Выравнивание временных рядов: прогнозирование с использованием DTW*

А. А. Романенко
Московский физико-технический институт, ФУПМ, каф. «Интеллектуальные системы»

Временной ряд — это повсеместно встечающаяся форма представления данных во многих научных дисциплинах. Задача, сопутствующая появлению временных рядов, — сравнение одной последовательности данных с другой. Dynamic time warping (DTW) представляет собой технику эффективного выравнивая временных рядов. Методы DTW используются при распознавании речи, при анализе информации в робототехнике, в промышленности, в медицине и других сферах. Предлагается классический алгоритм DTW и упоминаются его возможные модификации. В работе описывается алгоритм поиска в последовательности подпоследовательности, «больше всего похожей» на данную последовательность. Приведены результаты работы алгоритма.
Ключевые слова: *выравнивание временных рядов, DTW алгоритм, time warping, warping path*

Введение

Временной ряд — это повсеместно встечающаяся форма представления данных во многих научных дисциплинах. Распространенная задача, связанная с временными рядами, это сравнение одной последовательности с другой. Например, прогнозирование цен на акции базируется на сравнивании текущей подпоследовательности ряда с найденной подпоследовательностью ряда, сохраненного в истории [11]. Для распознавания речи, рукописного текста и подписи успешно применяется методы, требующие сравнения двух временных рядов [4, 2]. В некоторых случаях достаточно в качестве расстояния между последовательностями выбрать евклидово расстояние. Но часто сравнение таким простым путем дает ошибочные результаты. Dynamic time warping (DTW) представляет собой технику эффективного выравнивая временных рядов. Методы DTW используются в перечисленных выше отраслях, при анализе информации в робототехнике [3], в медицине [1, 9], в биоинформатике [12] и других сферах.

Далее будет описан классический DTW алгоритм. Он дает неплохие результаты, но конечно же он не позволяет достичь наилучшего выравнивания. Это связано с тем, что этот алгоритм очень чувствителен к искажениям временного ряда по оси Y. Также описан алгоритм поиска в последовательности подпоследовательности, «больше всего похожей» на данную последовательность. Приведены результаты работы алгоритма на искусственных и реальных временных рядах.

Постановка задачи

Классический алгоритм DTW.

Пусть даны две последовательности Q и C (временных ряда) длиной n и m соответственно:

$$Q = q_1, q_2, \ldots, q_n,\ C = c_1, c_2, \ldots, c_m.$$

Научный руководитель В. В. Стрижов

Классический DTW алгоритм по этим последовательностям строит *путь наименьшей стоимости*. Поясним, что это значит.
Определим матрицу $\Omega^{n\times m}$ так, чтобы её элемент (i, j) соответствовал расстоянию между i-ым и j-ым элементами последовательностей Q и C, то есть соответствовал выравниванию между q_i и c_j. Мы будем брать евклидово расстояние:

$$d(q_i, c_j) = (q_i - c_j)^2.$$

В качестве метрики можно взять и другие функции, например:

$$d(q_i, c_j) = |q_i - c_j|.$$

По матрице Ω построим некоторый *путь* W. Этот путь выражает соответствие между Q и C. k-ый элемент W определяется как $w_k = (i, j)$. Далее под $d(w_k)$, где $w_k = (i, j)_k$, будем понимать $d(q_i, c_j)$, т. е.

$$d(w_k) = d(q_i, c_j) = (q_i - c_j)^2.$$

Итак, мы имеем

$$W = w_1, w_2, \ldots, w_k, \ldots, w_K,$$

где K — длина пути. K очевидно удовлетворяет следующему условию:

$$\min(m, n) \leqslant K < m + n - 1.$$

Пусть путь W удовлетворяет следующим условиям:

— Граничные условия
Обычно предполагают, что $w_1 = (1, 1)$ и $w_K = (n, m)$, т. е. начало и конец W находятся на диагонали в противоположных углах Ω.

— Непрерывность
Пусть $w_k = (a, b)$ и $w_{k-1} = (p, q)$. Тогда

$$a - p \leqslant 1, b - q \leqslant 1$$

Это ограничение нужно, чтобы в шаге пути W участвовали только соседние элементы матрицы (включая соседние по диагонали).

— Монотонность
Пусть $w_k = (a, b)$ и $w_{k-1} = (p, q)$. Тогда

$$a - p \geqslant 1, b - q \geqslant 1$$

Это ограничение нужно, чтобы точки W монотонно перемещались во времени.

Путей, удовлетворяющих этим трем условиям, может быть очень много. Однако нам нужен путь, на котором достигается минимум *стоимости пути*:

$$DTW(Q, C) = \min \left\{ \frac{1}{K} \sqrt{\sum_{k=1}^{K} d(w_k)} \right\}.$$

Знаменатель K нужен для того, чтобы учесть различную длину W.

Таким образом, *путь наименьшей стоимости (выравнивающий путь)* для последовательностей Q и C это путь W, на котором достигается минимум стоимости пути $DTW(Q, C)$.
Классичесский DTW алгоритм поиска пути минимальной стоимости рекурсивно находит длину пути наименьшей стоимости $\gamma_{i,j}$ до каждого элемента матрицы Ω :

$$\gamma_{i,j} = d(w_{i,j}) + \min(\gamma_{i,j-1}, \gamma_{i-1,j}, \gamma_{i-1,j-1})$$

О других способах вычисления $\gamma_{i,j}$ можно изнать из [5, 8, 6].
Заметим, что евклидово расстояние между Q и C — это частный случай алгоритма DTW, когда путь наименьшей стоимости W ограничен условием

$$w_k = (i, j)_k,\ i = j = k.$$

Постановка задачи. Теперь пусть даны две последовательности Q и C длиной n и m соответственно ($n \gg m$):

$$Q = q_1, q_2, \ldots, q_n,\ C = c_1, c_2, \ldots, c_m.$$

Требуется, используя классический DTW алгоритм, найти такую подпоследовательность Q' последовательности Q, которая «больше всего похожа» на C, т. е. ту подпоследовательность, на которой достигается значение функции

$$DTW(Q', C) = \min_{Q', W} \left\{ \frac{1}{K} \sqrt{\sum_{k=1}^{K} d(w_k)} \right\}. \quad (1)$$

Описание алгоритма поиска «похожей» подпоследовательности

Идея алгоритма состоит в том, чтобы с помощью первого применения классического DTW алгоритма отсеять подпоследовательности, которые не могут удовлетворить (1). А затем обычным перебором из оставшихся подпоследовательностей выбрать оптимальную с помощью классического DTW алгоритма.

Вычислительный эксперимент

Алгоритм тестировался как на искусственных временных рядах, так и на реальных.

Поиск подпоследовательности в синусоиде. Возьмем в качестве Q синусоиду на отрезке $[0; \pi]$, длина Q: $n = 1001$. А в качестве C возьмем ее подпоследовательность длины $m = 350$. С помощью нашего алгоритма найдем подпоследовательность Q' и путь W,на которых выполняется (1). (Рис. 1–3)

Как видно из графиков, алгоритм нашел оптимальную подпоследовательность Q' и на простейших модельных данных работает правильно.

Поиск подпоследовательности в зашумленной синусоиде Проведем тест подобный предыдущему, только в качестве Q возьмем зашумленную синусоиду, а в качестве C — часть гладкой синусоиды. (Рис. 4–6)

Видно, что алгоритм работает правильно, подпоследовательность найдена верно.
Стоит отметить, что эксперимент проводился при 5% шуме. При больших шумах алгоритм прекращал работать.

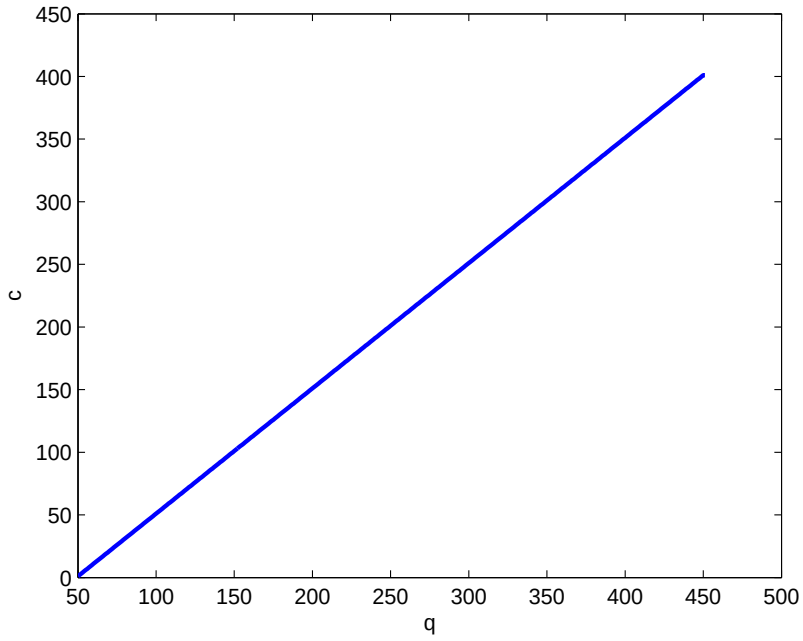

Рис. 1. Выравнивающий путь W.

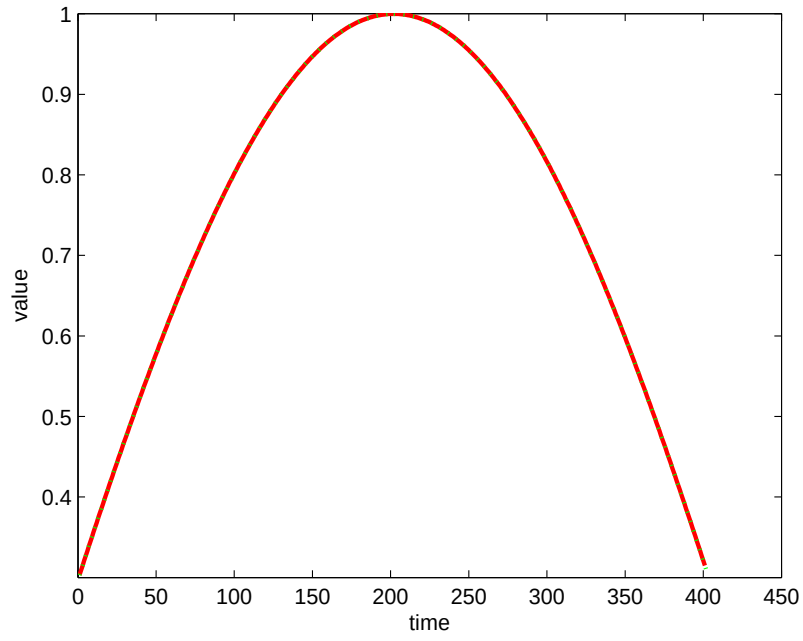

Рис. 2. Последовательность C (красным цветом) и найденная подпоследовательность Q' (зеленым цветом).

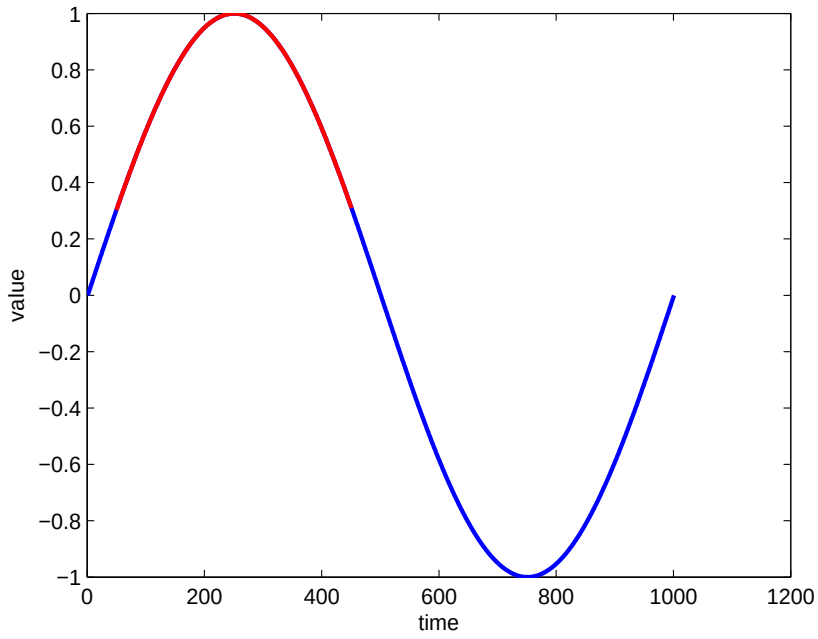

Рис. 3. Последовательность Q и подпоследовательность Q' в ней.

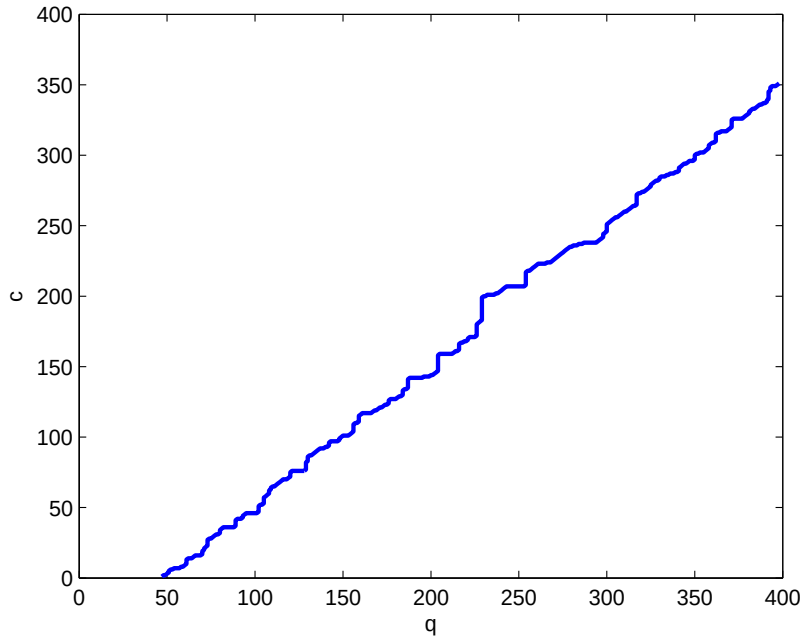

Рис. 4. Выравнивающий путь W.

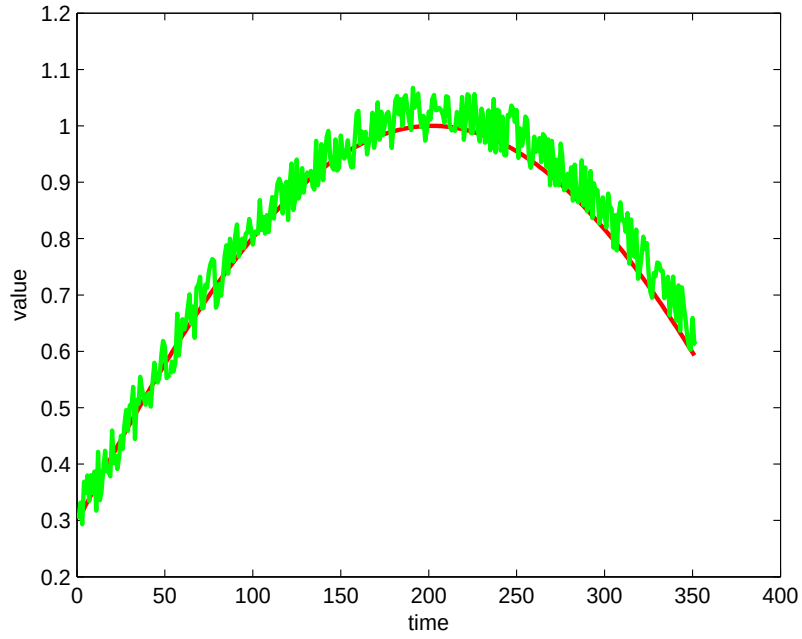

Рис. 5. Последовательность C (красным цветом) и найденная подпоследовательность Q' (зеленым цветом).

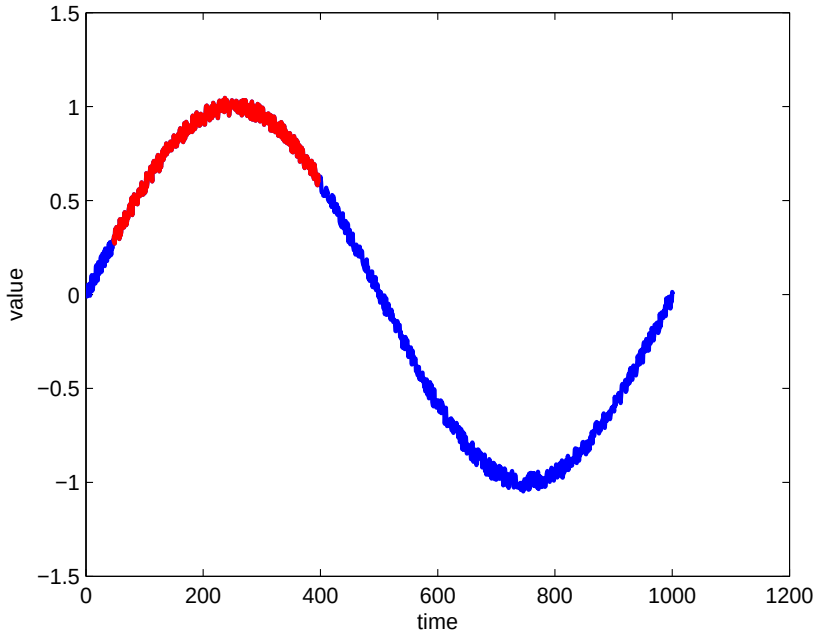

Рис. 6. Последовательность Q и подпоследовательность Q' в ней.

Работа на реальных данных

Далее предоставлены результаты работы алгоритма на реальных данных. В качестве Q возьмем зависимость продаж некого товара от времени, в качестве C небольшую последовательность, не являющуюся подпоследовательностью Q. (Рис. 7–9)

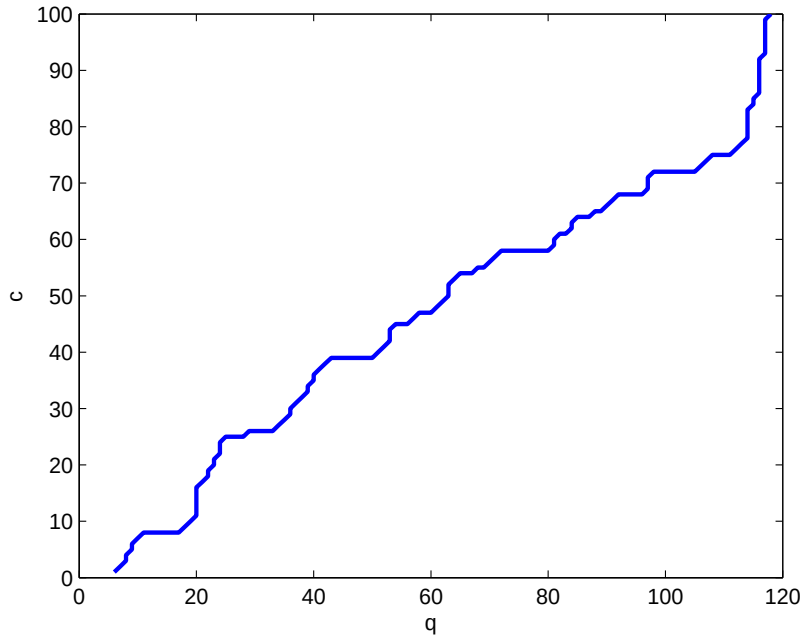

Рис. 7. Выравнивающий путь W.

На Рис. 7 конец выравнивающего пути практически вертикален. Это означает, что хвост последовательности C нужно немного растянуть по оси времени, что подтверждает Рис. 8.

Заключение

В работе рассматривается классический DTW алгоритм. Об его улучшениях и модификациях можно узнать из [10, 5, 8, 6, 7]. Также рассмотрим алгоритм поиска «похожей» подпоследовательности, основывающийся на классическом DTW алгоритме. Приведены примеры работы этого алгоритма.

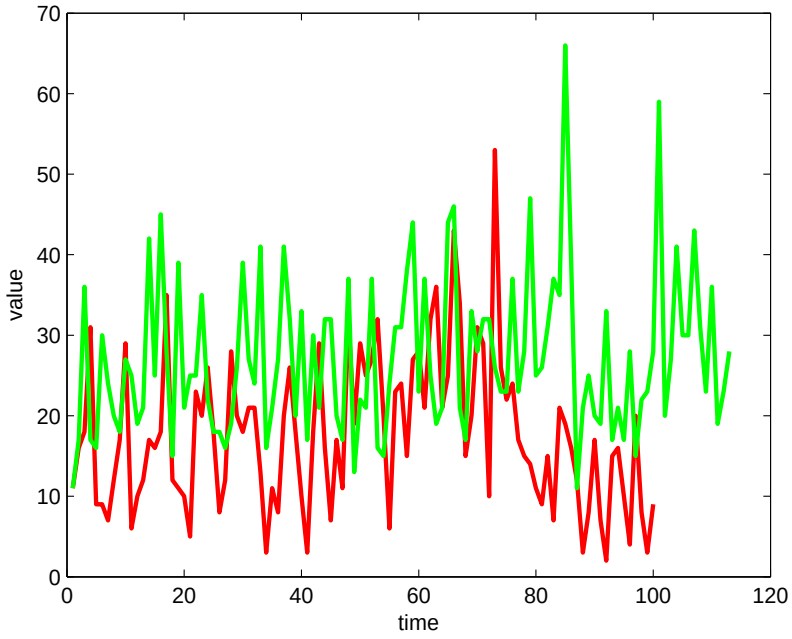

Рис. 8. Последовательность C (красным цветом) и найденная подпоследовательность Q' (зеленым цветом).

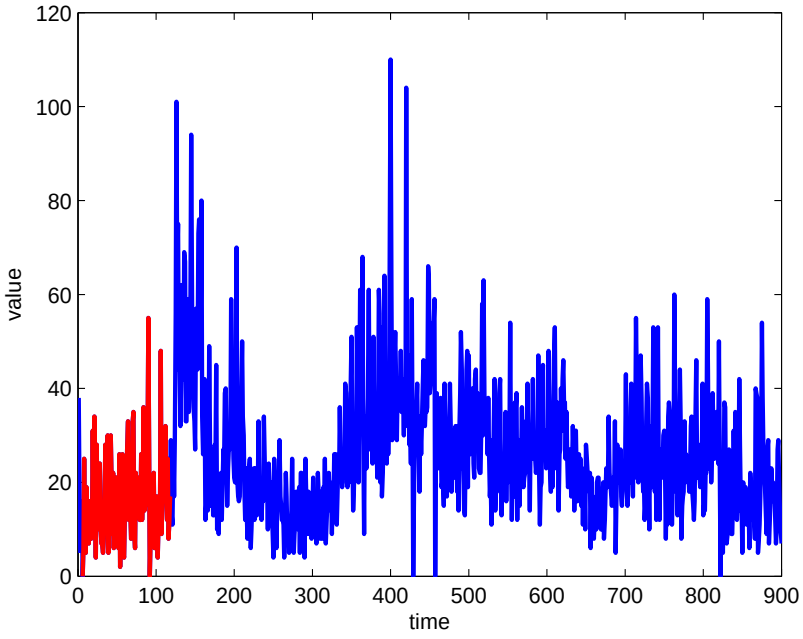

Рис. 9. Последовательность Q и подпоследовательность Q' в ней.

Литература

[1] Vullings, H. J. L. M.; Verhaegen, M. H. G., Verbruggen, H. B. *Time-Warping Proceedings of the Second International Symposium on Advances in Intelligent Data Analysis, Reasoning about Data*, ECG Segmentation Using Time-Warping, 1997.

[2] Sakoe, Hiroaki and Chiba, Seibi *Readings in speech recognition*, 1990, pp.159-165.

[3] Oates, Tim and Schmill, Matthew D. and Cohen, Paul R. *A Method for Clustering the Experiences of a Mobile Robot that Accords with Human Judgments*, Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, 2000.

[4] Ralph Niels *Dynamic Time Warping: An intuitive way of handwriting recognition?*, 2004.

[5] Eamonn J. Keogh and Michael J. Pazzani *Derivative Dynamic Time Warping*, In First SIAM International Conference on Data Mining (SDM'2001), 2001.

[6] Eamonn J. Keogh and Michael J. Pazzani *Scaling up Dynamic Time Warping to Massive Datasets*, 1999.

[7] Selina Chu and Eamonn Keogh and David Hart and Michael Pazzani *Iterative Deepening Dynamic Time Warping for Time Series*, In Proc 2 nd SIAM International Conference on Data Mining, 2002.

[8] Ann Chotirat and Ratanamahatana Eamonn and Keogh *Everything you know about Dynamic Time Warping is Wrong*, The 31st Annual International Symposium on Forecasting, 2004.

[9] E.G. Caiani and A. Porta and G. Turiel and M. Muzzupappa and S. Pieruzzi and F. Grema and C. Malliani and A. Cerutti and S. Cerutti *Warped-average template technique to track on a cycle-by-cycle basis the cardiac filling phases on left ventricular volume*, IEEE Computers in Cardiology, 1998.

[10] Donald J. Berndt and James Clifford *Using Dynamic Time Warping to Find Patterns in Time Series*, KDD Workshop, 1994.

[11] Georgios N. Banavas and Sue Denham and Michael J. Denham *Fast Nonlinear Deterministic Forecasting Of Segmented Stock Indices Using Pattern Matching And Embedding Techniques*, Society for Computational Economics, 2000.

[12] John Aach and George M. Church *Aligning Gene Expression Time Series With Time Warping Algorithms*, 2001.