	0.2988	0.4246	0.4140
	48	0.5563	0.6388	0.6308	0.3303	0.4432	0.4306
L_2	24	0.4876	0.6216	0.6258	0.2701	0.4173	0.4246
	36	0.4982	0.6459	0.6433	0.2701	0.4545	0.4489
	48	0.5336	0.6486	0.6530	0.2701	0.4546	0.4615

7 Заключение

Литература

[1] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn Keogh. *Querying and mining of time series data: Experimental comparison of representations and distance measures*, volume 1, pages 1542–1552. 2 edition, 8 2008.

[2] Eamonn J. Keogh and Michael J. Pazzani. Derivative dynamic time warping. In *In SIAM International Conference on Data Mining*, 2001.

[3] Stan Salvador and Philip Chan. Toward accurate dynamic time warping in linear time and space. *Intell. Data Anal*, 11(5):561–580, 2007.

[4] G.A. ten Holt, M.J.T. Reinders, and E.A. Hendriks. Multi-dimensional dynamic time warping for gesture recognition. In *Thirteenth annual conference of the Advanced School for Computing and Imaging*, 2007.

[5] Parinya Sanguansat. Multiple multidimensional sequence alignment using generalized dynamic time warping. 2012.

[6] T. Warren Liao. Clustering of time series data—a survey. *Pattern Recognition*, 38(11):1857 – 1874, 2005.

[7] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. Time-series clustering – a decade review. *Information Systems*, 53:16 – 38, 2015.

[8] Jennifer R. Kwapisz, Gary M. Weiss, and Samuel A. Moore. Activity recognition using cell phone accelerometers. *SIGKDD Explor. Newsl.*, 12(2):74–82, March 2011.

[9] Thanawin Rakthanmanon, Bilson Campana, Abdullah Mueen, Gustavo Batista, Brandon Westover, Qiang Zhu, Jesin Zakaria, and Eamonn Keogh. Searching and mining trillions of time series subsequences under dynamic time warping. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 262–270, New York, NY, USA, 2012. ACM.