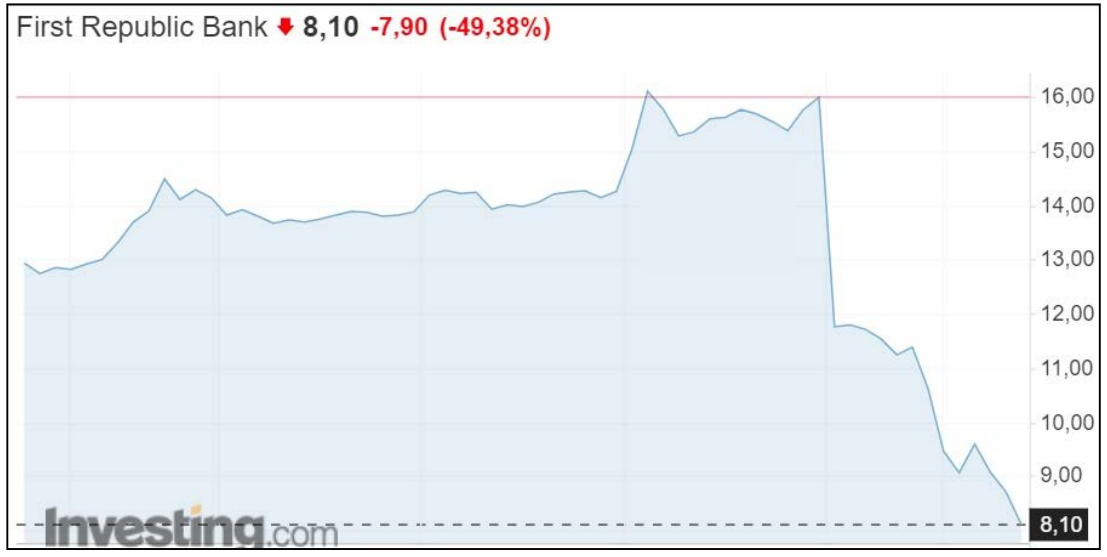
Поиск подпоследовательностей временного ряда по образцу



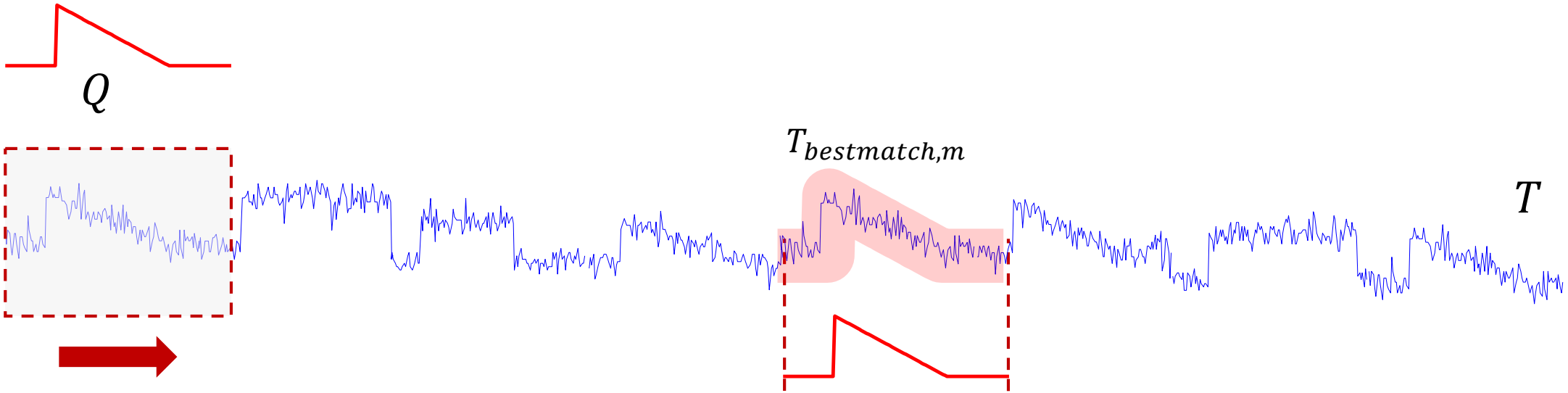
Возьми себе в образец героя древних времен, наблюдай его, иди за ним вслед, поравняйся, обгони – слава тебе!

А.В. Суворов

Содержание

- Постановка задачи
- Метрика Евклида
- Мера DTW
- Поиск по образцу на основе DTW
- Оптимизации поиска на основе DTW

Поиск по образцу (subsequence matching/similarity search)



В ряде T найти подпоследовательность $T_{bestmatch,m}$, наиболее похожую на запрос Q:

$$\forall T_{i,m} \in S_T^m \;\; \mathrm{Dist}\left(T_{bestmatch,m}, Q\right) \leq \mathrm{Dist}\left(T_{i,m}, Q\right)$$

Содержание

- Постановка задачи
- **Метрика Евклида**
- Мера DTW
- Поиск по образцу на основе DTW
- Оптимизации поиска на основе DTW

Расстояние (метрика) $\mathrm{Dist}: M \times M \to \mathbb{R}$**:** $\forall x, y, z \in M$ **выполнены**

- Аксиома тождества:

$$\mathrm{Dist}(x, x) = 0$$

- Аксиома положительности:

$$\mathrm{Dist}(x, y) \geq 0$$

- Аксиома симметричности:

$$\mathrm{Dist}(x, y) = \mathrm{Dist}(y, x)$$

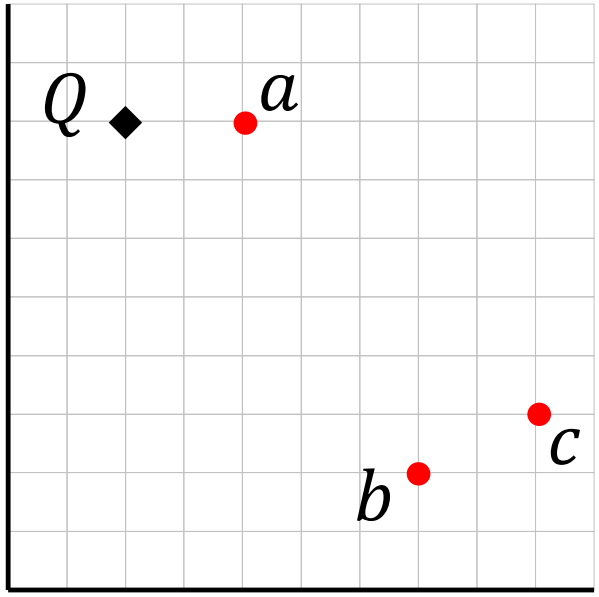
- Аксиома треугольника (неравенство треугольника):

$$\mathrm{Dist}(x, z) \leq \mathrm{Dist}(x, y) + \mathrm{Dist}(y, z)$$

Аксиома положительности избыточна:

$$0 = \mathrm{Dist}(x, x) \leq \mathrm{Dist}(x, y) + \mathrm{Dist}(y, x) = 2 \cdot \mathrm{Dist}(x, y)$$

Почему важно неравенство треугольника



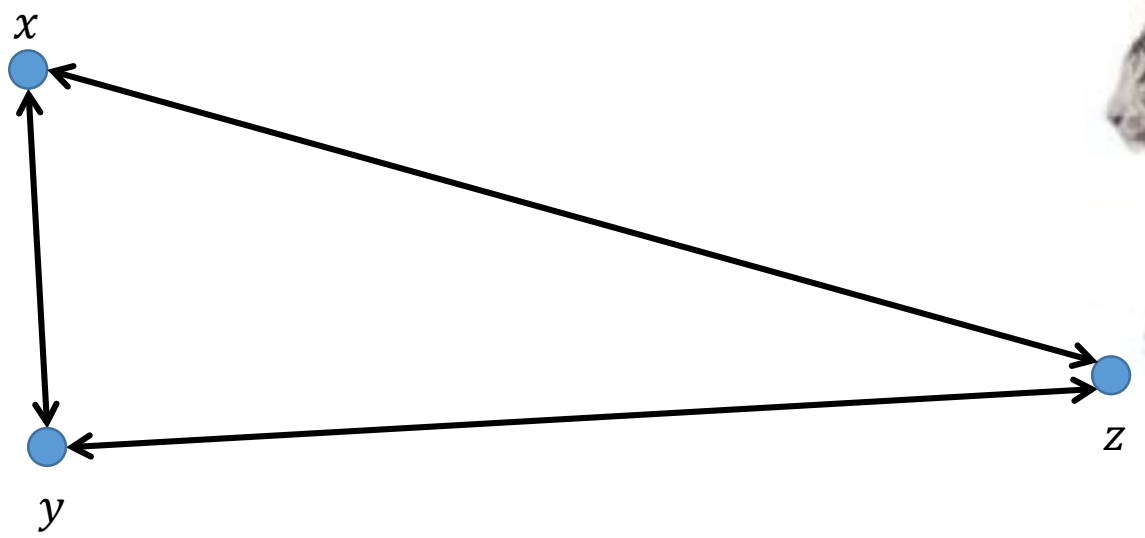
Dist(·,·)	a	b	c
a	0	6.70	7.07
b	6.70	0	2.30
c	7.07	2.30	0

- Поиск объекта, ближайшего к Q:

1. $\mathrm{Dist}(Q, a) = 2$ (***bsf***, ***best-so-far***)
2. $\mathrm{Dist}(Q, b) = 7.81$
3. $\mathrm{Dist}(Q, b) \leq \mathrm{Dist}(Q, c) + \mathrm{Dist}(b, c)$
   $\mathrm{Dist}(Q, b) - \mathrm{Dist}(b, c) \leq \mathrm{Dist}(Q, c)$
   $7.81 - 2.30 \leq \mathrm{Dist}(Q, c)$
   $5.51 \leq \mathrm{Dist}(Q, c)$
   $2 = \mathrm{Dist}(Q, a) < 5.51 \leq \mathrm{Dist}(Q, c)$

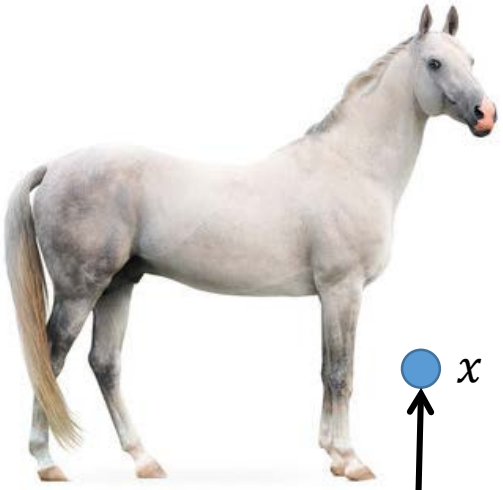
- Объект c можно отбросить, не вычисляя $\mathrm{Dist}(Q, c)$

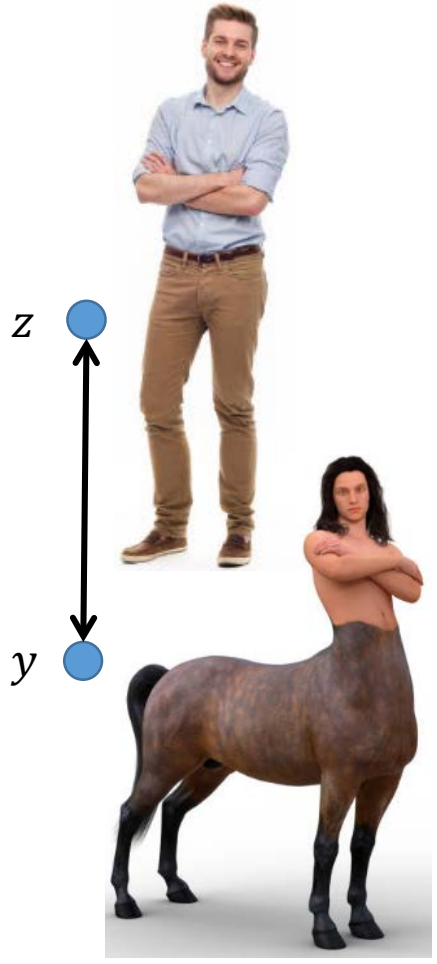
Метрика: неравенство треугольника



$$\forall x, y, z$$
$$\mathrm{Dist}(x, z) \leq \mathrm{Dist}(x, y) + \mathrm{Dist}(y, z)$$

Не-метрика: неравенство треугольника РАБОТАЕТ, НО НЕ ВСЕГДА





$$\neg(\forall x, y, z \quad \mathrm{Dist}(x, z) \leq \mathrm{Dist}(x, y) + \mathrm{Dist}(y, z))$$

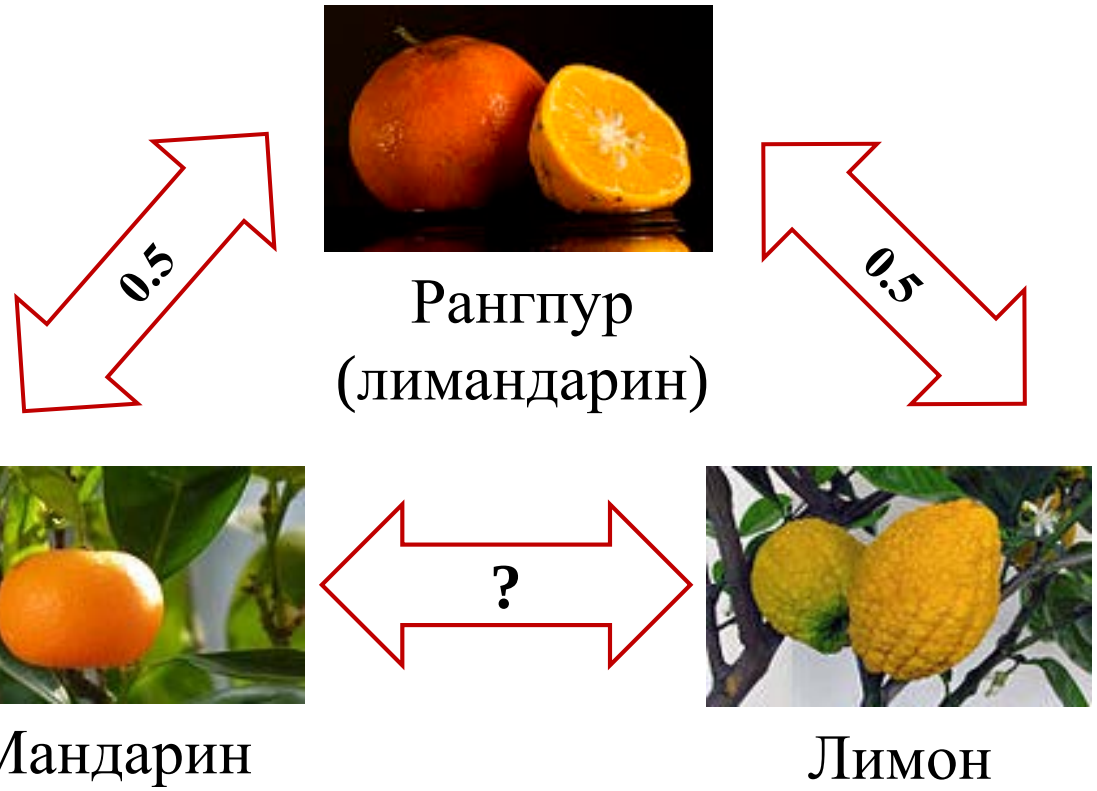
$$\Downarrow$$

$$\exists\, x, y, z \quad \mathrm{Dist}(x, z) > \mathrm{Dist}(x, y) + \mathrm{Dist}(y, z)$$

Метрика и не-метрика без неравенства треугольника

ED, ED^2, ED_{norm} и др.

DTW, MPdist и др.

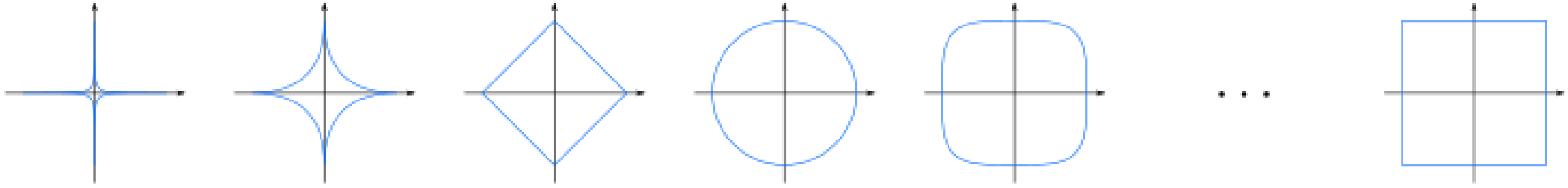


Расстояние Минковского

- $\mathrm{Dist}(Q, C) = \sqrt[p]{\sum_{i=1}^{m} |q_i - c_i|^p}$
- $p < 1$: не метрика (нет аксиомы треугольника)
- $p \geq 1$: метрика
  - $p = 1$: Манхэттенское расстояние
  - $p = 2$: Евклидово расстояние
  - $p = \infty$: расстояние Чебышёва

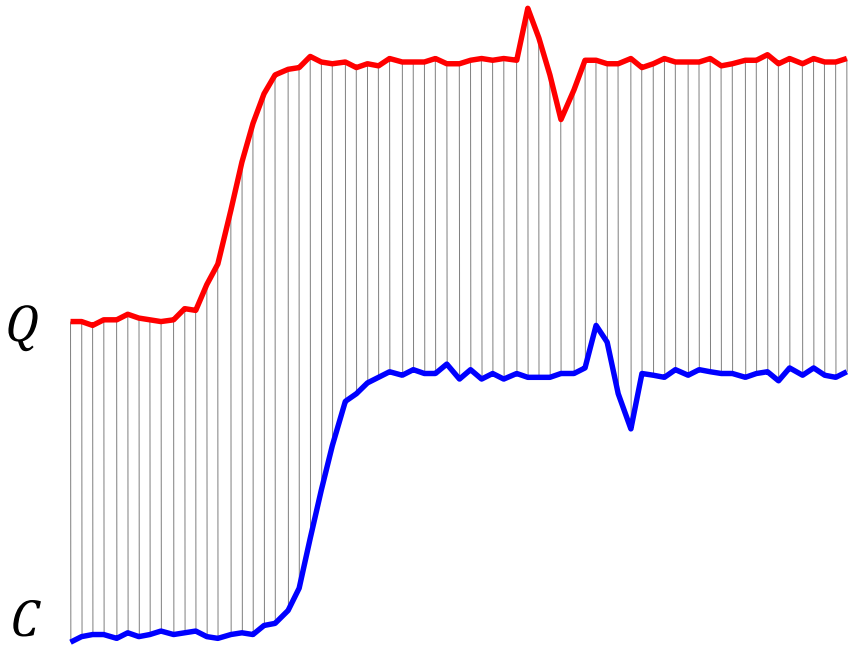
Герман Минковский
1864-1909

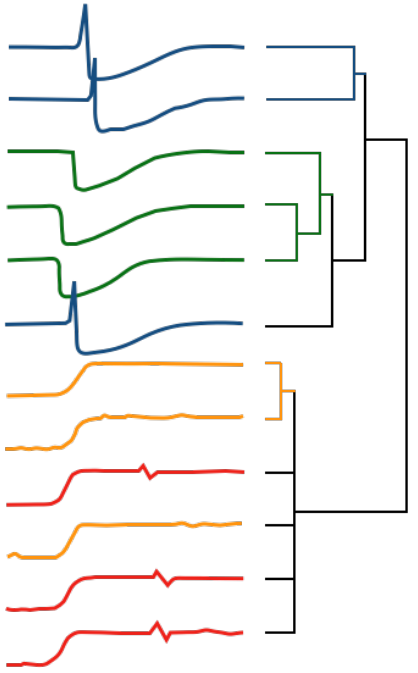
Единичная окружность при различных p



Евклидово расстояние*: $\mathrm{ED}(Q,C)=\sqrt{\sum_{i=1}^{m}(q_i-c_i)^2}$

- Интуитивно понятное расстояние, вычислительная сложность $O(m)$
- $\mathrm{ED}^2(Q,C)=\sum_{i=1}^{m}(q_i-c_i)^2$ – также метрика и вычисляется быстрее
- Сравнение равных по длине рядов по принципу «один к одному», не всегда адекватно учитывает форму рядов

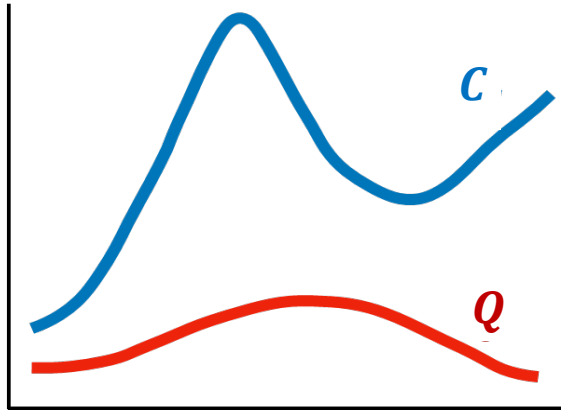


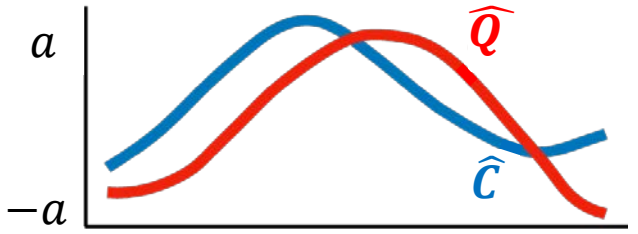


Евклид* 325-265 д.н.э.

Z-нормализованное евклидово расстояние

Нормализация позволяет сравнивать ряды без учета разницы амплитуд





$$\mu_{\hat{Q}}=\mu_{\hat{C}}=0$$

$$\sigma_{\hat{Q}}=\sigma_{\hat{C}}=1$$

$$\mathrm{ED}^2_{\mathrm{norm}}(Q,C)=\mathrm{ED}^2\left(\hat{Q},\hat{C}\right)$$

$$\hat{T}=(\hat{t}_1,\ldots,\hat{t}_m),\qquad \hat{t}_i=\frac{t_i-\mu}{\sigma}$$

$$\mu=\frac{1}{n}\sum_{i=1}^{m}t_i,\qquad \sigma^2=\frac{1}{m}\sum_{i=1}^{m}t_i^2-\mu^2$$

$\mathbf{ED_{norm}}$ связано с корреляцией Пирсона

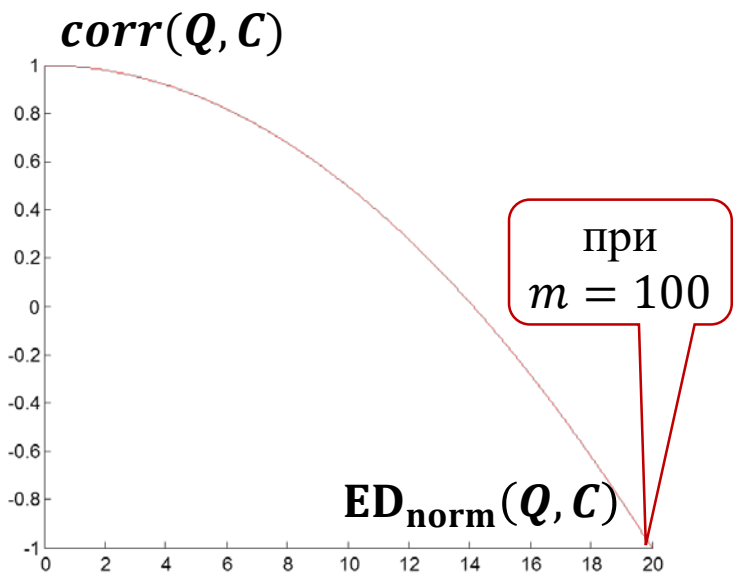
- $corr(Q,C) = \frac{\sum_{i=1}^{m} q_i c_i - m\mu_Q\mu_C}{m\sigma_Q\sigma_C}$ не является метрикой

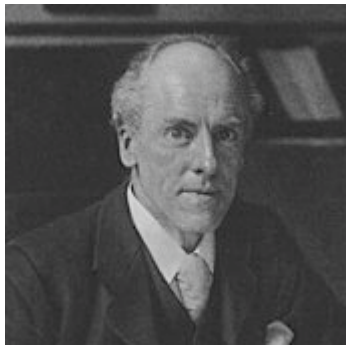
- Связь $\mathrm{ED}_{\mathrm{norm}}$ и $corr^*$

$$\mathrm{ED}_{\mathrm{norm}}(Q,C) = \mathrm{ED}(\hat{Q},\hat{C}) =$$

$$= \sqrt{2m(1 - corr(Q,C))} =$$

$$= \sqrt{2m(1 - \frac{Q \cdot C - m\mu_Q\mu_C}{m\sigma_Q\sigma_C})}$$





Карл Пирсон (Karl Pearson)
1857-1936

- $\mathrm{ED}^2_{\mathrm{norm}}$ также метрика и вычисляется еще быстрее

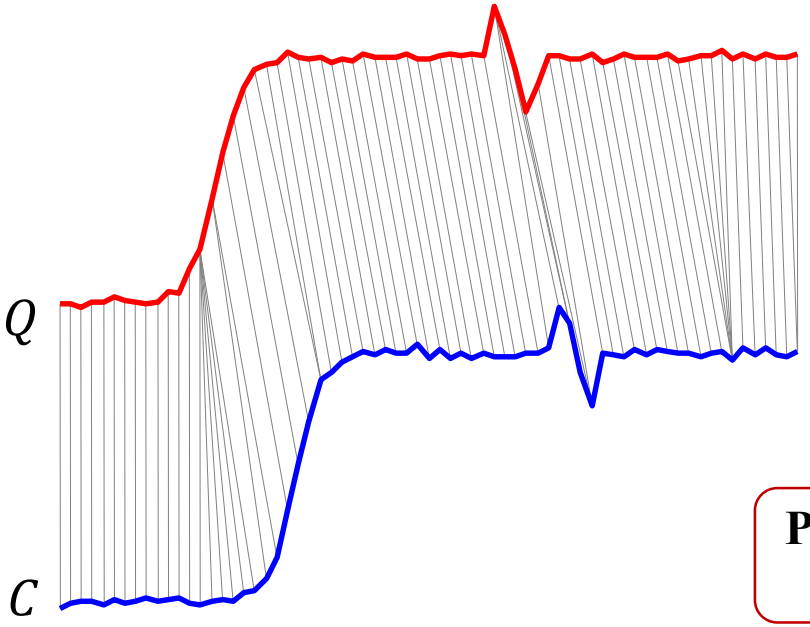
* Mueen A., Nath S., Liu J. Fast approximate correlation for massive time-series data. Proc. of the ACM SIGMOD Int. Conf. on Management of Data, SIGMOD 2010, Indianapolis, Indiana, USA, June 6-10, 2010. P. 171–182. DOI: 10.1145/1807167.1807188

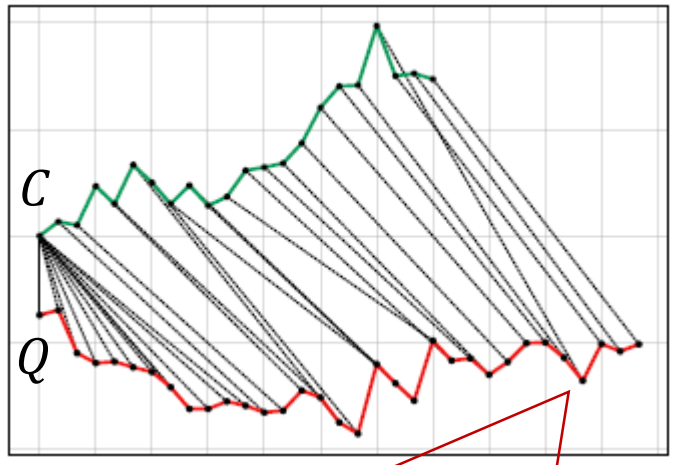
Содержание

- Постановка задачи
- Метрика Евклида
- **Мера DTW**
- Поиск по образцу на основе DTW
- Оптимизации поиска на основе DTW

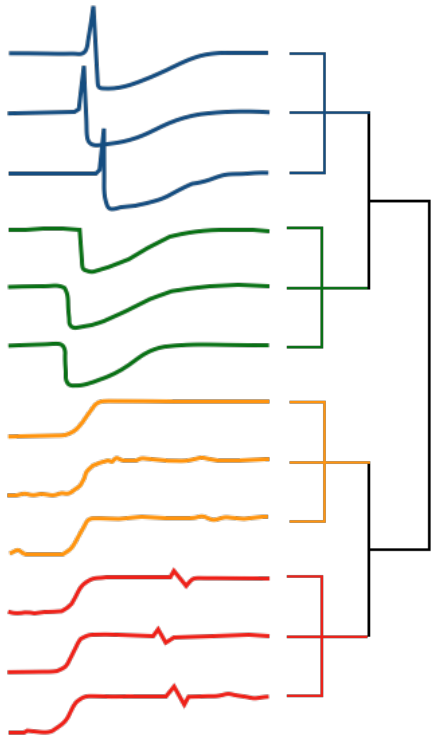
Динамическая трансформация времени: DTW, Dynamic Time Warping*

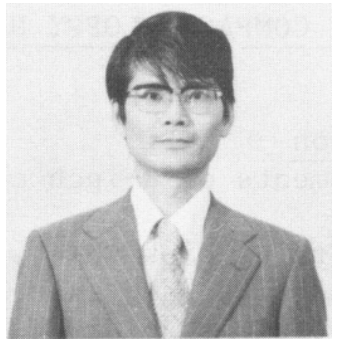
- **Не метрика**: не выполняется неравенство треугольника!
- Вычислительная сложность $O(m^2)$
- Сравнение рядов по принципу «один к много», адекватно учитывает форму рядов





Ряды могут быть разной длины!
Выч. сложность $O(m_1 \cdot m_2)$



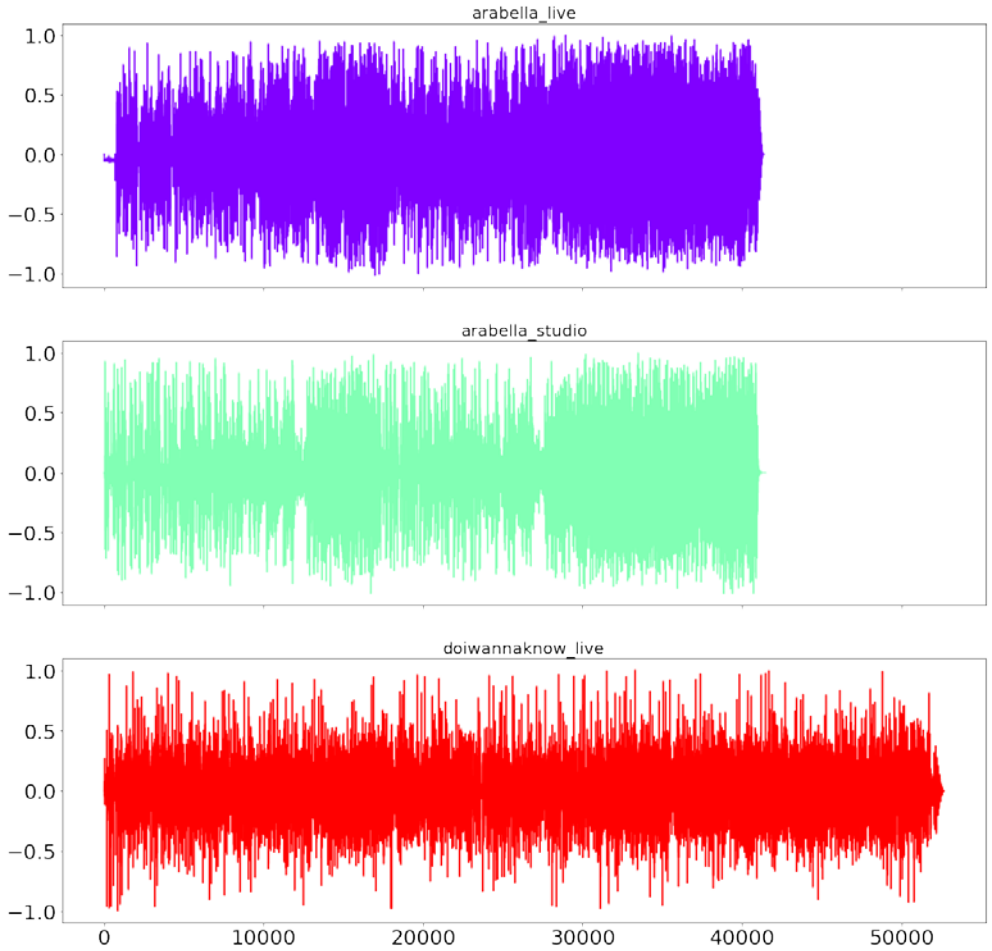


Хироаки Сако
(Hiroaki Sakoe)

Сэйби Чиба
(Sabi Chiba)

* Sakoe H., Chiba S. Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. on Acoustics, Speech, and Signal Processing, 1978. 26(1), 43-49. DOI: 10.1109/TASSP.1978.1163055

Применение DTW: распознавание речи



Arabella (studio)

Arabella (live)

Do I Wanna Know (live)

ED (норм.)	Arabella (live)	Do I Wanna Know (live)
Arabella (studio)	0.043	0.038

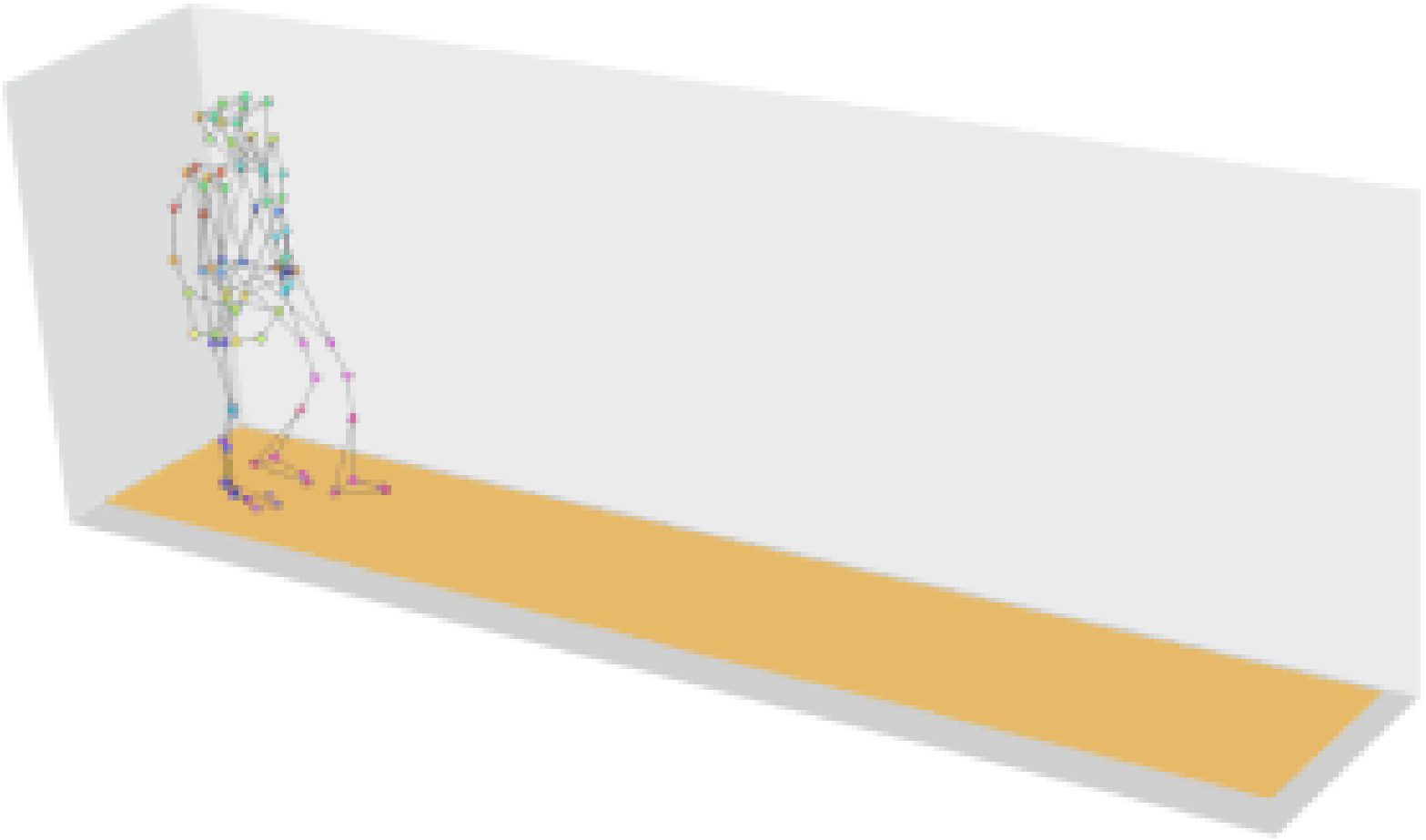
ED не отличает
разные записи разных песен!

DTW отличает
разные записи одной песни!

DTW (норм.)	Arabella (live)	Do I Wanna Know (live)
Arabella (studio)	0.82	1

Mora P. Dynamic Time Warping: Explanation and extensive testing on audio and tabular data. URL

Применение DTW: биометрия



- Две записи походки одного человека с помощью системы захвата движения
- Скорость в попытках разная, но DTW помогает понять, что траектории конечностей имеют большое сходство

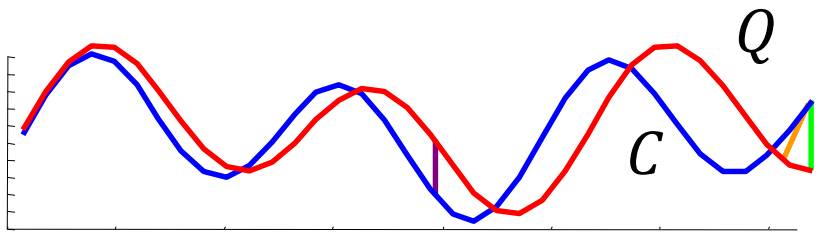
Применение DTW: классификация петроглифов

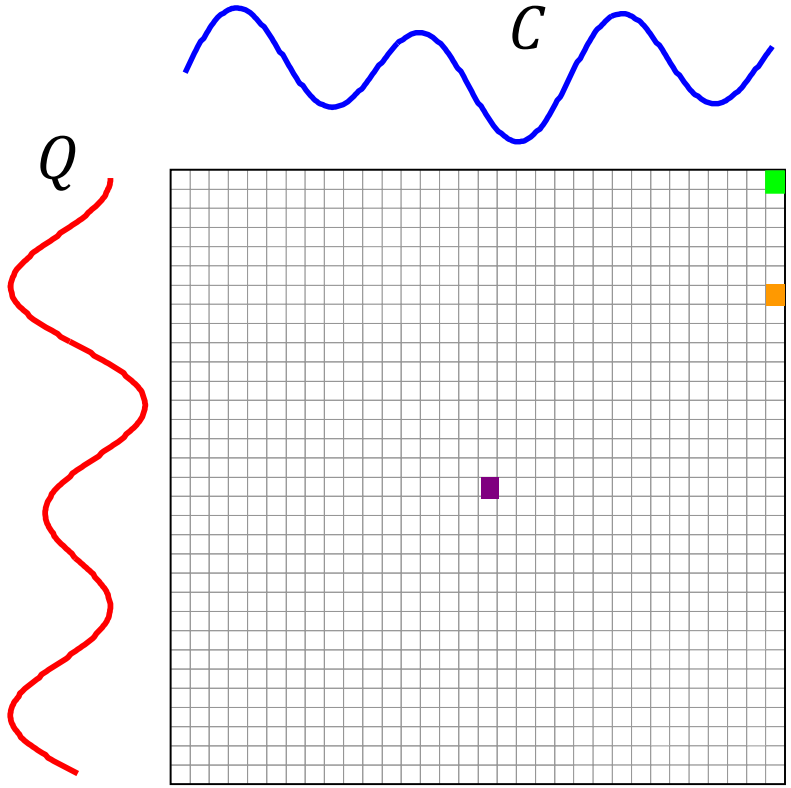
Применение DTW: биология

Вычисление DTW(Q, C): 1. Матрица расстояний

Построим матрицу расстояний $d \in \mathbb{R}^{m \times m}$ между точками Q и C:

$\boldsymbol{d}(i,j) = \text{Dist}(q_i, c_j)$, допустимо $\textbf{Dist}(\cdot,\cdot) = \left(\boldsymbol{q_i} - \boldsymbol{c_j}\right)^{\mathbf{2}}$ или $\text{Dist}(\cdot,\cdot) = |q_i - c_j|$



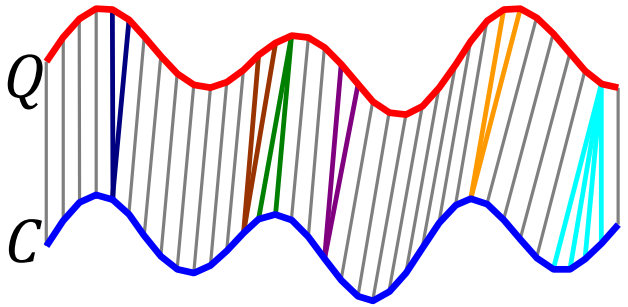


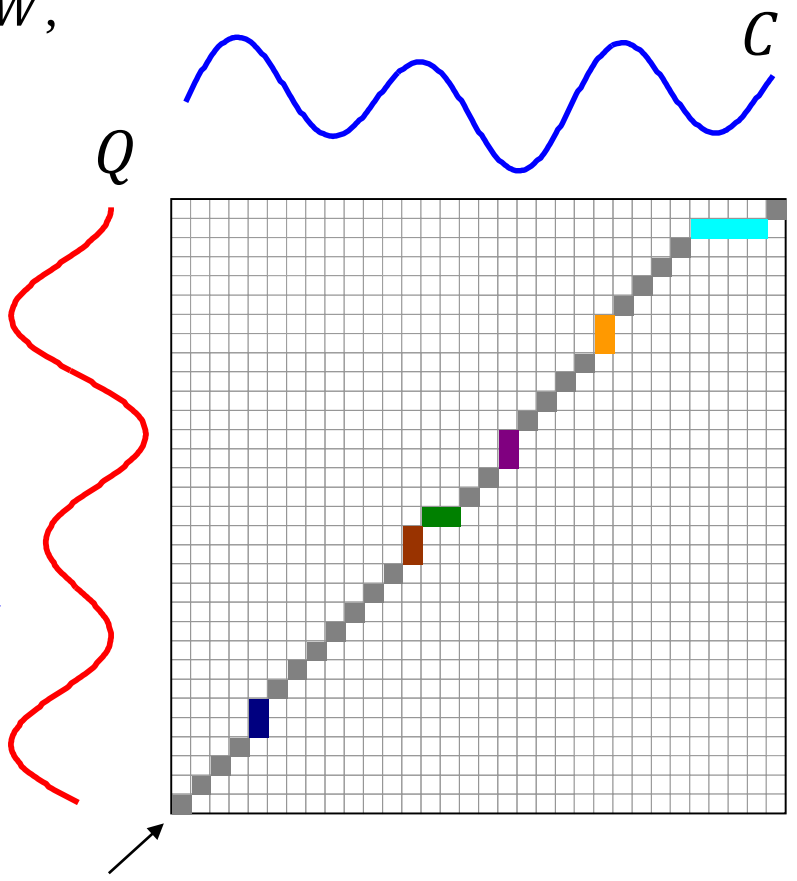
Вычисление DTW(Q, C): 2. Матрица и путь трансформации

Построим матрицу трансформации D и найдем в ней путь трансформации W, который устанавливает соответствие между Q и C, минимизируя общее расстояние между ними:

- путь $W = w_1, \dots, w_K$, длина пути $m \le K < 2m$
- элемент пути $w_k = (i,j)_k$, $d(w_k) = \text{Dist}(q_i, c_j) = d(i,j)$

1. **Полнота пути**
   $w_1 = (1,1), w_K = (m,m)$
2. **Непрерывность пути**
   $\forall\, w_k = (w_i, w_j)$ и $w_{k+1} = (w_{i+1}, w_{j+1})$:
   $w_i - w_{i+1} \le 1$ и $w_j - w_{j+1} \le 1$
3. **Монотонность пути**
   $\forall\, w_k = (w_i, w_j)$ и $w_{k+1} = (w_{i+1}, w_{j+1})$:
   $w_i - w_{i-1} \ge 0$ и $w_j - w_{j-1} \ge 0$





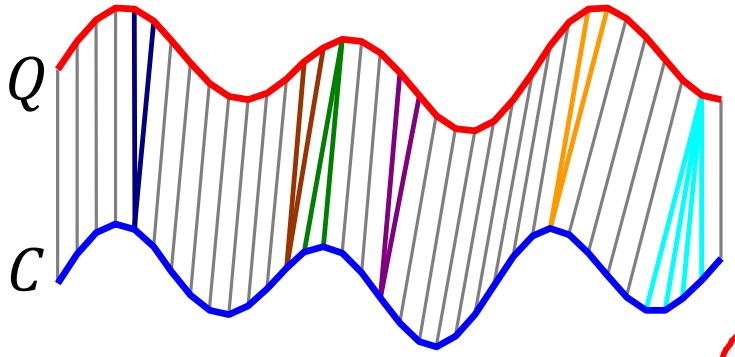
Путь трансформации *W*

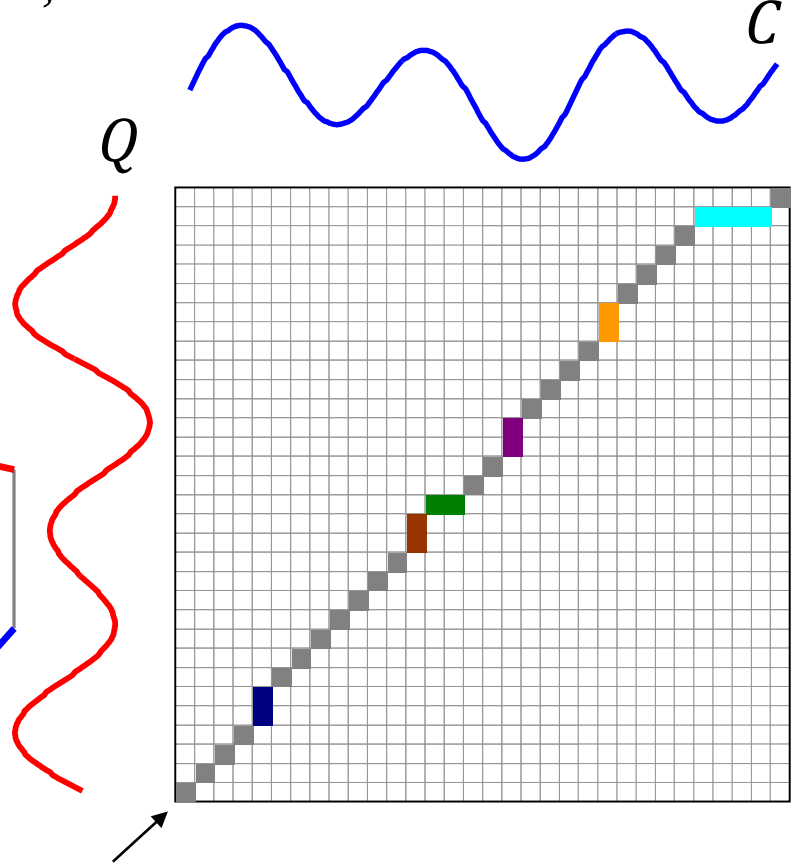
Вычисление DTW(Q, C): 2. Матрица и путь трансформации

Построим матрицу трансформации D и найдем в ней путь трансформации W, который устанавливает соответствие между Q и C, минимизируя общее расстояние между ними:

- путь $W = w_1, \dots, w_K$, длина пути $m \le K < 2m$
- элемент пути $w_k = (i,j)_k$, $d(w_k) = \text{Dist}(q_i, c_j) = d(i,j)$

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



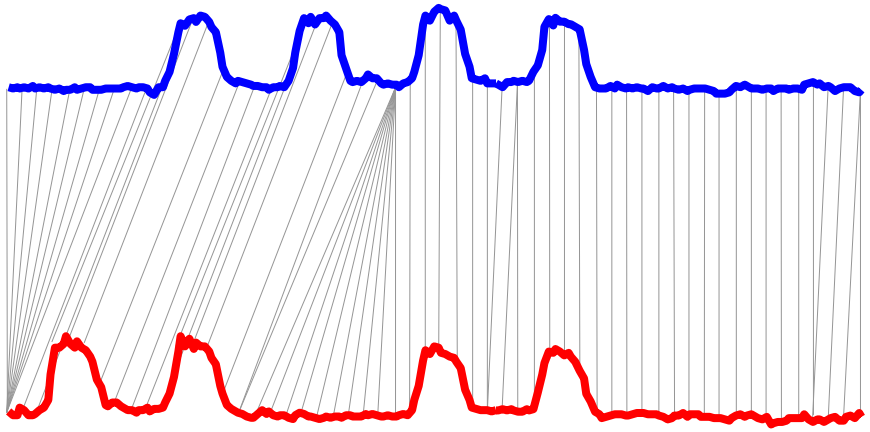


Путь трансформации *W*

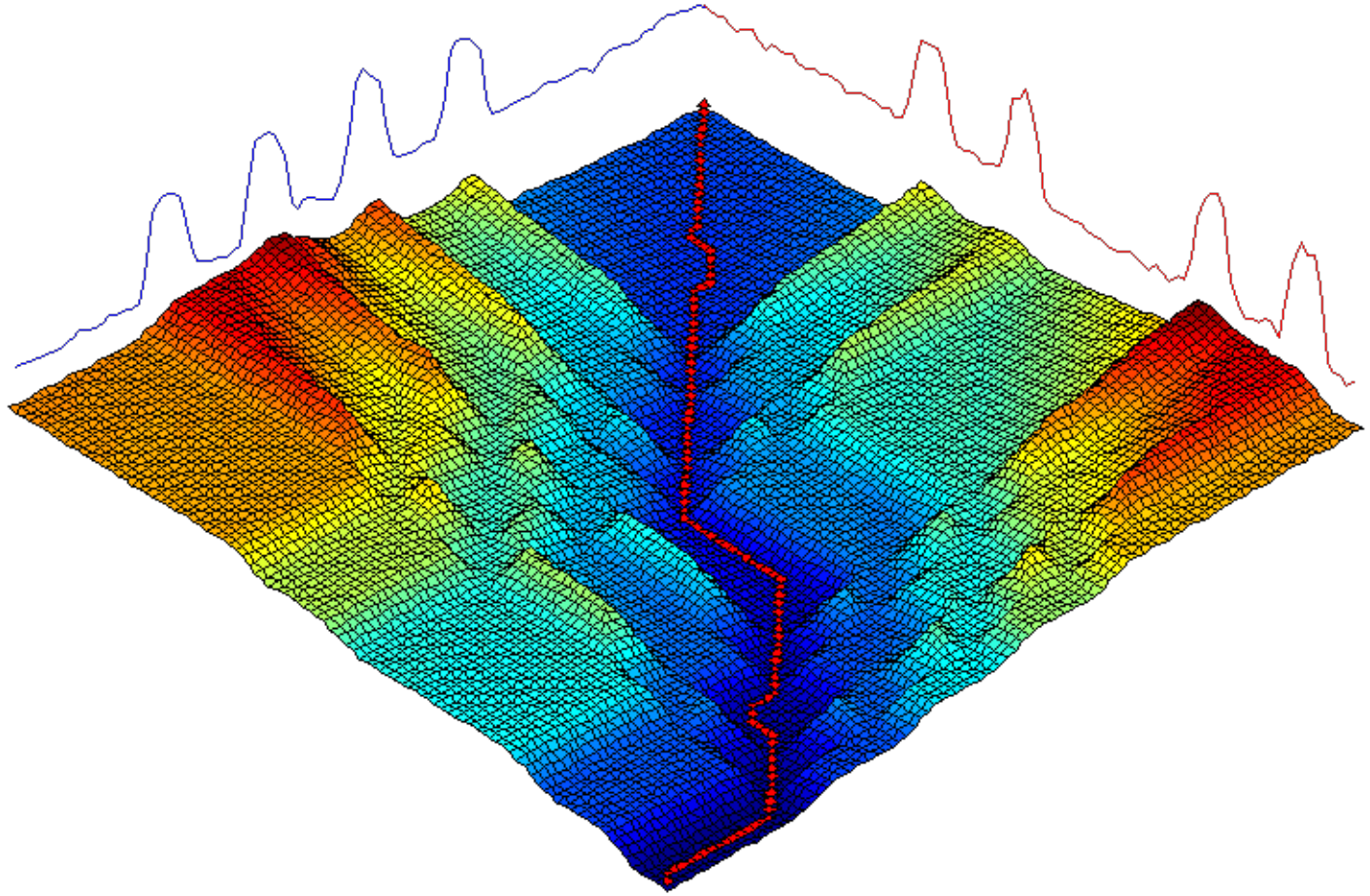
Пример матрицы и пути трансформации

Недельное энергопотребление вычислительного центра (Голландия, 1997)*

C: 4-дневная рабочая неделя,
Понедельник – выходной



Q: 4-дневная рабочая неделя,
Среда – выходной

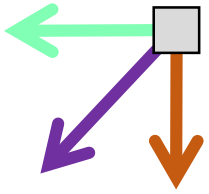


* van Wijk J.J., van Selow R.R. Cluster and calendar based visualization of time series data. INFOVIS 1999: 4-9. DOI: 10.1109/INFVIS.1999.801851

Вычисление DTW(Q, C): матрица и путь трансформации

Матрица трансформации

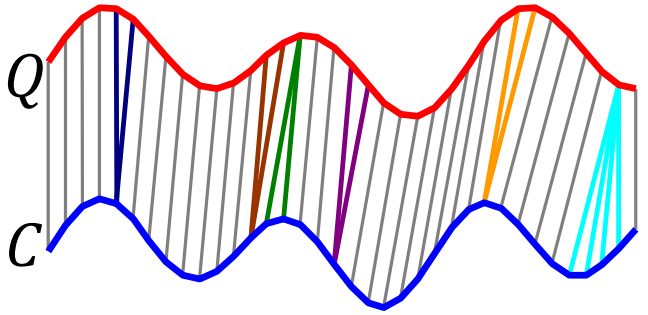
$D \in \mathbb{R}^{(m+1)\times(m+1)}$

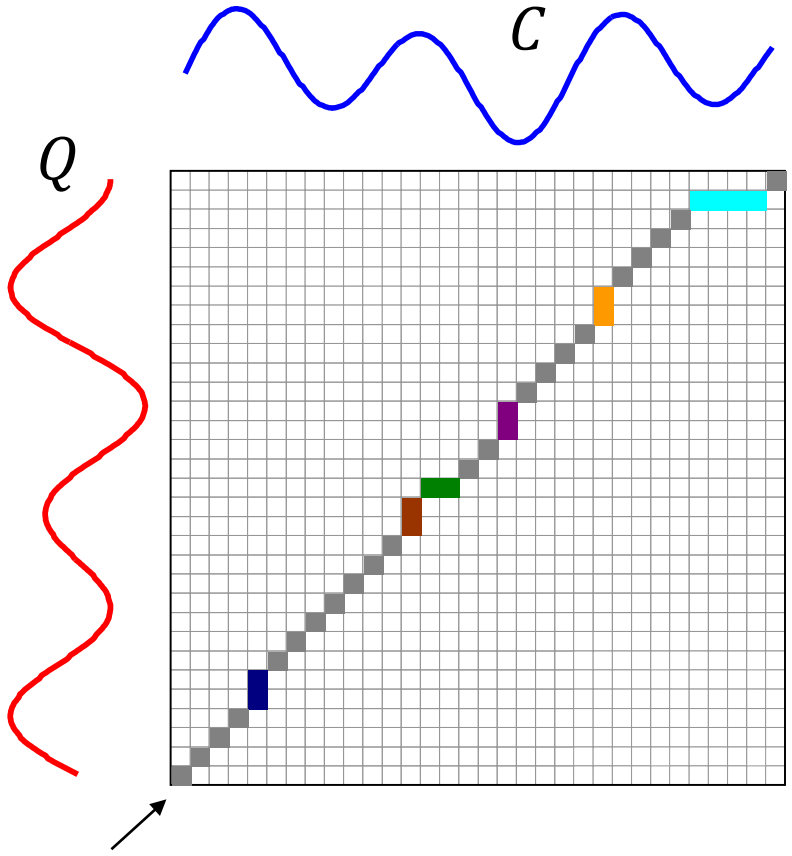


$$\mathrm{DTW}(Q,C) = D(m,m)$$

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

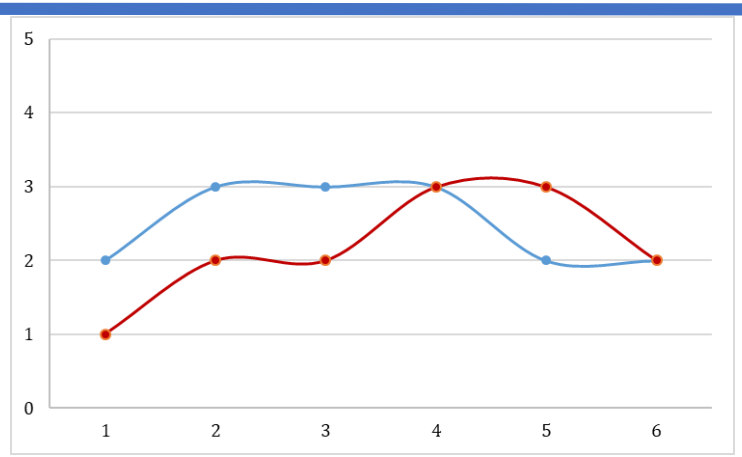
$$D(0,0) = 0, D(i,0) = D(0,j) = \infty$$





Путь трансформации w

Пример вычисления DTW



$$DTW(Q, C) = D(m, m)$$

$$D(i,j) = (q_i - c_j)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

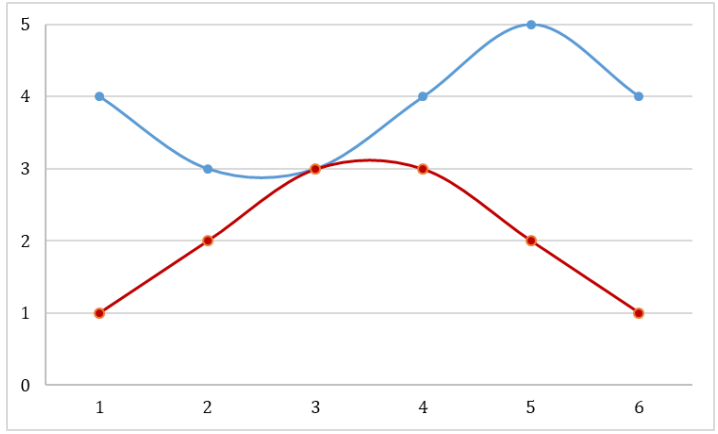
$$D(0,0) = 0, \qquad D(i,0) = D(0,j) = +\infty$$

$$C = (2,3,3,3,2,2), Q = (1,2,2,3,3,2)$$

d	2	3	3	3	2	2	
2	0	1	1	1	0	0	6
3	1	0	0	0	1	1	5
3	1	0	0	0	1	1	4
2	0	1	1	1	0	0	3
2	0	1	1	1	0	0	2
1	1	4	4	4	1	1	1
$\rightarrow j$	1	2	3	4	5	6	$\uparrow i$

D	2	3	3	3	2	2	
2	+∞	3	2	2	2	1	**1**	6
3	+∞	3	1	1	1	2	3	5
3	+∞	2	1	1	1	2	3	4
2	+∞	1	2	3	4	4	4	3
2	+∞	1	2	3	4	4	4	2
1	+∞	1	5	9	13	14	15	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
$\rightarrow j$	0	1	2	3	4	5	6	$\uparrow i$

Пример вычисления DTW



$$\mathrm{DTW}(Q, C) = D(m, m)$$

$$D(i,j) = (q_i - c_j)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, \qquad D(i,0) = D(0,j) = +\infty$$

$$C = (4,3,3,4,5,4), Q = (1,2,3,3,2,1)$$

d	4	3	3	4	5	4	
1	9	4	4	9	16	9	6
2	4	1	1	4	9	4	5
3	1	0	0	1	4	1	4
3	1	0	0	1	4	1	3
2	4	1	1	4	9	4	2
1	9	4	4	9	16	9	1
$\rightarrow j$	1	2	3	4	5	6	$\uparrow i$

D	4	3	3	4	5	4	
1	+∞	28	15	15	20	30	**28**	6
2	+∞	19	11	11	14	20	19	5
3	+∞	15	10	10	11	15	16	4
3	+∞	14	10	10	11	15	16	3
2	+∞	13	10	11	15	24	28	2
1	+∞	9	13	17	26	42	51	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
$\rightarrow j$	0	1	2	3	4	5	6	$\uparrow i$

Вычисление DTW

Algorithm DTW $(Q, C \in \mathbb{R}^m)$

$D \in \mathbb{R}^{(1+m)\times(1+m)}, d \in \mathbb{R}^{m\times m}$

$D := \overline{+\infty}$; $D(0,0) := 0$

for $i := 1$ **to** m

  for $j := 1$ to m

    $d(i,j) := \mathrm{Dist}(q_i, c_j)$

    $D(i,j) := d(i,j) + \min\{D(i-1,j), D(i,j-1), D(i-1,j-1)\}$

return $D(m,m)$

Вычисление DTW: сложность (игра не стоит свеч?)

Algorithm DTW $(Q, C \in \mathbb{R}^m)$

$D \in \mathbb{R}^{(1+m)\times(1+m)}, d \in \mathbb{R}^{m\times m}$

$D := \overline{+\infty}; D(0,0) := 0$

for $i := 1$ **to** m

  for $j := 1$ to m

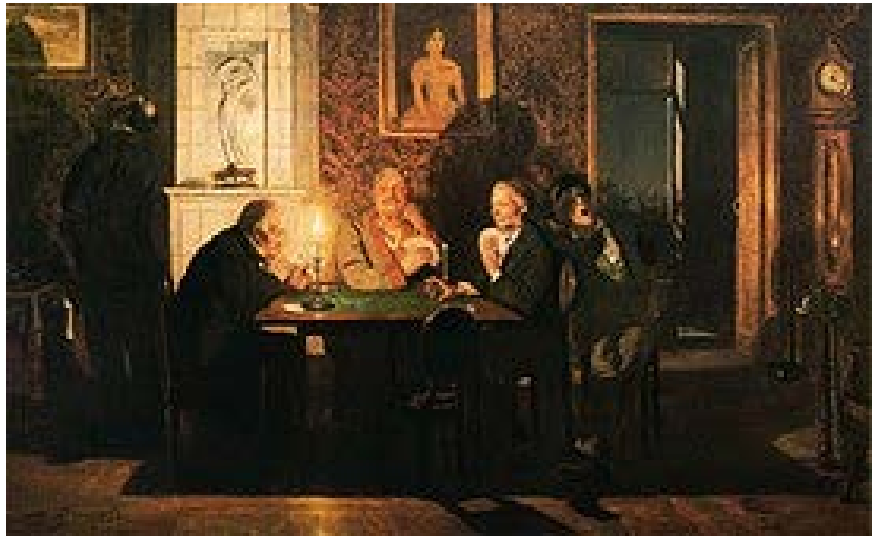
    $d(i,j) := \mathrm{Dist}(q_i, c_j)$

    $D(i,j) := d(i,j) + \min\{D(i-1,j), D(i,j-1), D(i-1,j-1)\}$

return $D(m,m)$

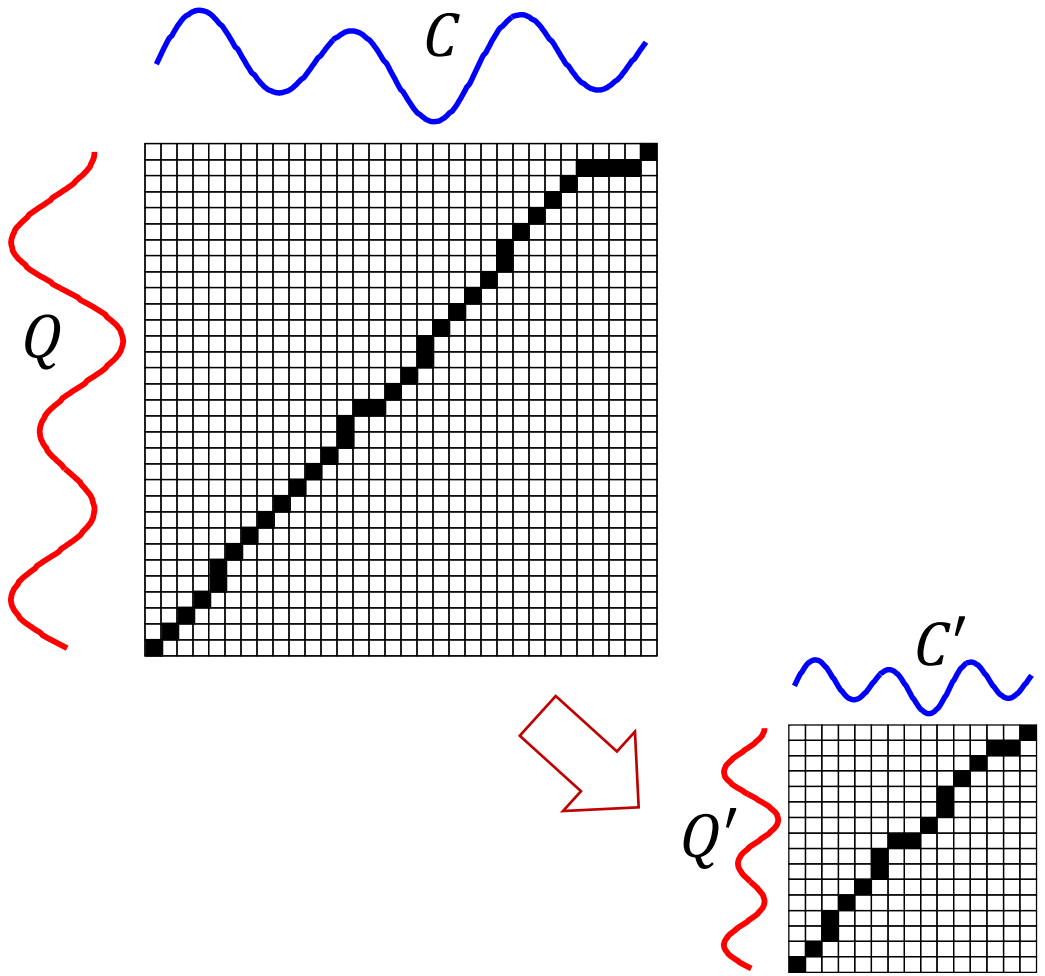
Вычислительная сложность $\boldsymbol{O(m^2)}$

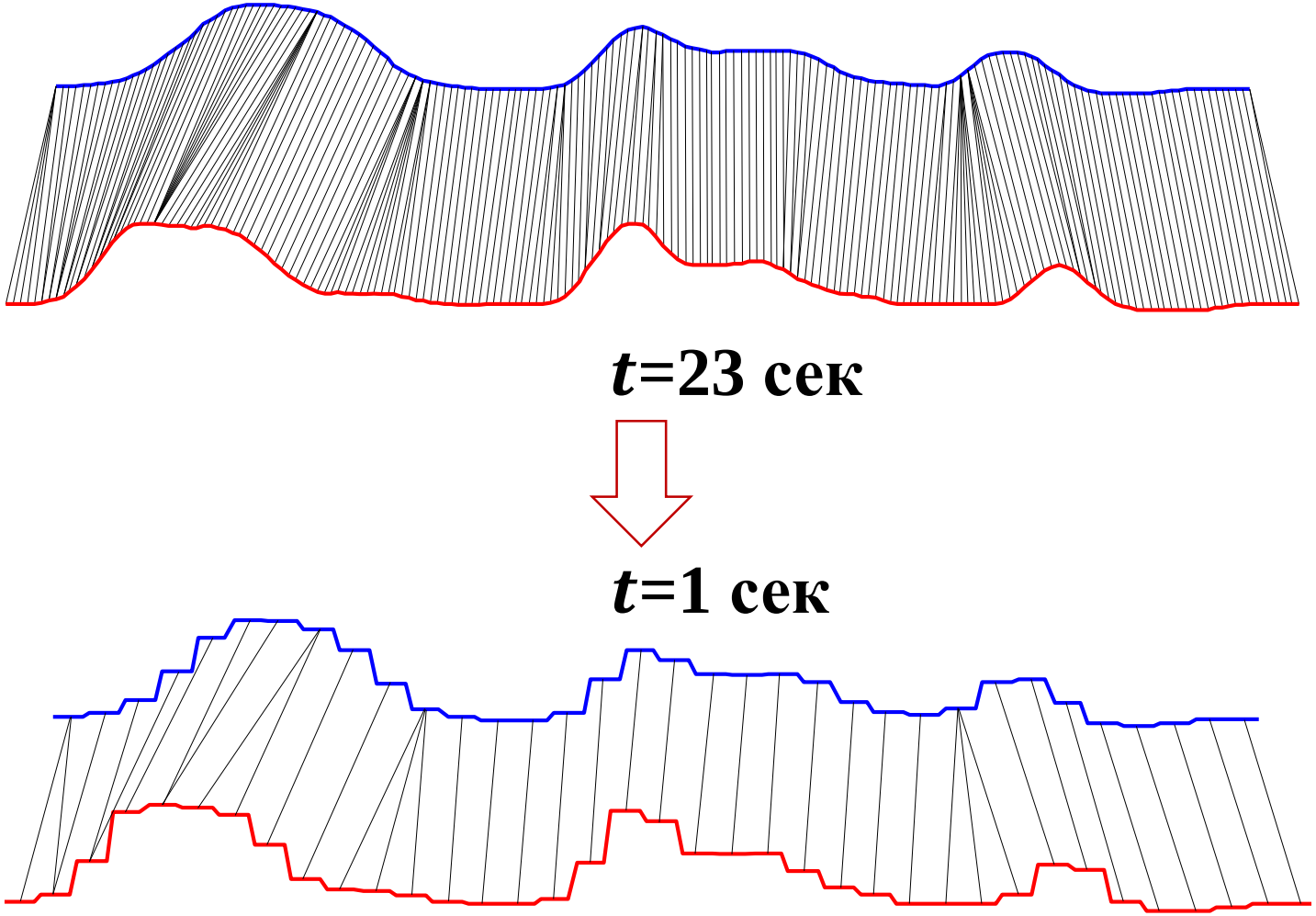
Пространственная сложность $\boldsymbol{O(m^2)}$



В.М. Васнецов. «Преферанс»

Снижение сложности DTW: сжатие ряда (downsampling)

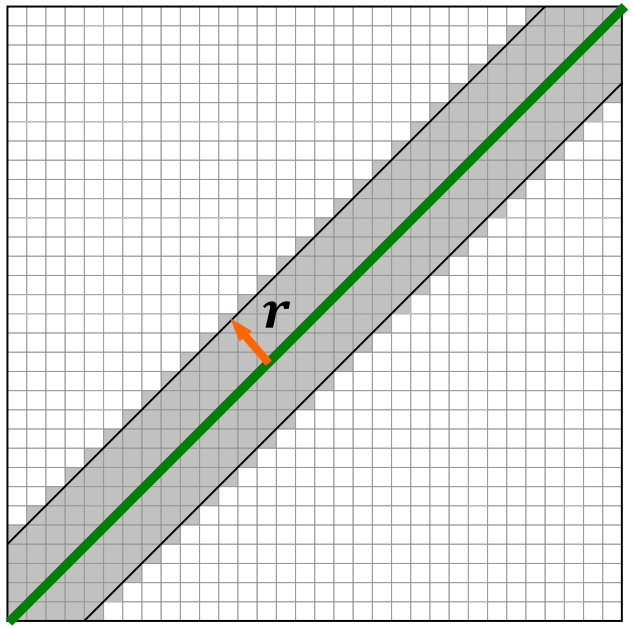




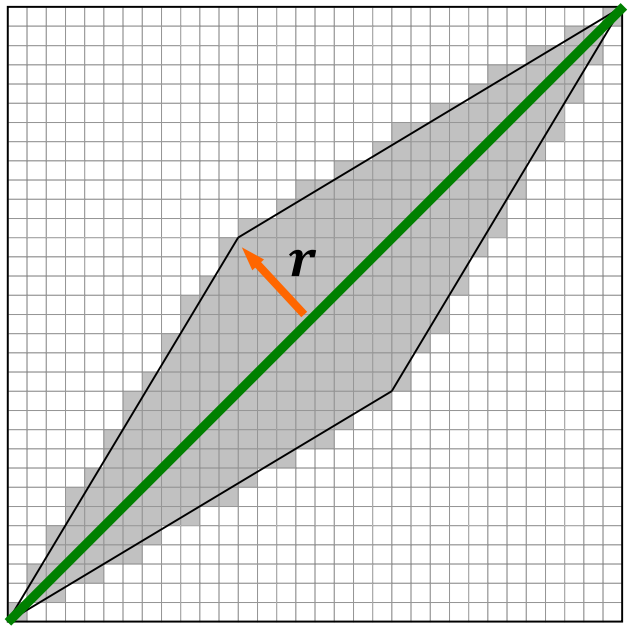
Снижение сложности DTW: ограничение пути трансформации

- Путь не должен отклоняться от диагонали более чем на $\boldsymbol{r}$
- Сложность: $\boldsymbol{O(rm)}$

Полоса Сако–Чиба



Параллелограмм Итакуры



$$\mathbf{DTW}(Q, C) = D(m, m)$$

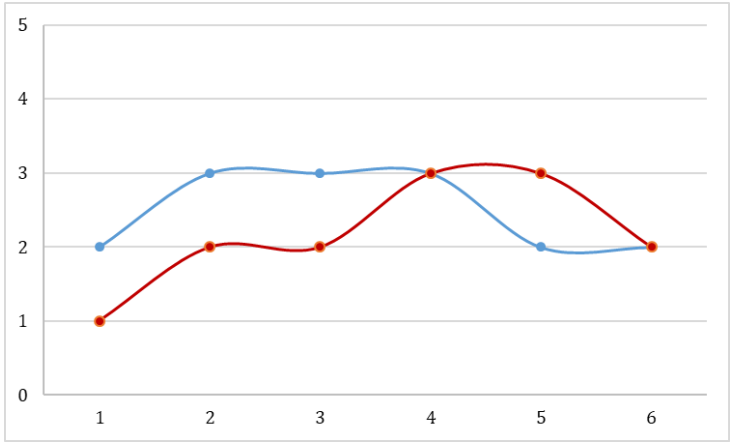
$$D(i,j) = (q_i - c_i)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, D(i,0) = D(0,j) = +\infty;\ 1 \le i,j \le m;$$

$$\mathbf{0} \le \boldsymbol{r} \le \boldsymbol{m} - \mathbf{1}, j - \boldsymbol{r} \le i \le j + \boldsymbol{r}$$

$$D(i,j) = +\infty, \qquad j + \boldsymbol{r} < i < j - \boldsymbol{r}$$

Пример вычисления DTW с ограничением



$$\mathbf{DTW}(Q, C) = D(m, m)$$

$$D(i,j) = (q_i - c_i)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, D(i,0) = D(0,j) = +\infty;\ 1 \le i,j \le m;$$

$$\mathbf{0} \le r \le \boldsymbol{m} - \mathbf{1}, j - r \le i \le j + r$$

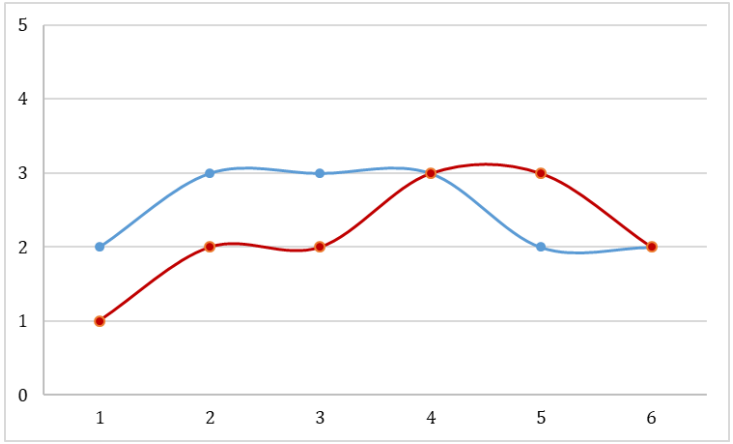
$$D(i,j) = +\infty, \qquad j + r < i < j - r$$

$$C = (2,3,3,3,2,2), Q = (1,2,2,3,3,2)$$

$r = 2$

		2	3	3	3	2	2	
2	+∞	+∞	+∞	+∞	2	1	1	6
3	+∞	+∞	+∞	1	1	2	3	5
3	+∞	+∞	1	1	1	2	3	4
2	+∞	1	2	3	4	4	+∞	3
2	+∞	1	2	3	4	+∞	+∞	2
1	+∞	1	5	9	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ **j**	0	1	2	3	4	5	6	↑ **i**

Пример вычисления DTW с ограничением



$$\mathbf{DTW}(Q, C) = D(m, m)$$

$$D(i,j) = (q_i - c_i)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, D(i,0) = D(0,j) = +\infty;\ 1 \le i,j \le m;$$

$$\mathbf{0} \le r \le \boldsymbol{m} - \mathbf{1}, j - r \le i \le j + r$$

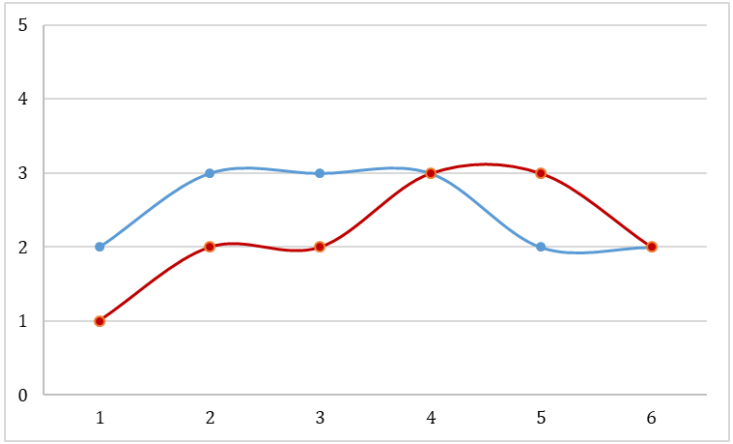
$$D(i,j) = +\infty, \qquad j + r < i < j - r$$

$$C = (2,3,3,3,2,2), Q = (1,2,2,3,3,2)$$

$r = 1$

		2	3	3	3	2	2	
2	+∞	+∞	+∞	+∞	+∞	2	2	6
3	+∞	+∞	+∞	+∞	2	3	4	5
3	+∞	+∞	+∞	2	2	3	+∞	4
2	+∞	+∞	2	3	4	+∞	+∞	3
2	+∞	1	2	3	+∞	+∞	+∞	2
1	+∞	1	5	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ **j**	0	1	2	3	4	5	6	↑ **i**

Пример вычисления DTW с ограничением



$$\mathbf{DTW}(Q, C) = D(n, n)$$

$$D(i,j) = (q_i - c_i)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, D(i,0) = D(0,j) = +\infty;\ 1 \le i,j \le m;$$

$$\mathbf{0} \le r \le \boldsymbol{m} - \mathbf{1}, j - r \le i \le j + r$$

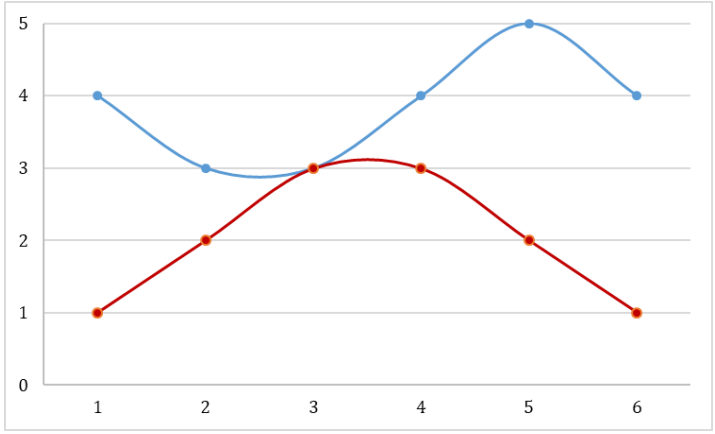
$$D(i,j) = +\infty, \qquad j + r < i < j - r$$

$$C = (2,3,3,3,2,2), Q = (1,2,2,3,3,2)$$

$r = 0$

		2	3	3	3	2	2	
2	+∞	+∞	+∞	+∞	+∞	+∞	1	6
3	+∞	+∞	+∞	+∞	+∞	1	+∞	5
3	+∞	+∞	+∞	+∞	1	+∞	+∞	4
2	+∞	+∞	+∞	1	+∞	+∞	+∞	3
2	+∞	+∞	1	+∞	+∞	+∞	+∞	2
1	+∞	1	+∞	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ *j*	0	1	2	3	4	5	6	↑ *i*

Пример вычисления DTW с ограничением



$$\mathbf{DTW}(Q, C) = D(m, m)$$

$$D(i,j) = (q_i - c_i)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, D(i,0) = D(0,j) = +\infty;\ 1 \le i,j \le m;$$

$$\mathbf{0} \le r \le \boldsymbol{m} - \mathbf{1}, j - r \le i \le j + r$$

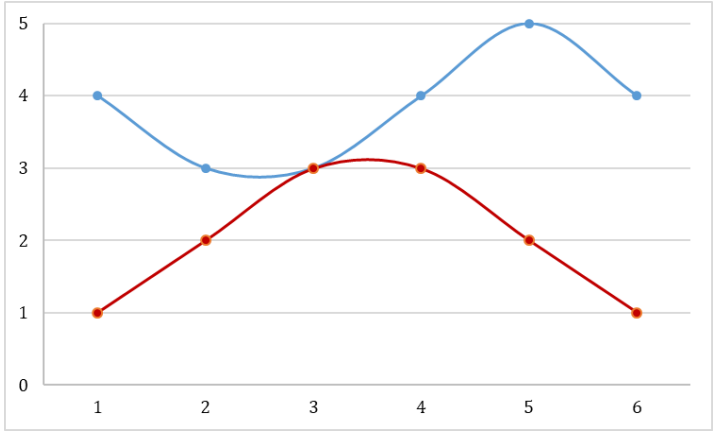
$$D(i,j) = +\infty, \qquad j + r < i < j - r$$

$$C = (4,3,3,4,5,4), Q = (1,2,3,3,2,1)$$

$r = 2$

		4	3	3	4	5	4	
1	+∞	+∞	+∞	+∞	20	30	**28**	6
2	+∞	+∞	+∞	11	14	20	19	5
3	+∞	+∞	10	10	11	15	16	4
3	+∞	14	10	10	11	15	+∞	3
2	+∞	13	10	11	15	+∞	+∞	2
1	+∞	9	13	17	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ *j*	0	1	2	3	4	5	6	↑ *i*

Пример вычисления DTW с ограничением



$$\mathbf{DTW}(Q, C) = D(m, m)$$

$$D(i,j) = (q_i - c_i)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, D(i,0) = D(0,j) = +\infty;\ 1 \le i,j \le m;$$

$$\mathbf{0} \le r \le \mathbf{m} - \mathbf{1}, j - r \le i \le j + r$$

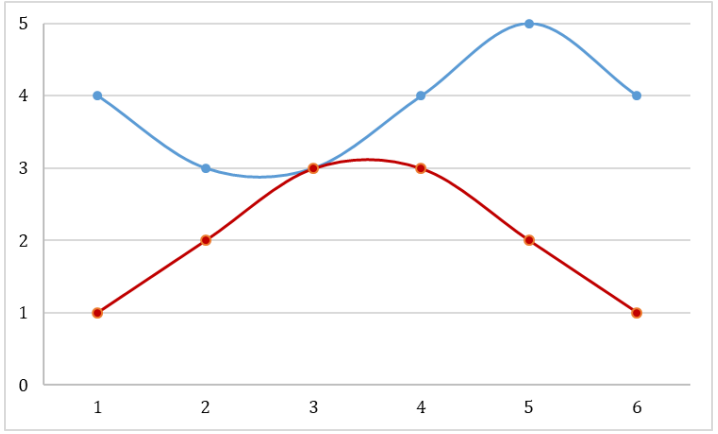
$$D(i,j) = +\infty, \qquad j + r < i < j - r$$

$$C = (4,3,3,4,5,4), Q = (1,2,3,3,2,1)$$

$r = 1$

		4	3	3	4	5	4	
1	+∞	+∞	+∞	+∞	+∞	30	**28**	6
2	+∞	+∞	+∞	+∞	14	20	19	5
3	+∞	+∞	+∞	10	11	15	+∞	4
3	+∞	+∞	10	10	11	+∞	+∞	3
2	+∞	13	10	11	+∞	+∞	+∞	2
1	+∞	9	13	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ j	0	1	2	3	4	5	6	↑ i

Пример вычисления DTW с ограничением



$$\mathbf{DTW}(Q, C) = D(m, m)$$

$$D(i,j) = (q_i - c_i)^2 + \min \begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1) \end{cases}$$

$$D(0,0) = 0, D(i,0) = D(0,j) = +\infty;\ 1 \le i,j \le m;$$

$$\mathbf{0} \le r \le \boldsymbol{m} - \mathbf{1}, j - r \le i \le j + r$$

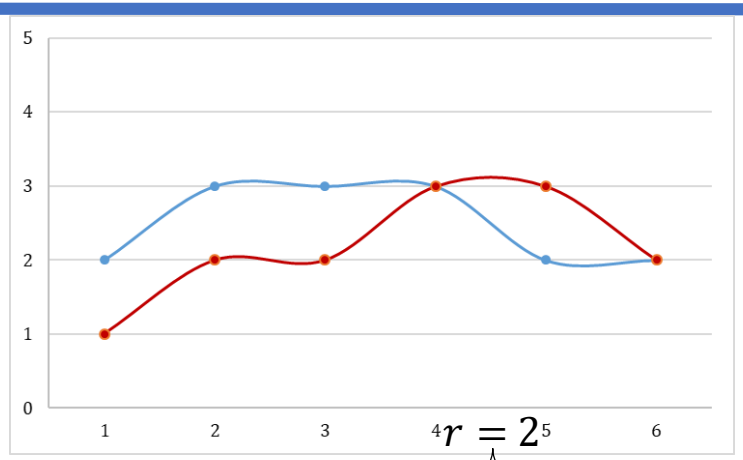
$$D(i,j) = +\infty, \qquad j + r < i < j - r$$

$$C = (4,3,3,4,5,4), Q = (1,2,3,3,2,1)$$

$r = 0$

		4	3	3	4	5	4	
1	+∞	+∞	+∞	+∞	+∞	+∞	**29**	6
2	+∞	+∞	+∞	+∞	+∞	20	+∞	5
3	+∞	+∞	+∞	+∞	11	+∞	+∞	4
3	+∞	+∞	+∞	10	+∞	+∞	+∞	3
2	+∞	+∞	10	+∞	+∞	+∞	+∞	2
1	+∞	9	+∞	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ *j*	0	1	2	3	4	5	6	↑ *i*

Сравнение результатов вычисления DTW



		2	3	3	3	2	2	
2	+∞	3	2	2	2	1	1	6
3	+∞	3	1	1	1	2	3	5
3	+∞	2	1	1	1	2	3	4
2	+∞	1	2	3	4	4	4	3
2	+∞	1	2	3	4	4	4	2
1	+∞	1	5	9	13	14	15	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ j		0	1	2	3	4	5	6 ↑ i

$r = 2$

		2	3	3	3	2	2	
2	+∞	+∞	+∞	+∞	2	1	1	6
3	+∞	+∞	+∞	1	1	2	3	5
3	+∞	+∞	1	1	1	2	3	4
2	+∞	1	2	3	4	4	+∞	3
2	+∞	1	2	3	4	+∞	+∞	2
1	+∞	1	5	9	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ j		0	1	2	3	4	5	6 ↑ i

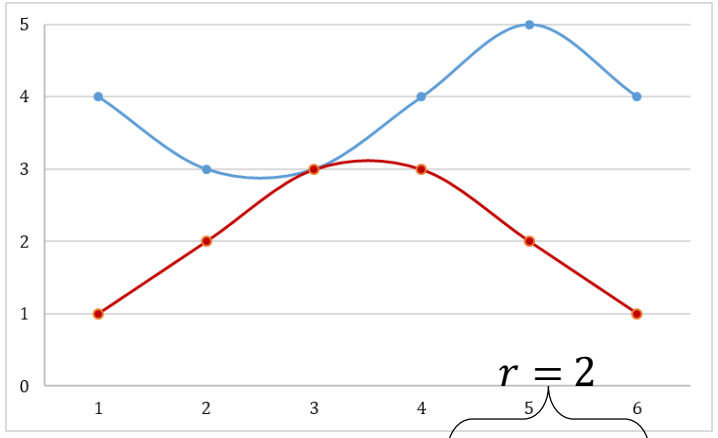
$r = 1$

		2	3	3	3	2	2	
2	+∞	+∞	+∞	+∞	+∞	2	2	6
3	+∞	+∞	+∞	+∞	2	3	4	5
3	+∞	+∞	+∞	2	2	3	+∞	4
2	+∞	+∞	2	3	4	+∞	+∞	3
2	+∞	1	2	3	+∞	+∞	+∞	2
1	+∞	1	5	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ j		0	1	2	3	4	5	6 ↑ i

$r = 0$

		2	3	3	3	2	2	
2	+∞	+∞	+∞	+∞	+∞	+∞	1	6
3	+∞	+∞	+∞	+∞	+∞	1	+∞	5
3	+∞	+∞	+∞	+∞	1	+∞	+∞	4
2	+∞	+∞	+∞	1	+∞	+∞	+∞	3
2	+∞	+∞	1	+∞	+∞	+∞	+∞	2
1	+∞	1	+∞	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
→ j		0	1	2	3	4	5	6 ↑ i

Сравнение результатов вычисления DTW



		4	3	3	4	5	4	
1	+∞	+∞	+∞	+∞	20	30	**28**	6
2	+∞	+∞	+∞	11	14	20	19	5
3	+∞	+∞	10	10	11	15	16	4
3	+∞	14	10	10	11	15	+∞	3
2	+∞	13	10	11	15	+∞	+∞	2
1	+∞	9	13	17	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
$\rightarrow j$		0	1	2	3	4	5	6 $\uparrow i$

$r = 1$

		4	3	3	4	5	4	
1	+∞	+∞	+∞	+∞	+∞	30	**28**	6
2	+∞	+∞	+∞	+∞	14	20	19	5
3	+∞	+∞	+∞	10	11	15	+∞	4
3	+∞	+∞	10	10	11	+∞	+∞	3
2	+∞	13	10	11	+∞	+∞	+∞	2
1	+∞	9	13	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
$\rightarrow j$		0	1	2	3	4	5	6 $\uparrow i$

$r = 0$

		4	3	3	4	5	4	
1	+∞	+∞	+∞	+∞	+∞	+∞	**29**	6
2	+∞	+∞	+∞	+∞	+∞	20	+∞	5
3	+∞	+∞	+∞	+∞	11	+∞	+∞	4
3	+∞	+∞	+∞	10	+∞	+∞	+∞	3
2	+∞	+∞	10	+∞	+∞	+∞	+∞	2
1	+∞	9	+∞	+∞	+∞	+∞	+∞	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
$\rightarrow j$		0	1	2	3	4	5	6 $\uparrow i$

		4	3	3	4	5	4	
1	+∞	28	15	15	20	30	**28**	6
2	+∞	19	11	11	14	20	19	5
3	+∞	15	10	10	11	15	16	4
3	+∞	14	10	10	11	15	16	3
2	+∞	13	10	11	15	24	28	2
1	+∞	9	13	17	26	42	51	1
	0	+∞	+∞	+∞	+∞	+∞	+∞	0
$\rightarrow j$		0	1	2	3	4	5	6 $\uparrow i$

Вычисление DTW с ограничением

Algorithm DTW $(Q, C \in \mathbb{R}^m$, **int** $r)$

$D \in \mathbb{R}^{(1+m)\times(1+m)}, d \in \mathbb{R}^{m\times m}$

$D := \overline{+\infty}; D(0,0) := 0$

for $i := 1$ **to** m

  for $j := \max(1, i-r)$ **to** $\min(m, i+r)$

    $d(i,j) := \mathrm{Dist}(q_i, c_j)$

    $D(i,j) := d(i,j) + \min\{D(i-1,j), D(i,j-1), D(i-1,j-1)\}$

return $D(m,m)$

Вычисление DTW с ограничением: сложность

Algorithm DTW $(Q, C \in \mathbb{R}^m$, **int** $r)$
$D \in \mathbb{R}^{(1+m)\times(1+m)}, d \in \mathbb{R}^{m\times m}$

$D := \overline{+\infty}; D(0,0) := 0$
for $i := 1$ **to** m
  for $j := \max(1, i-r)$ **to** $\min(m, i+r)$
    $d(i,j) := \mathrm{Dist}(q_i, c_j)$
    $D(i,j) := d(i,j) + \min\{D(i-1,j), D(i,j-1), D(i-1,j-1)\}$
return $D(m,m)$

Дж.У. Хенс. «Козырь»

Вычислительная сложность $\boldsymbol{O(rm)}$

Пространственная сложность $\boldsymbol{O(m^2)}$

Содержание

- Постановка задачи
- Метрика Евклида
- Мера DTW
- **Поиск по образцу на основе DTW**
- Оптимизации поиска на основе DTW

Сложность DTW снижена, но как ускорить поиск по образцу?

Algorithm NaiveSearch ($Q \in \mathbb{R}^m, T, r$)

$bsf := +\infty$
for each $C_i \in S_T^m$
  $dist := \mathrm{DTW}(Q, C_i, r)$
  if $dist < bsf$
    $bsf := dist$
    $C_{\mathrm{bestmatch}} := C_i$
return $C_{\mathrm{bestmatch}}$

- Неравенство треугольника не работает, $\mathrm{DTW}(Q, C_i)$ нельзя применить в качестве нижней границы!
- Вычислительная сложность $O(nmr)$

Нижняя граница (lower bound)

Algorithm LBsearch $(Q \in \mathbb{R}^m, T, r)$

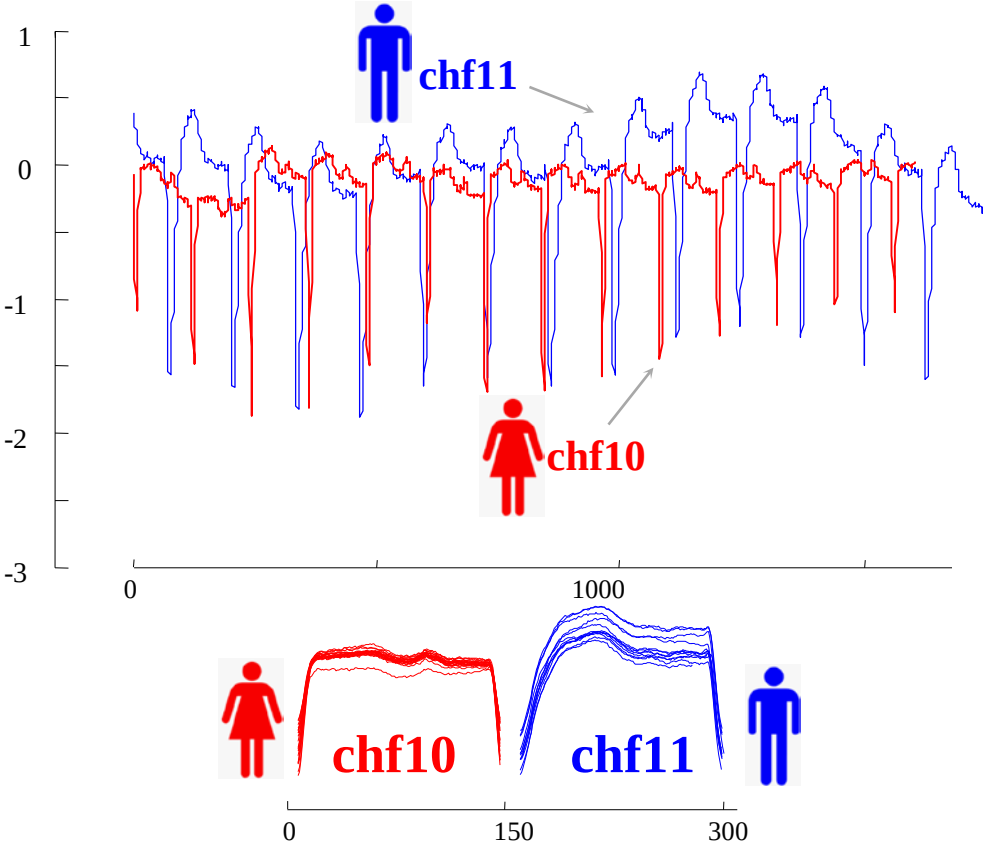
$bsf := +\infty$;
for each $C_i \in S_T^m$
  if $\mathbf{LB}(Q, C_i) < bsf$ **then**
    $dist := \mathrm{DTW}(Q, C_i, r)$
    if $dist < bsf$
      $bsf := dist$
      $C_{\mathrm{bestmatch}} := C_i$
return $C_{\mathrm{bestmatch}}$

- Функция **LB**: $\mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}_+ \cup \{0\}$ со сложностью меньше $O(m^2)$

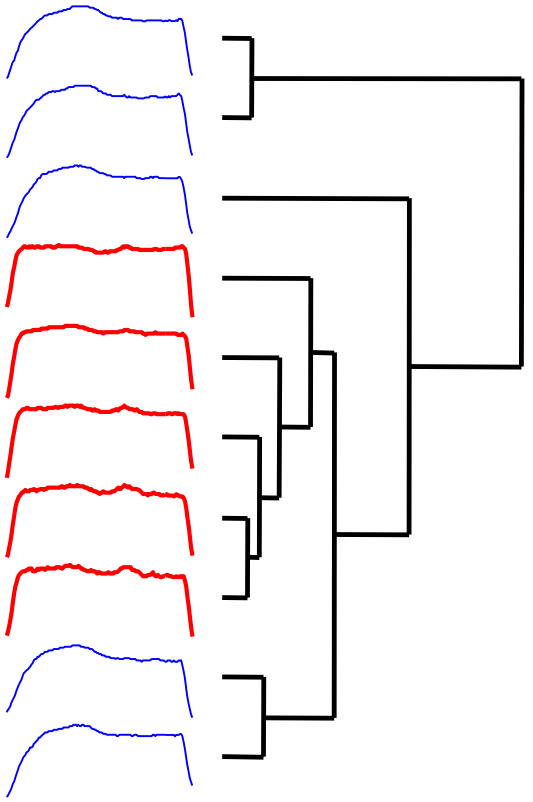
  $\forall\, C, Q$: $\mathbf{LB}(Q, C) \le \mathrm{DTW}(Q, C)$
- Если $\mathbf{LB}(Q, C_i) > bsf$,
  то $\mathrm{DTW}(Q, C_i) > bsf$,
  т.е. C_i заведомо непохож на Q
  и не нужно вычислять $\mathrm{DTW}(Q, C_i)$
- Необходима **z-нормализация S_T^m и Q (всех подпоследовательностей** и запроса)

Важность z-нормализации подпоследовательностей

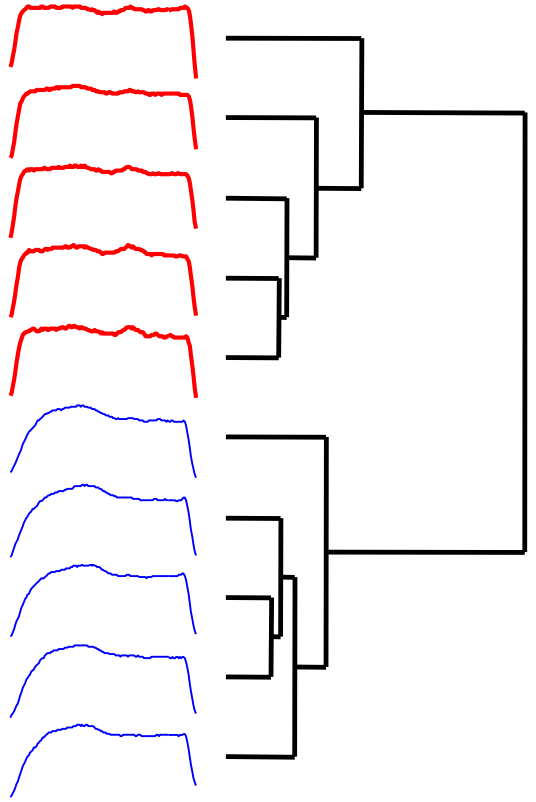
BIDMC Congestive Heart Failure Database*



Не нормализованные

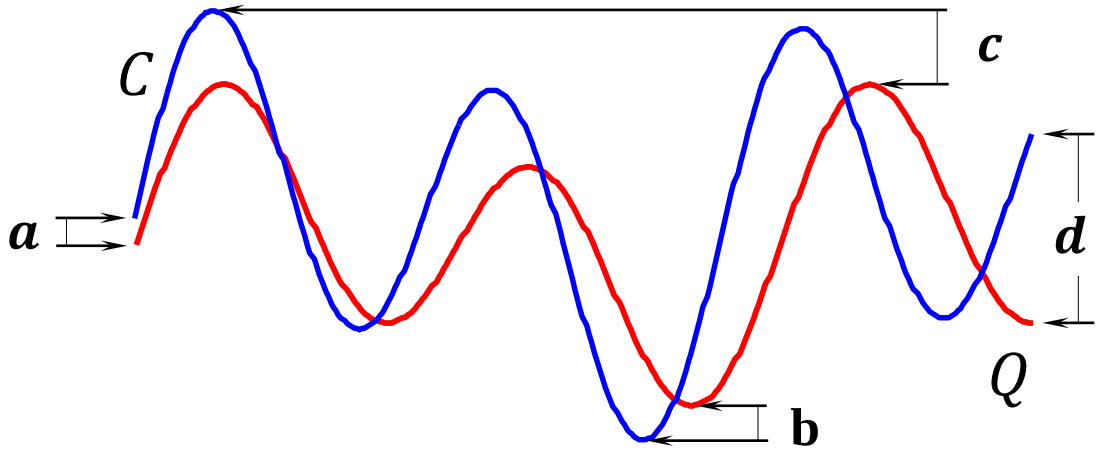


Z-нормализованные



* BIDMC Congestive Heart Failure Database. URL: https://www.kaggle.com/datasets/shymammoth/bidmc-congestive-heart-failure

Нижние границы $\mathbf{LB_{Kim}}$ и $\mathbf{LB_{KimFL}}$*



Санг-Вук Ким
(Sang-Wook Kim)

- Квадрат разности между парой точек Q и C:
  - первая и последняя, сложность $O(1)$:

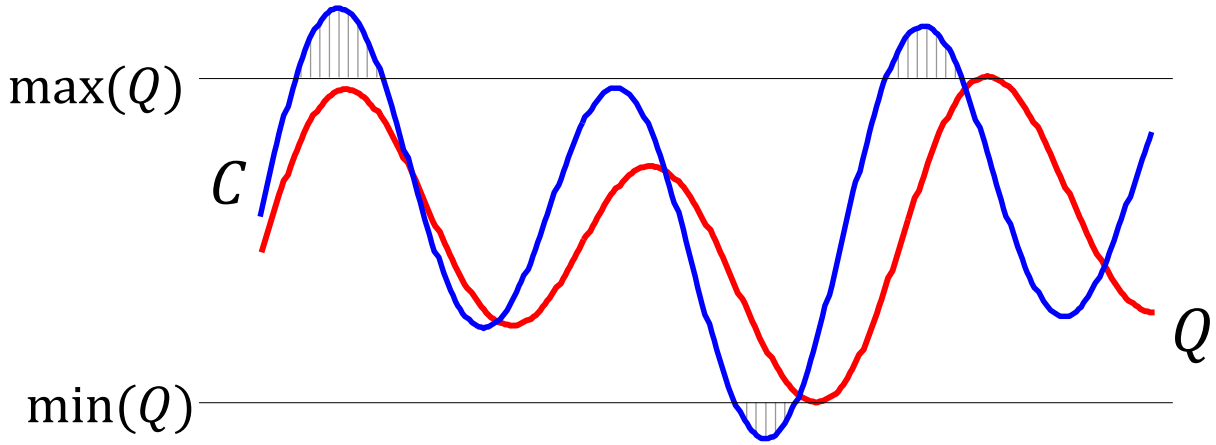
$$\mathrm{LB}_{\mathrm{KimFL}}(Q,C) = \boldsymbol{a}^2 + \boldsymbol{d}^2 = (q_1 - c_1)^2 + (q_m - c_m)^2$$

  - минимальная и максимальная, сложность $O(m)$:

$$\mathrm{LB}_{\mathrm{KimFL}}(Q,C) = \boldsymbol{c}^2 + \boldsymbol{d}^2 = (q_{\max} - c_{\max})^2 + (q_{\min} - c_{\min})^2$$

*Kim S., *et al.* An index-based approach for similarity search supporting time warping in large sequence databases. Proc. of the 17th Int. Conf. on Data Engineering, ICDE 01, April 2-6, 2001, Heidelberg, Germany, pp. 607-614. DOI: 10.1109/ICDE.2001.914875

Нижняя граница LB_{Yi}*



Байоюнг-Ки Йи
(Byoung-Kee Yi)

Сумма квадратов длин ||| дает мин. вклад в DTW, сложность $O(m)$:

$$\mathrm{LB_{Yi}}(Q,C)=\sum_{c_i>\max(q_1,\ldots,q_m)} c_i^2+\sum_{c_i<\min(q_1,\ldots,q_m)} c_i^2$$

* Yi B., Jagadish H., Faloutsos C. Efficient retrieval of similar time sequences under time warping. Proc. of the 14th Int. Conf. on Data Engineering, ICDE 98, Orlando, Florida, USA, February 23-27, 1998, pp. 23-27. DOI: 10.1109/ICDE.1998.655778

Нижняя граница $\mathbf{LB_{Keogh}}$*

$$\mathbf{LB_{Keogh}}(Q,C) = \sum_{i=1}^{m} \begin{cases} (c_i - u_i)^2, c_i > u_i \\ (c_i - \ell_i)^2, c_i < \ell_i \\ 0, \quad otherwise \end{cases}$$

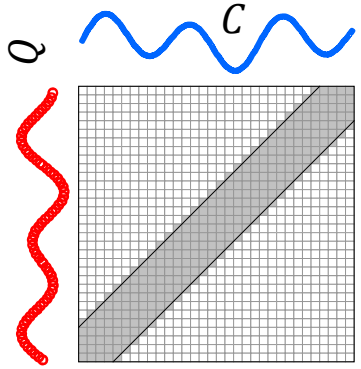
Верхняя оболочка U

$$u_i = \max_{i-r \le k \le i+r} q_k$$

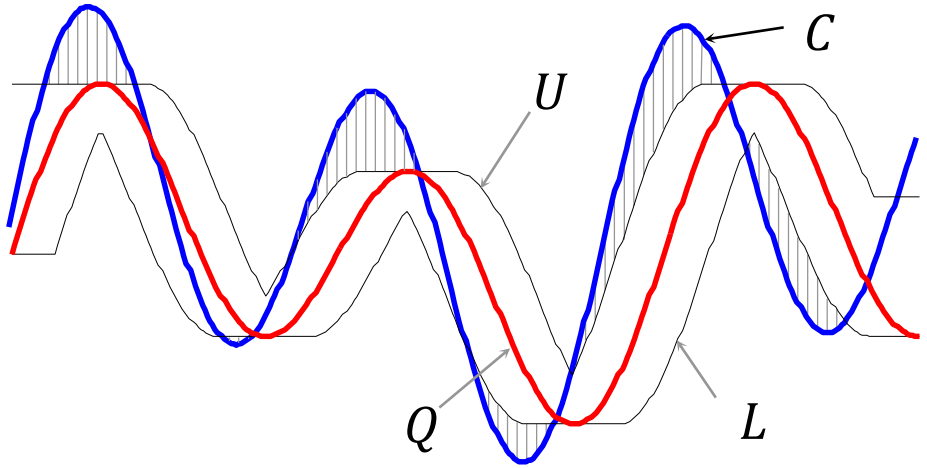
Нижняя оболочка L

$$\ell_i = \min_{i-r \le k \le i+r} q_k$$

Имонн Кеог
(Eamonn Keogh)



Полоса Сако—Чиба



* Ding H., Trajcevski G., Scheuermann P., Wang X., Keogh, E.J. 2008. Querying and mining of time series data: Experimental comparison of representations and distance measures. J. VLDB. 2008. 1 (2). pp. 1542–1552. URL: http://www.vldb.org/pvldb/vol1/1454226.pdf

Нижняя граница $\mathbf{LB_{Keogh}}$*

$$\mathbf{LB_{Keogh}}(Q,C) = \sum_{i=1}^{m} \begin{cases} (c_i - u_i)^2, c_i > u_i \\ (c_i - \ell_i)^2, c_i < \ell_i \\ 0, \quad otherwise \end{cases}$$

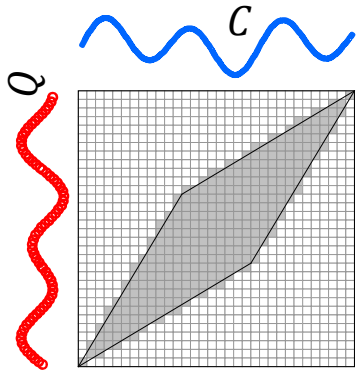
Верхняя оболочка U

$$u_i = \max_{i-r \le k \le i+r} q_k$$

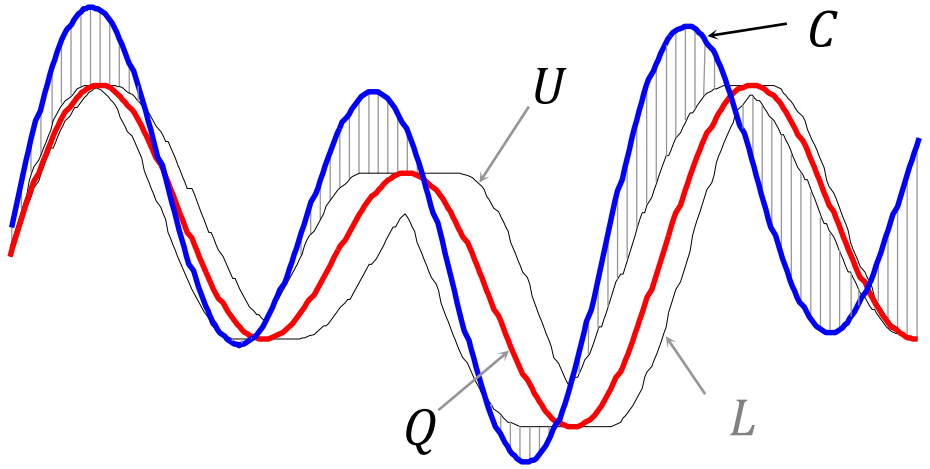
Нижняя оболочка L

$$\ell_i = \min_{i-r \le k \le i+r} q_k$$

Имонн Кеог
(Eamonn Keogh)

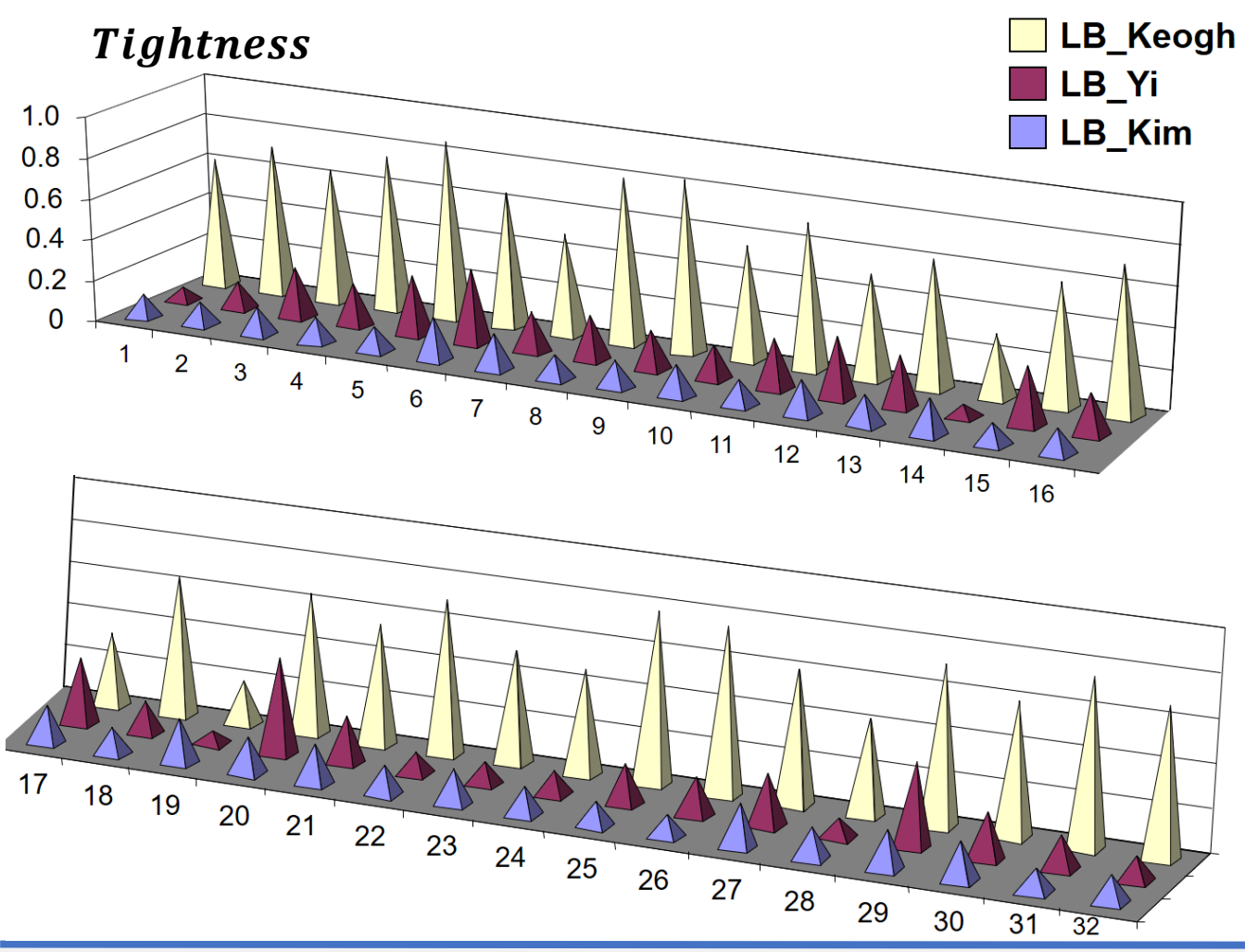


Параллелограмм Итакуры



* Ding H., Trajcevski G., Scheuermann P., Wang X., Keogh, E.J. 2008. Querying and mining of time series data: Experimental comparison of representations and distance measures. J. VLDB. 2008. 1 (2). pp. 1542–1552. URL: http://www.vldb.org/pvldb/vol1/1454226.pdf

Сравнение нижних границ: Tightness (узость)

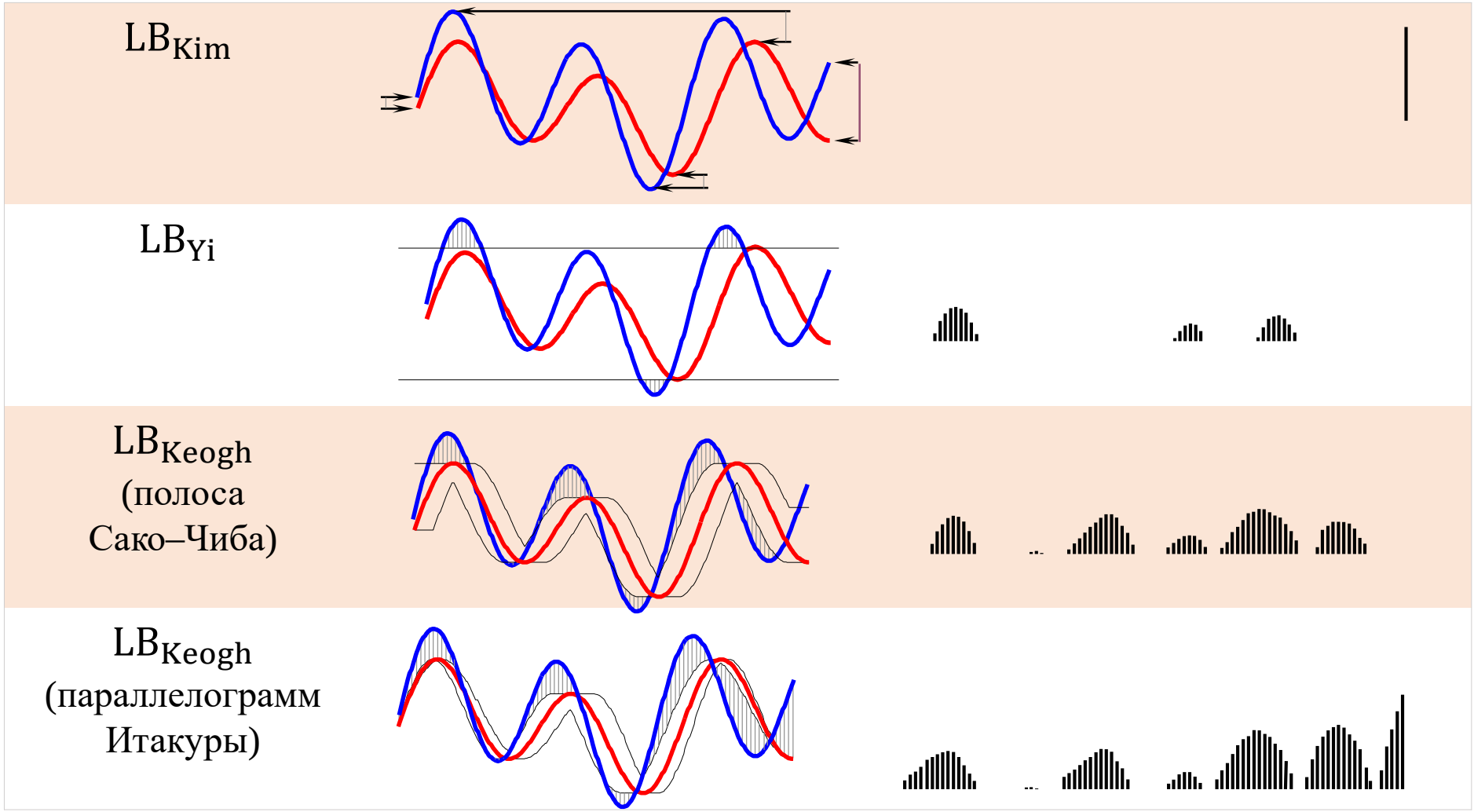


- Взяли 32 временных ряда из различных предметных областей
- В каждом ряде взяли 50 случайных подпоследовательностей наиболее типичной длины 256
- Для каждой пары подпоследовательностей Q, C вычислили $\mathrm{LB}(Q, C)$ и $\mathrm{DTW}(Q, C)$ и усреднили
- В итоге вычислили метрику

$$\boldsymbol{Tightness} = \frac{\mathbf{LB}}{\mathbf{DTW}}$$

($0 \leq Tightness \leq 1$, большее значение лучше)

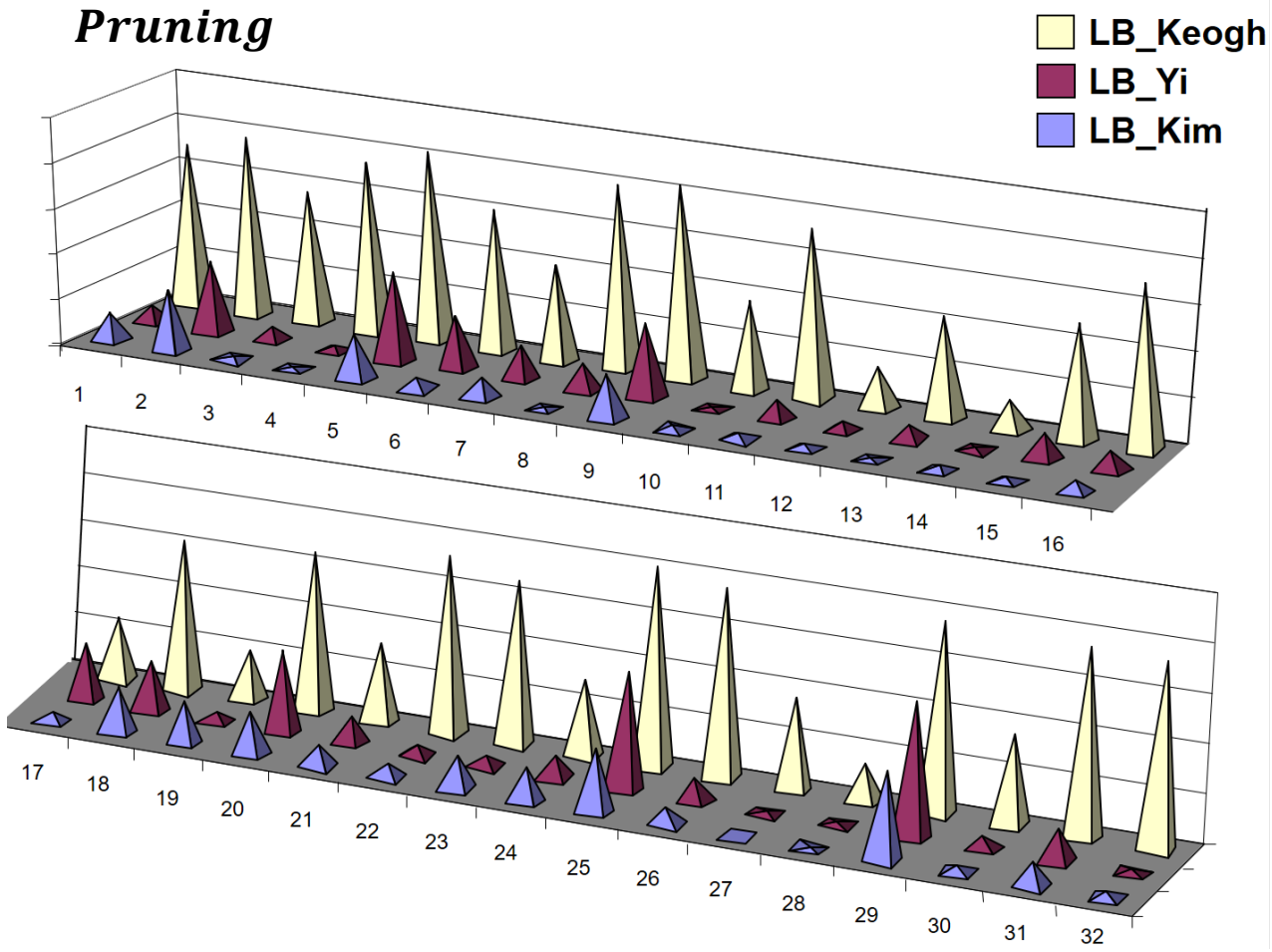
Сравнение нижних границ: Tightness (узость)



Tightness

нижней границы пропорциональна длине |||

Сравнение нижних границ: Pruning power (объем отбрасывания)



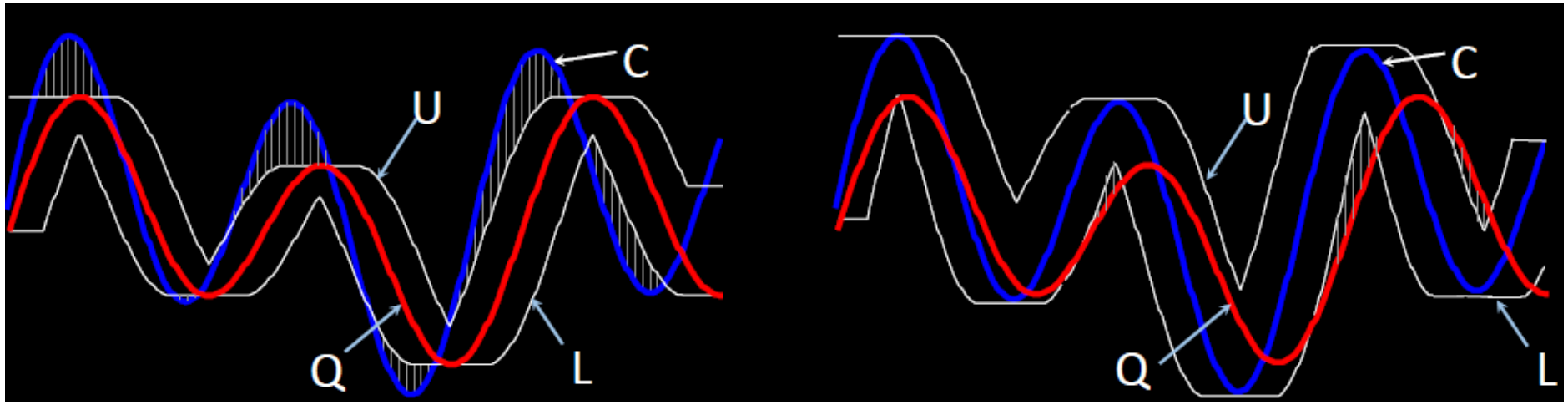
- Взяли 32 временных ряда из различных предметных областей
- В каждом ряде взяли 50 случайных подпоследовательностей наиболее типичной длины 256
- Каждую подпоследовательность взяли как запрос для поиска наиболее похожей подпоследовательности среди остальных и усреднили число случаев где НЕ вычисляли DTW
- В итоге вычисли метрику

$$\boldsymbol{Pruning} = \frac{\text{число случаев где НЕ вычисляли } \mathbf{DTW}}{|S_T^m|}$$

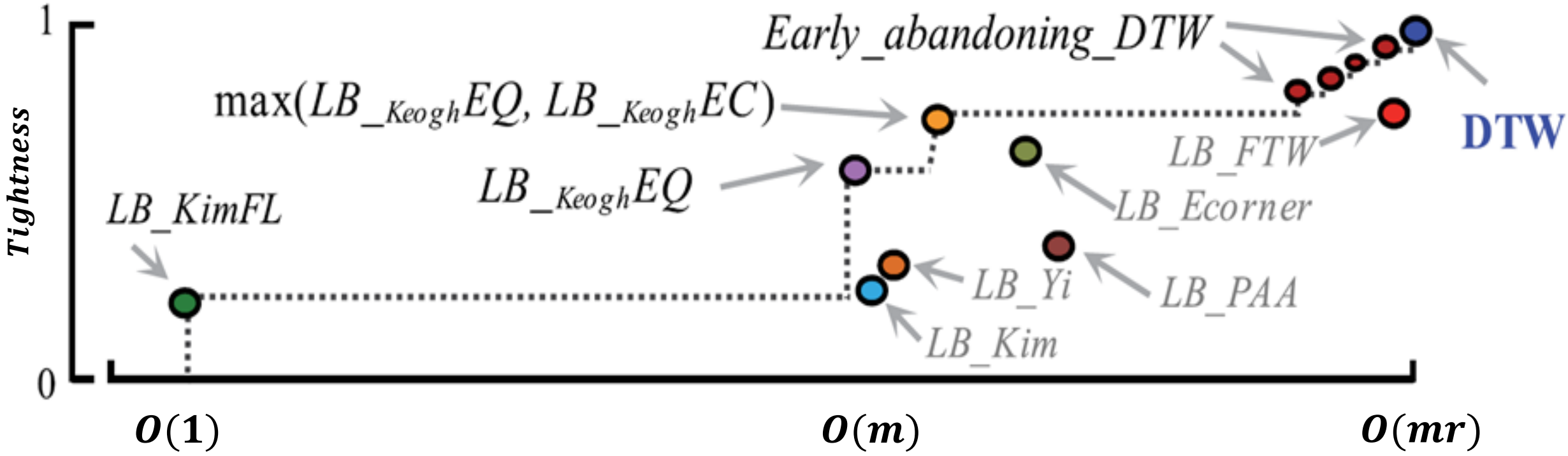
($0 \le Pruning \le 1$,
большее значение лучше)

Нижняя граница $\mathbf{LB_{Keogh}EC}$

- Оболочка строится вокруг запроса (аббр. Envelope around the Query): нижняя граница $\mathrm{LB_{Keogh}EQ} \equiv \mathrm{LB_{Keogh}}$
- Q и C меняются местами (аббр. Envelope around the Candidate): нижняя граница $\mathbf{LB_{Keogh}EC}(Q, C) = \mathrm{LB_{Keogh}EQ}(C, Q)$
  NB! В общем случае $\mathrm{LB_{Keogh}EC} \neq \mathrm{LB_{Keogh}EQ}$



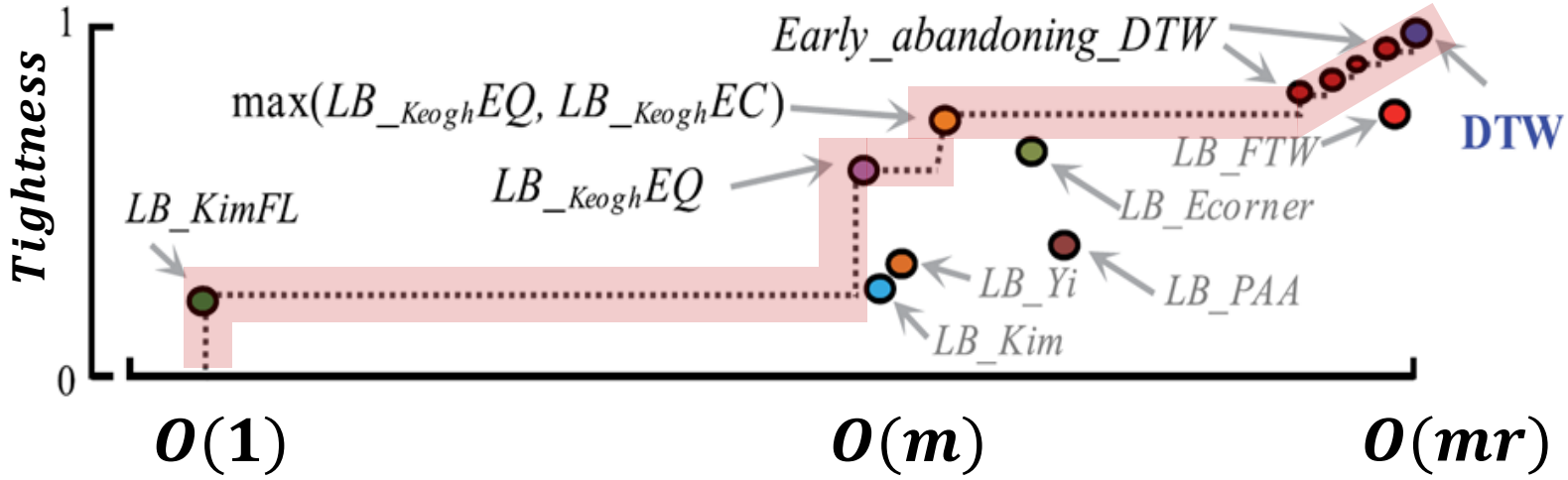
Прочие нижние границы* (18+)



* Rakthanmanon T. *et al.* Addressing big data time series: Mining trillions of time series subsequences under Dynamic Time Warping. TKDD. 2013. Vol. 7, No. 3. P. 10. DOI: 10.1145/2500489

Какие нижние границы применять?

- Все в линейке каскадом от менее к более вычислительно затратным
- Отказ от любой нижней границы в каскаде замедляет вычисления минимум вдвое
- Применение каскада позволяет достичь $\boldsymbol{Prune = 0.99999}$*



Большой каскад / Гос. музей-заповедник «Петергоф»

* Rakthanmanon T. *et al.* Addressing big data time series: Mining trillions of time series subsequences under Dynamic Time Warping. TKDD. 2013. Vol. 7, No. 3. P. 10. DOI: 10.1145/2500489

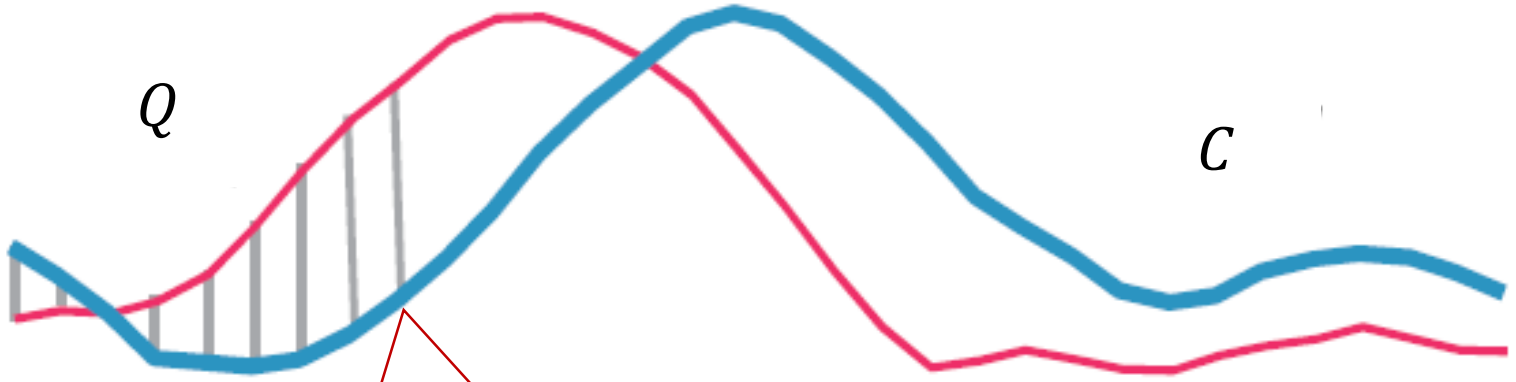
Содержание

Оптимизация: ранний останов вычислений (early abandoning)

Прекратить вычисления, если текущий результат больше bsf

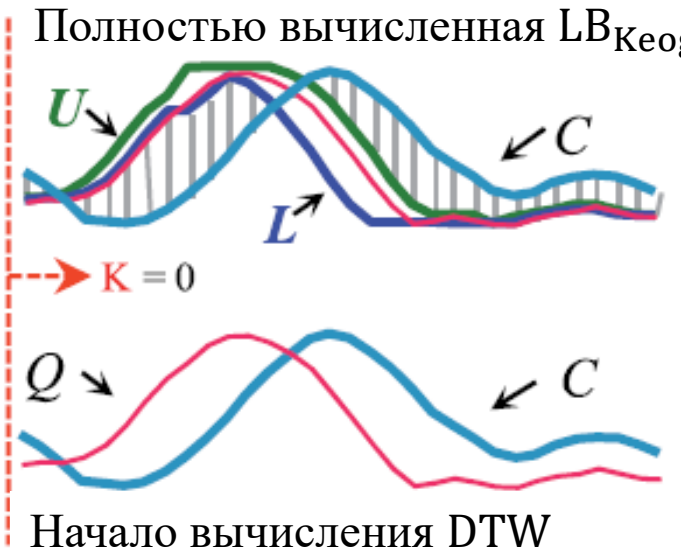
1. Ранний останов вычисления ED и LB_{Keogh}
2. Ранний останов вычисления DTW
3. Более раннее отбрасывание DTW с помощью LB_{Keogh}
4. Переупорядочивание раннего отбрасывания

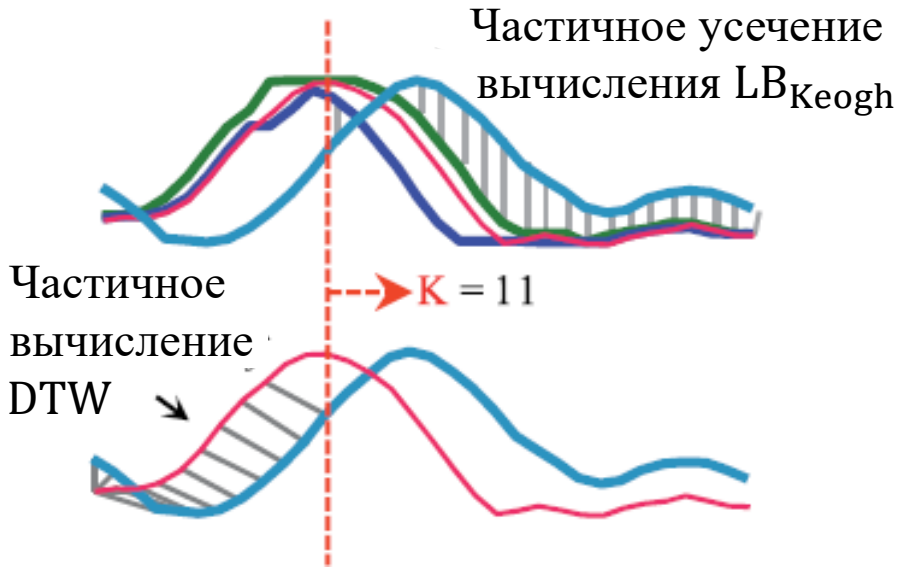
Ранний останов вычисления ED и LB_{Keogh}



На этой точке можно остановить вычисления, если текущий результат $\sum(q_i - c_i)^2$ превысил bsf

Ранний останов вычисления DTW

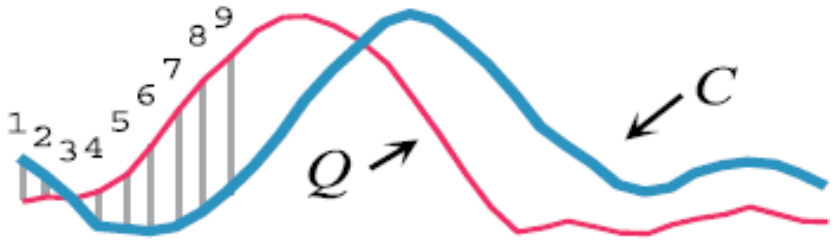




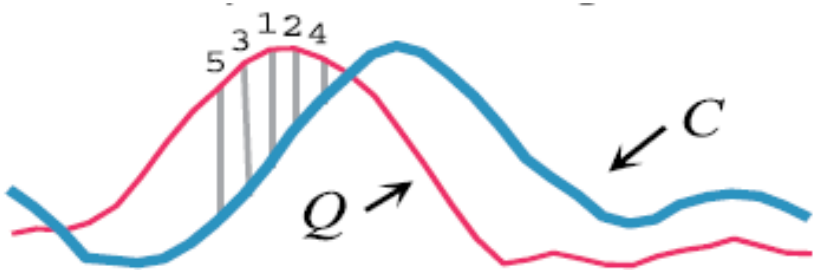
- Пусть LB_{Keogh} подсчитана, но необходимо вычислить DTW. Можно выполнять инкрементное вычисление DTW слева направо и по мере постепенного вычисления от 1 до K суммировать частичное накопление DTW с вкладом LB_{Keogh} от $K + 1$ до m
- $\min_j \mathrm{DTW}(Q_{1:K}, C_{1:K+j}) + \mathrm{LB}_{\mathrm{Keogh}}(Q_{K+r+1:m}, C_{K+r+1:m}) < DTW(Q_{1:m}, C_{1:m})$

Переупорядочивание вычислений (reordering) и их ранний останов

Обычный порядок вычислений
до их раннего останова



Оптимизированный порядок
вычислений до их раннего останова



- Эвристика: оптимальным упорядочением является сортировка индексов на основе абсолютных значений $\hat{Q}$. Обоснование: значение q_i будет сравниваться со многими значениями c_i во время поиска. Распределение многих c_i будет гауссовым, со средним значением 0. Поэтому промежутки из Q, наиболее удаленные от μ_Q, в среднем вносят наибольший вклад в расстояние
- Проверка эвристики: для подпоследовательности ряда ЭКГ вычислили полное евклидово расстояние до 10^6 других случайных подпоследовательностей ЭКГ. Затем нашли наилучший порядок вычислений: взяли c_i и отсортировали их по убыванию их вкладов в евклидово расстояние. В итоге сравнили это эмпирически оптимальное упорядочение с прогнозируемым (сортировка индексов по абсолютным значениям Q) и обнаружили, что ранговая корреляция составляет 0.999
- Прием подходит для ED, $\mathrm{LB}_{\mathrm{Keogh}}$ и может сочетаться с ранним остановом Z-нормализации

Литература

1. Rakthanmanon T., Campana B.J.L., Mueen A. *et al.* Addressing big data time series: Mining trillions of time series subsequences under Dynamic Time Warping. TKDD. 2013. Vol. 7, No. 3. P. 10. https://doi.org/10.1145/2500489.
2. Ratanamahatana C.A., Keogh E.J. Three myths about Dynamic Time Warping data mining. Proc. of the 2005 SIAM Int. Conf. on Data Mining, SDM 2005, Newport Beach, CA, USA, April 21–23, 2005. 2005. P. 506–510. https://doi.org/10.1137/1.9781611972757.50.
3. Sakoe H., Chiba S. Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. on Acoustics, Speech, and Signal Processing, 1978. 26(1), 43-49. https://doi.org/10.1109/TASSP.1978.1163055