Time Series Management

Michele Linardi Ph.D.

michele.linardi@orange.fr

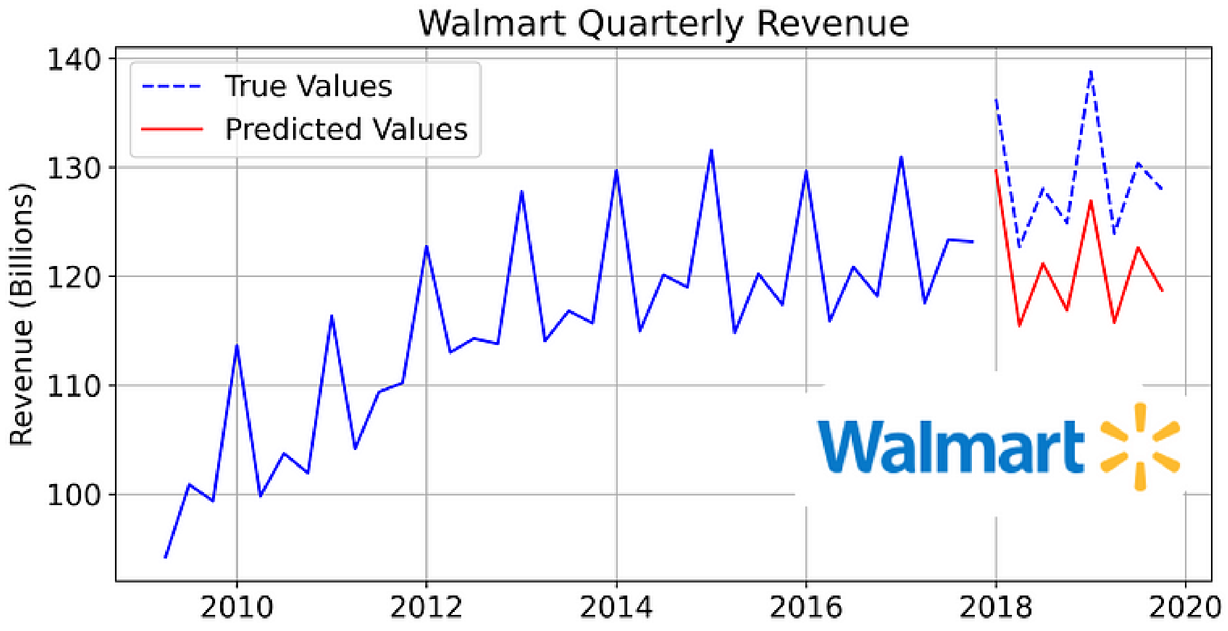
Some of the slides of thes course are taken from the **excellent** Tutorial of **Eammon Keogh**
A Decade of Progress in Indexing and Mining Large Time Series Databases. VLDB 2006.

Syllabus

- Ubiquity of data series collections
- Time series data mining
- Similarity Search
- Metrics
- Speedup computation with distance lower bounding

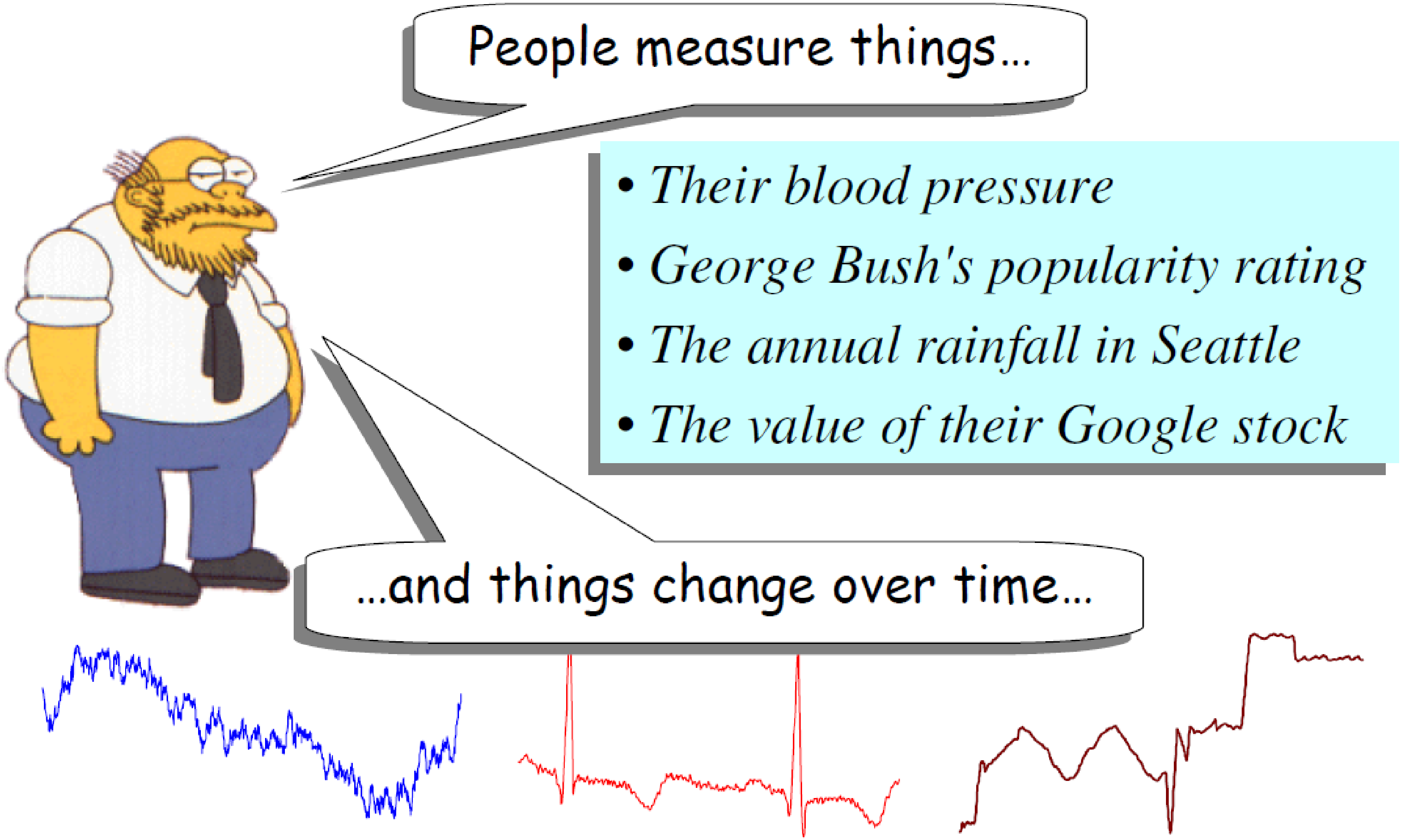
Time series data… quick recap

- A **univariate time series** is a sequence of measurements of the same variable collected over time. Most often, the measurements are made at regular time intervals.



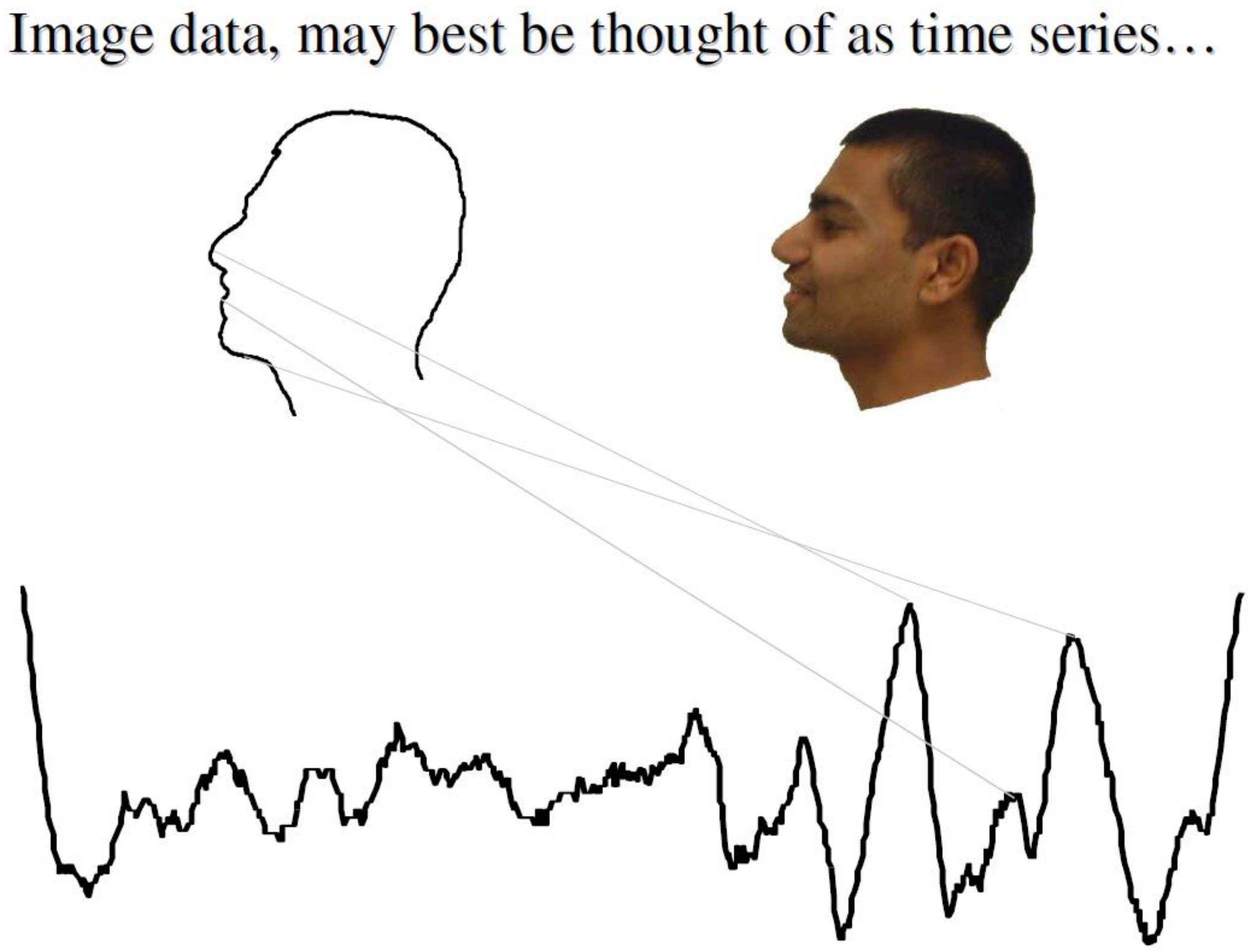
https://www.kaggle.com/code/anushkaml/walmart-time-series-sales-forecasting

Time series are ubiquitous

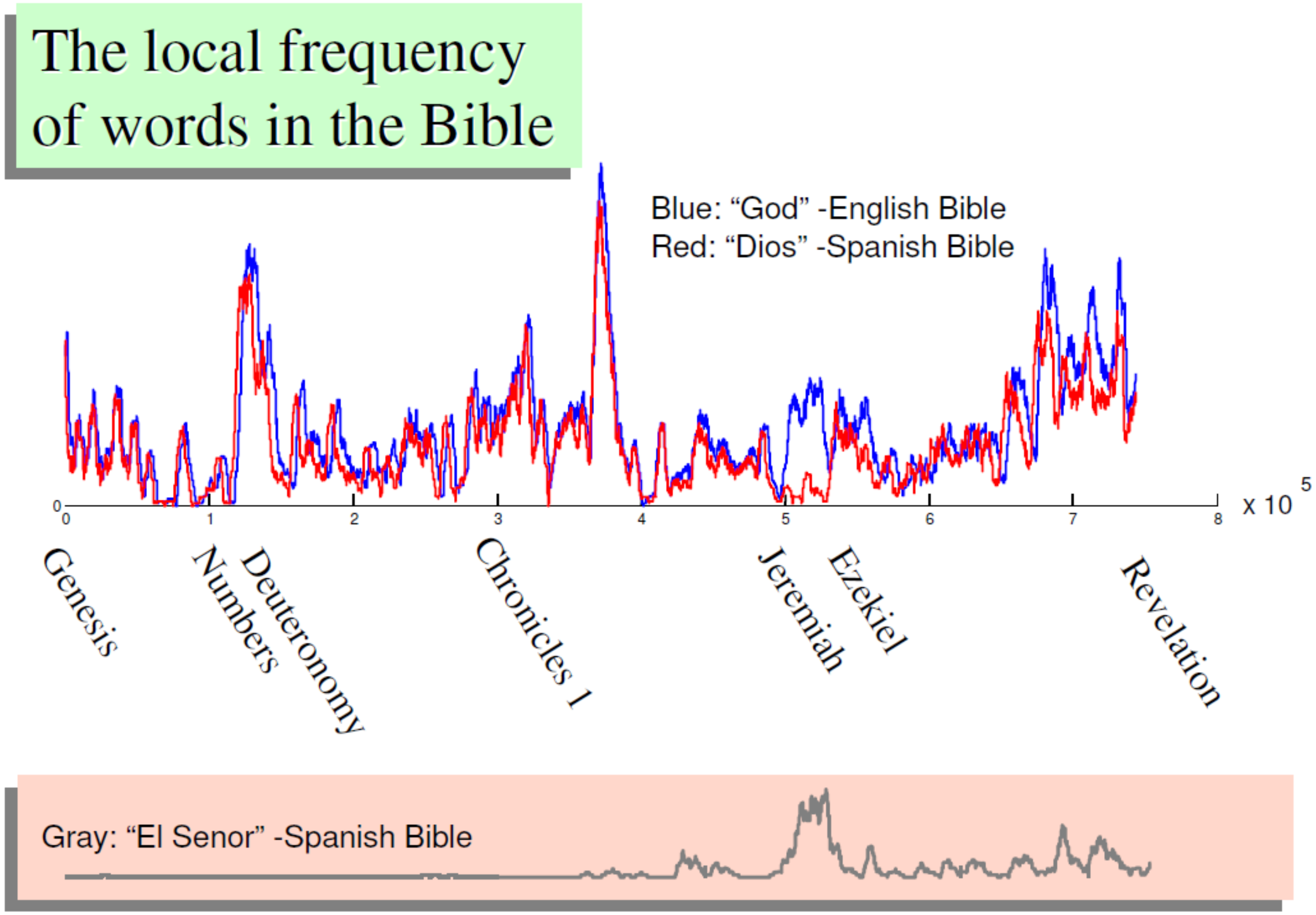


Thus time series occur in virtually every medical, scientific and businesses domain

Time series are ubiquitous (1/4)



Time series are ubiquitous (2/4)

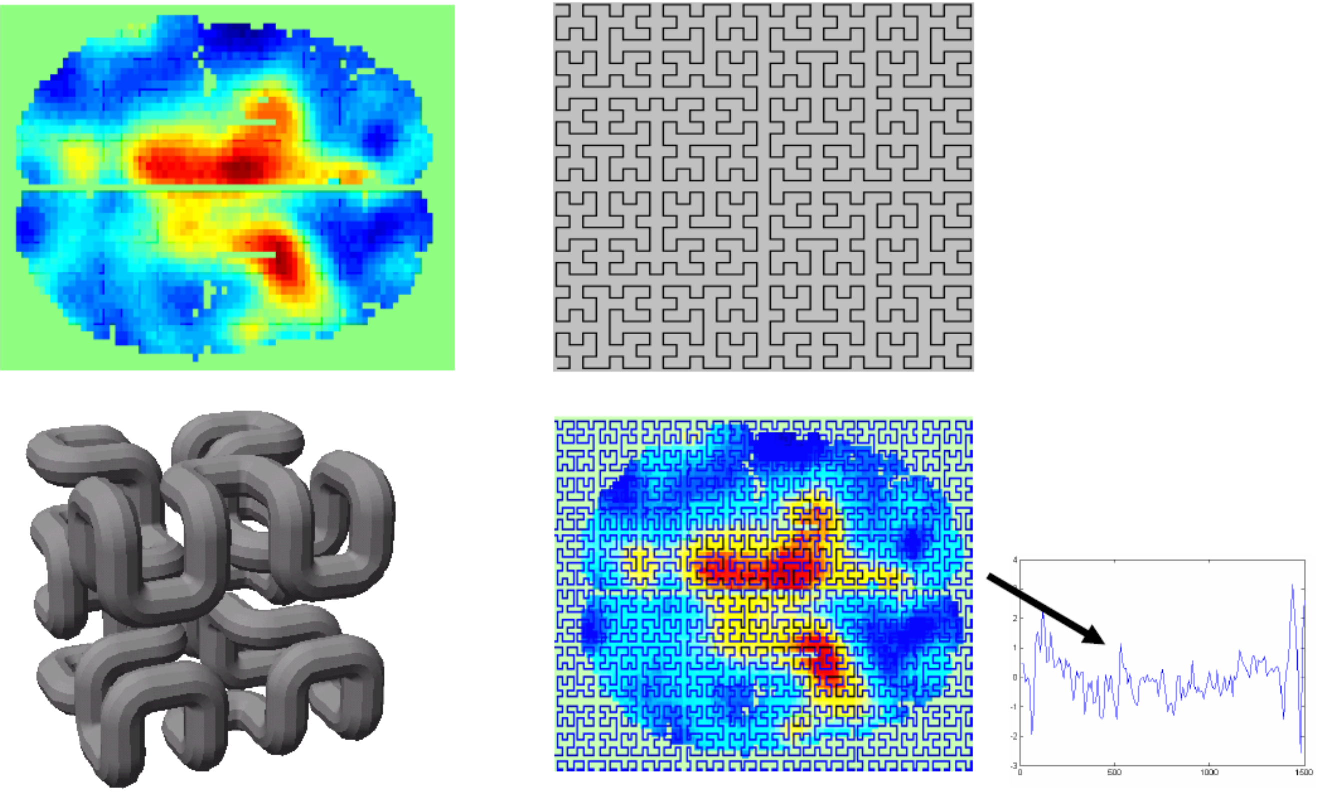


Time series are ubiquitous (3/4)

Video data, may best be thought of as time series…

Time series are ubiquitous (4/4)

Brain scans (3D voxels), may best be thought of as time series..



Wang, Kontos, Li and Megalooikonomou ICASSP 2004

Why is Working With Time Series is so difficult? (1/3)

1 Hour of EKG data: 1 Gigabyte.

Typical Weblog: 5 Gigabytes per week.

Space Shuttle Database: 200 Gigabytes and growing.

Macho Database (Canadian Astronomy Data Centre): 3 Terabytes, updated with 3 gigabytes a day.

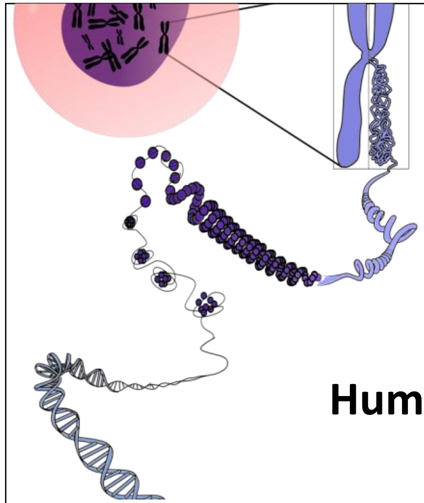
Why is Working With Time Series is so difficult? (2/3)

NASA's Solar Observatory

1.5 TB per day

Large Synoptic Survey Telescope (2019)

~30 TB per night



Human Genome project

130 TB

passenger aircrafts

20 TB per hour

data center and services monitoring

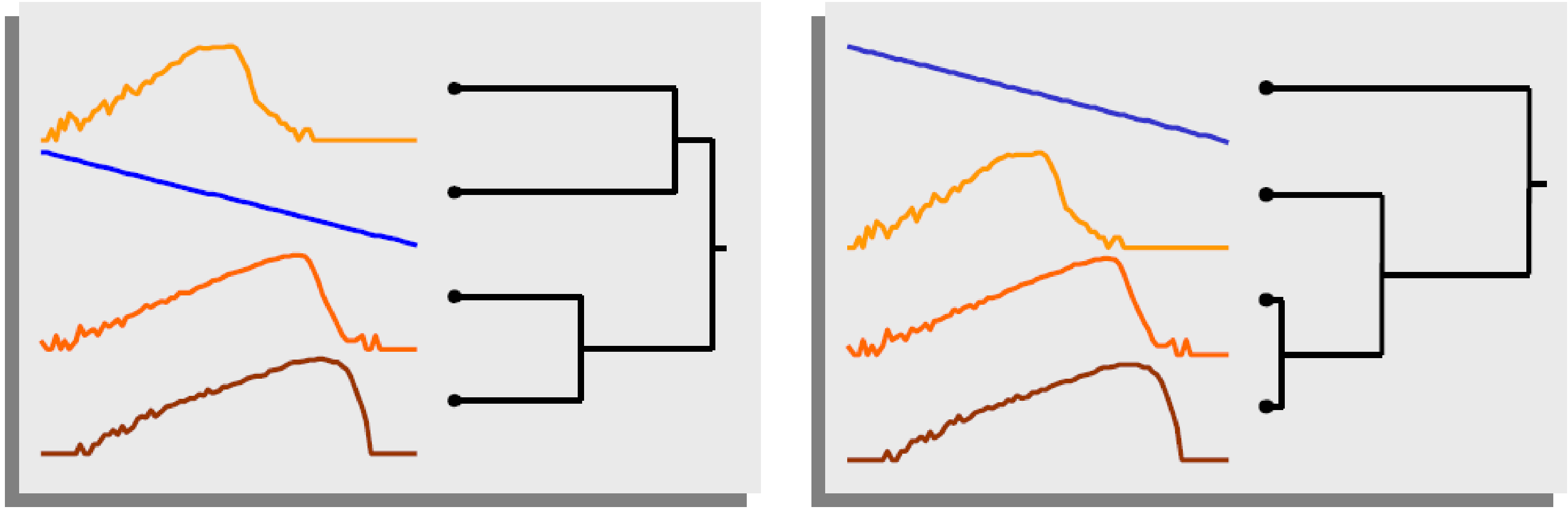
2B data series

4M points/sec

facebook

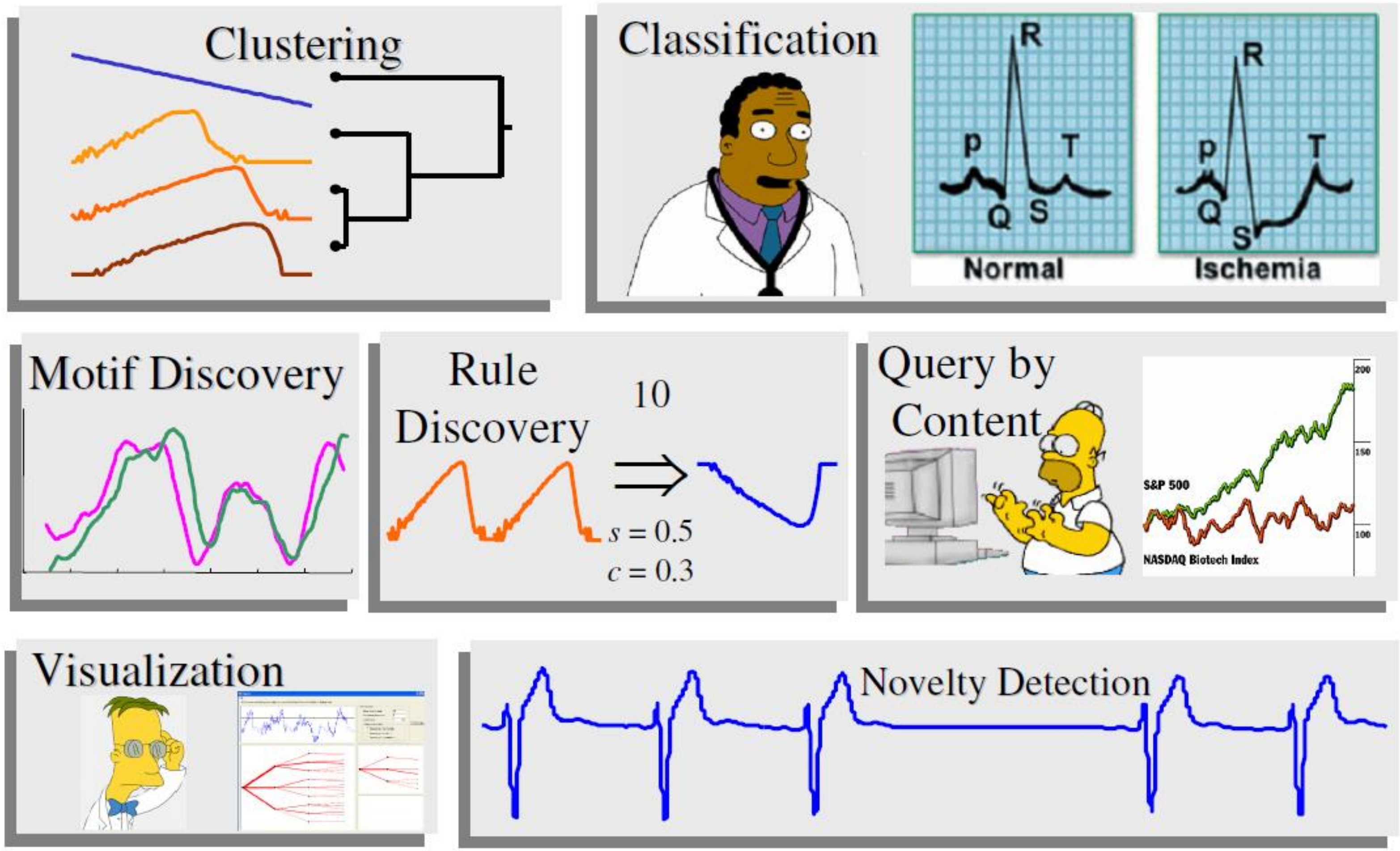
Why is Working With Time Series is so difficult? (3/3)

Answer: We are dealing with subjectivity



The definition of similarity depends on the user, the domain and the task at hand. We need to be able to handle this subjectivity.

Problems requiring Similarity Search



Important Data Mining questions

How do we define similarity ?

How do we search large time series collection quickly ?

What is similarity ?

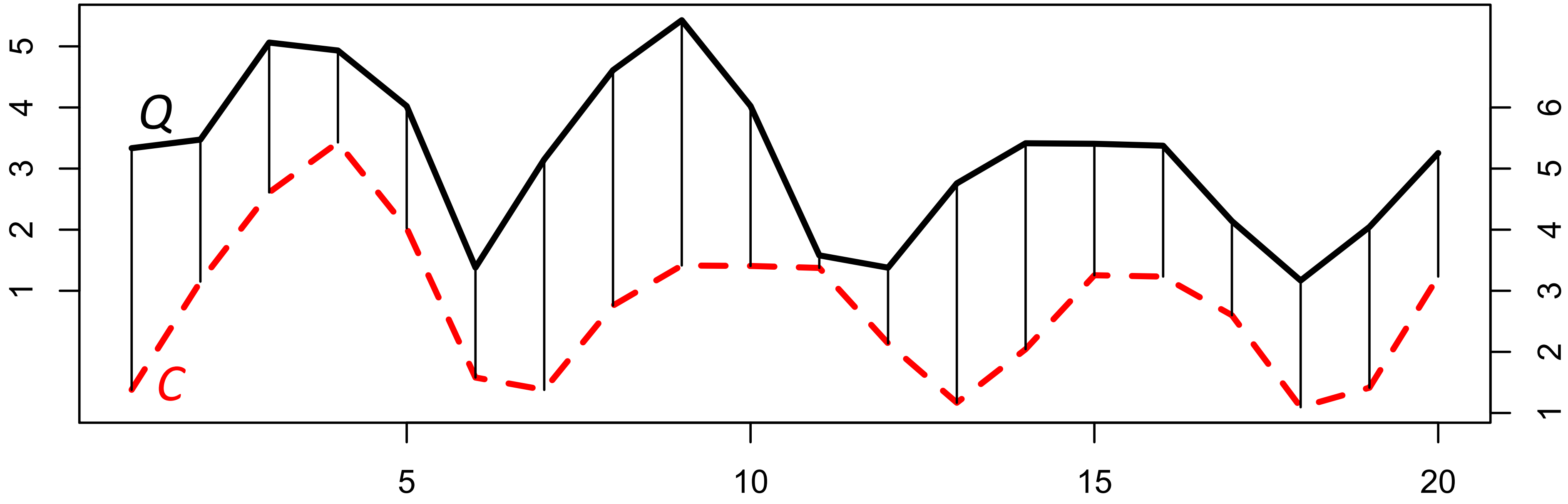
What is similarity ?

The quality or state of being similar; likeness; resemblance; as, a similarity of features.

Similarity is hard to define, but…"We know it when we see it"

The real meaning of similarity is a philosophical question.

Similarity Measure: Euclidean



Given two time series $Q = q_1 \ldots q_n$ and $C = c_1 \ldots c_n$

their Euclidean distance is defined as: $$ED(Q,C) = \sqrt{\sum_{i=1}^{n} (q_i - c_i)^2}$$

Preprocessing the data before distance calculations

- Z-Normalization (Amplitude Scaling)
- Linear Trend
- Noise

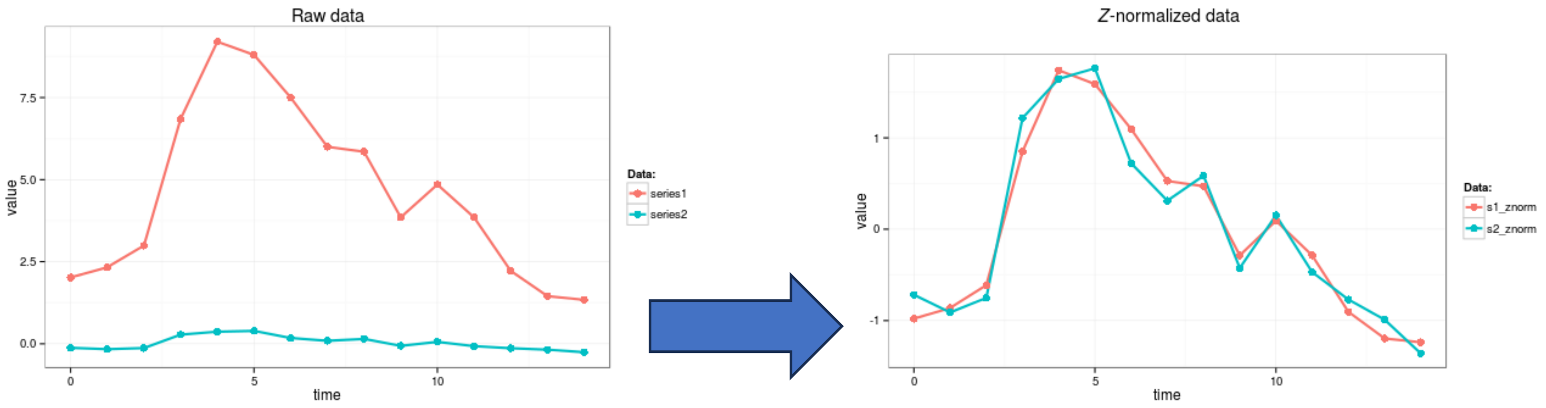
Z Normalization

Z-normalization, also known as

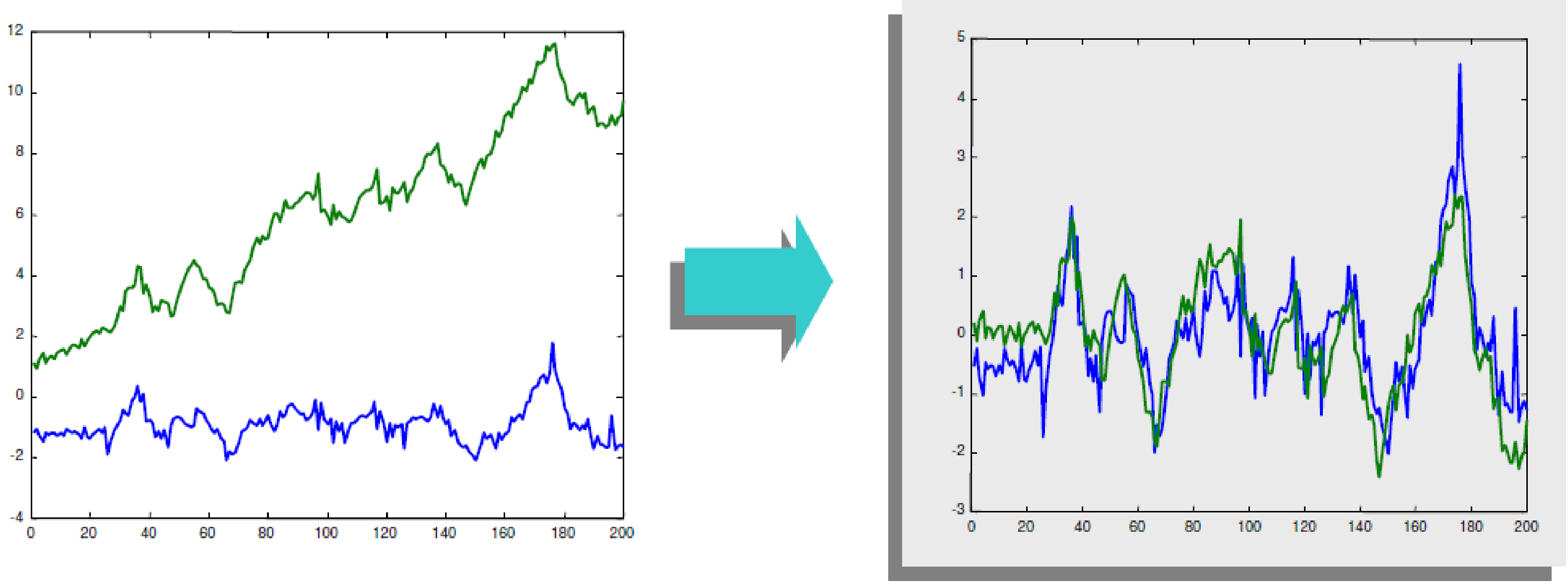
“Normalization to Zero Mean and Unit of Energy”.

The procedure ensures, that all elements of the input vector are transformed into the output vector whose mean is approximately 0 while the standard deviation is in a range close to 1.

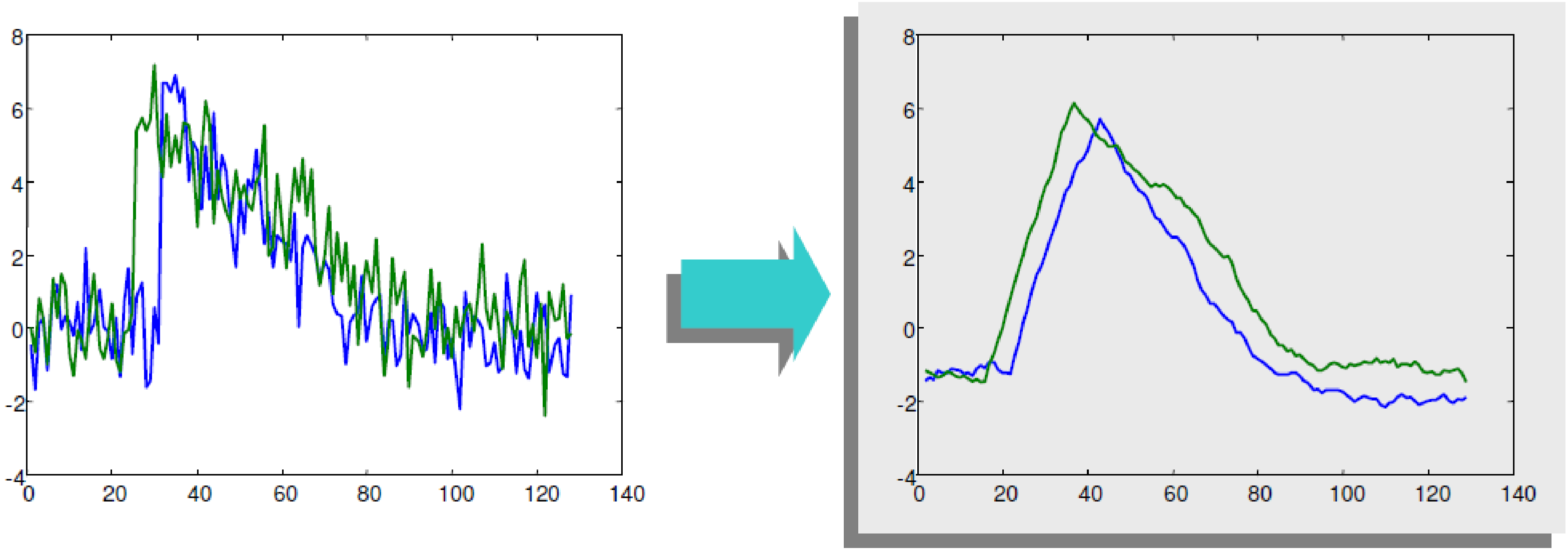
$$x_i^{'} = \frac{x_i - \mu}{\sigma}, \text{ where } i \in \mathbb{N}$$



Remove linear trend



Removing Noise

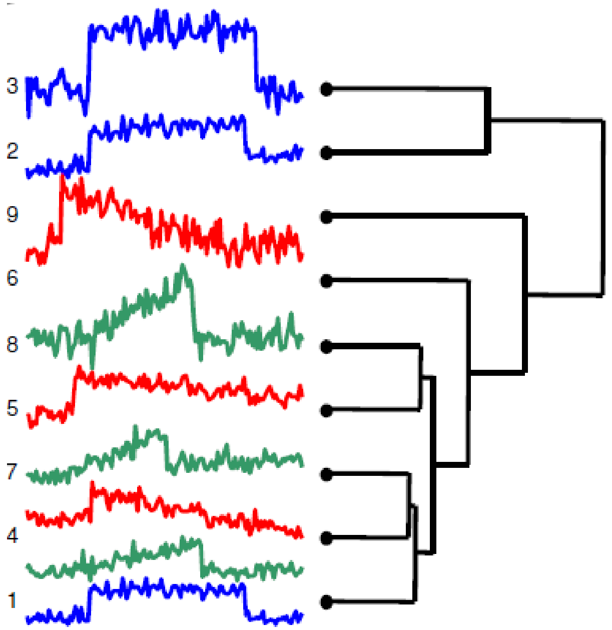


Smoothing function: Remove noise component

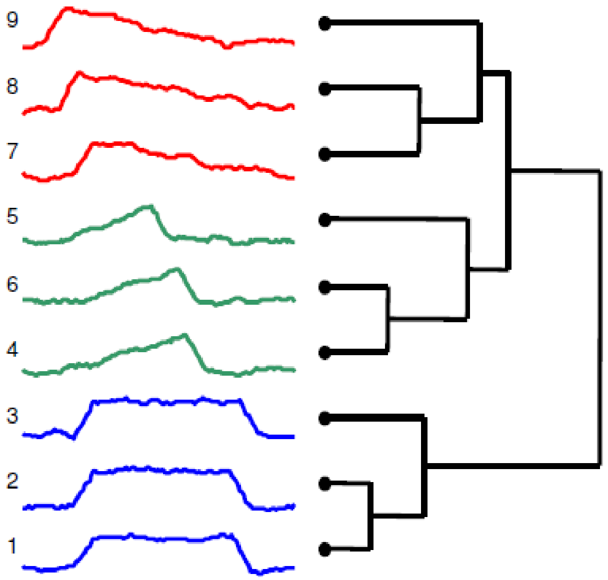
e.g., Average each datapoints value with its neighbors

Importance of data pre-processing

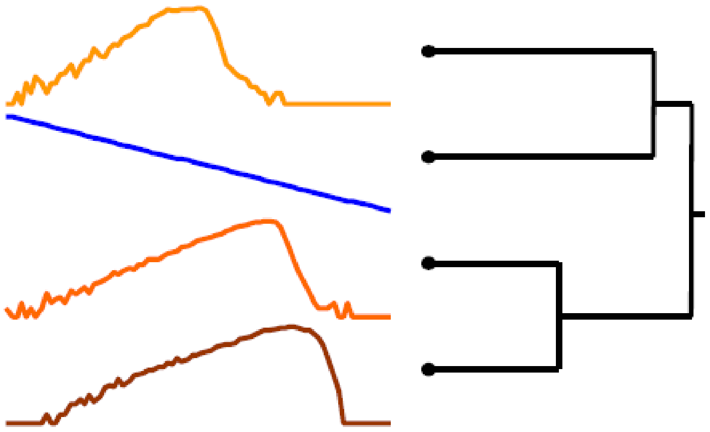
Clustered using Euclidean distance on the raw data.

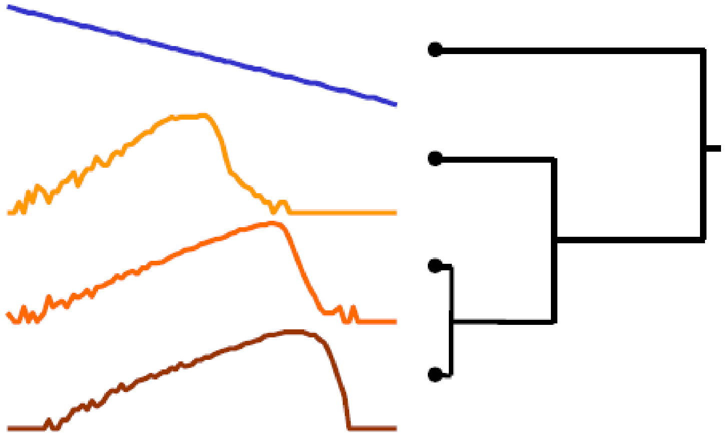


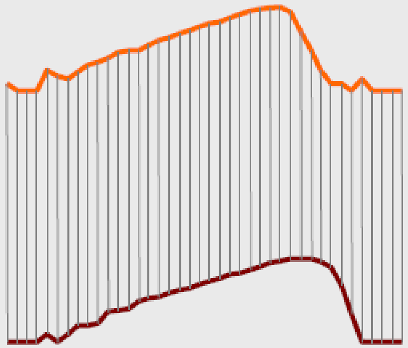
Clustered using Euclidean distance, after removing noise, linear trend, and Z-Normalization



Dynamic Time Warping

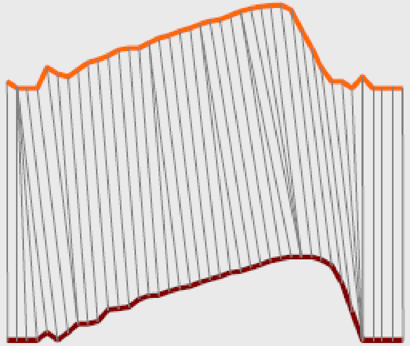






Fixed Time Axis

Sequences are aligned "one to one".

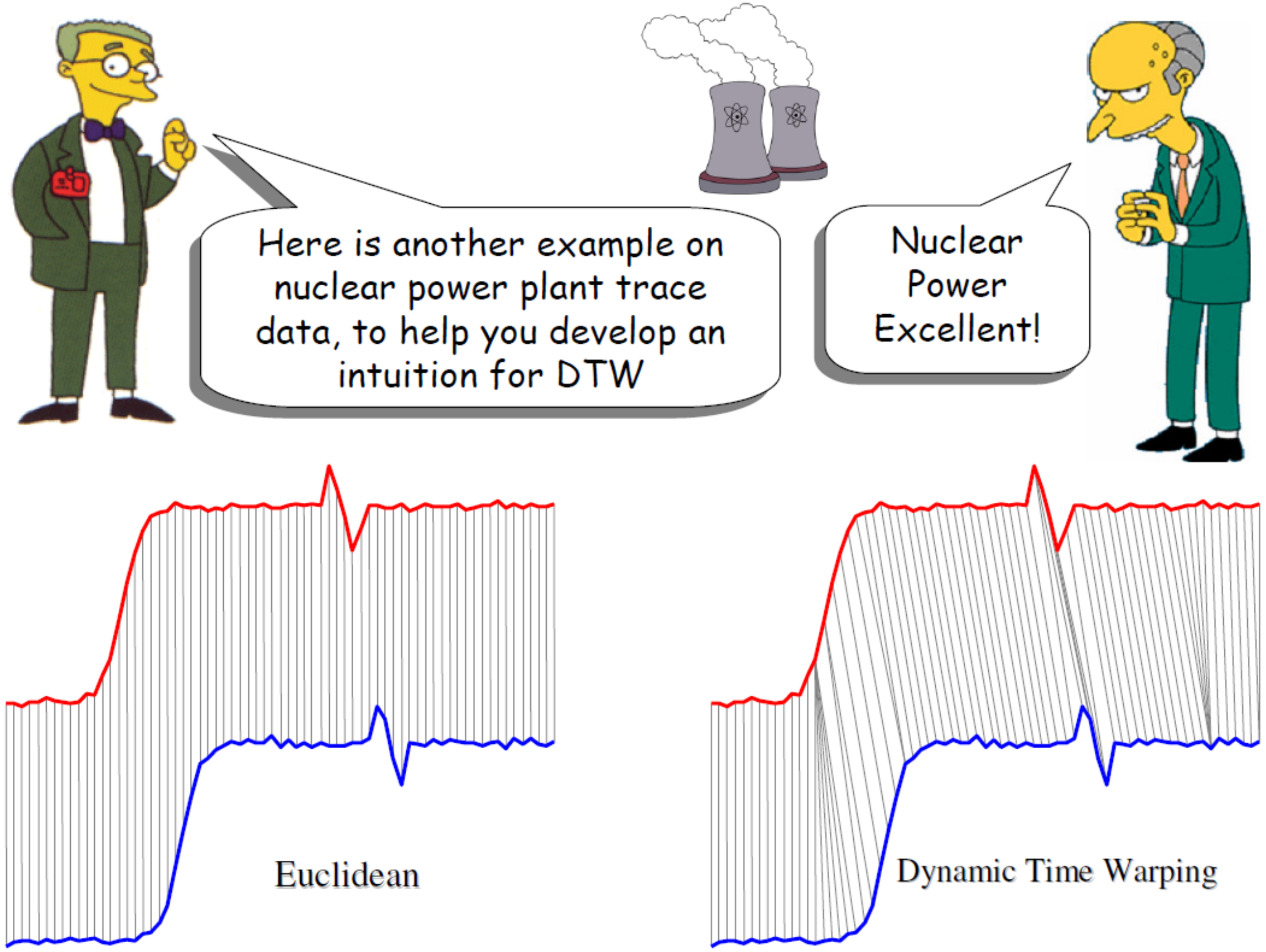


"Warped" Time Axis

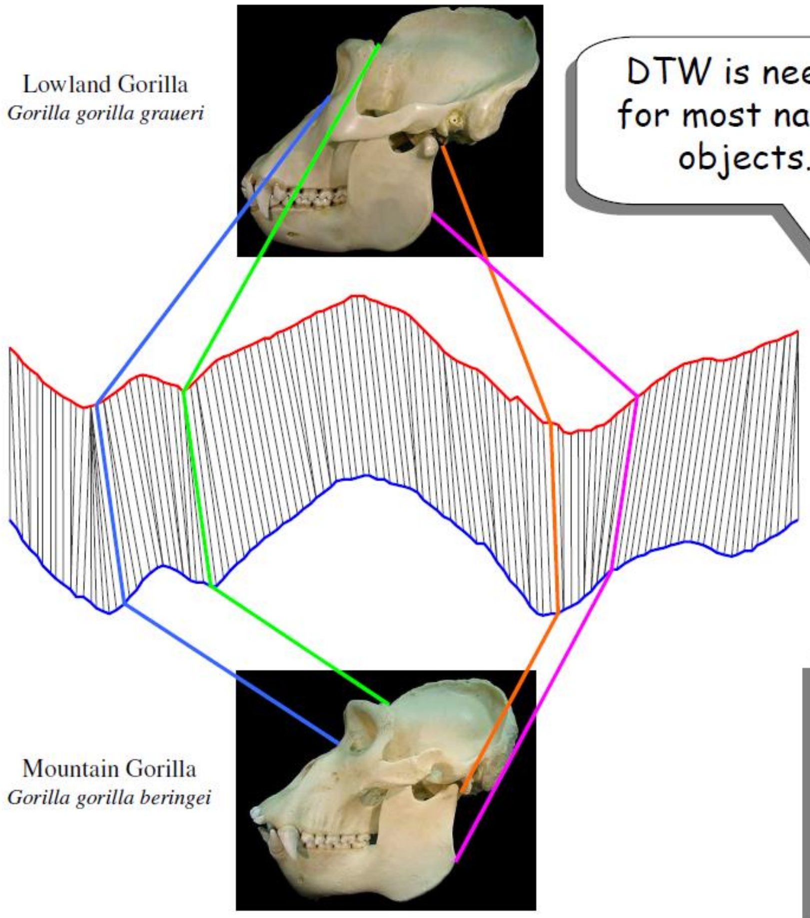
Nonlinear alignments are possible.

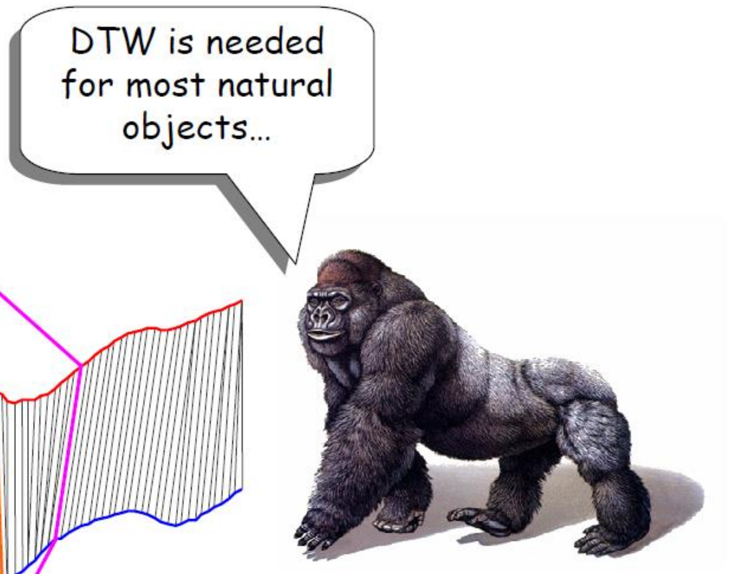
Note: We will first see the utility of DTW, then see how it is calculated.

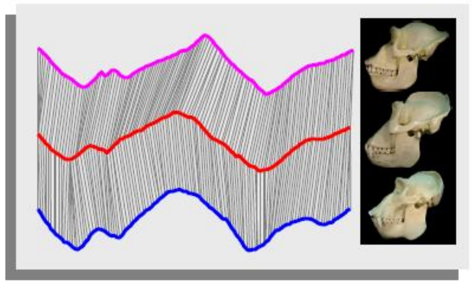
Some real-world example (1/3)



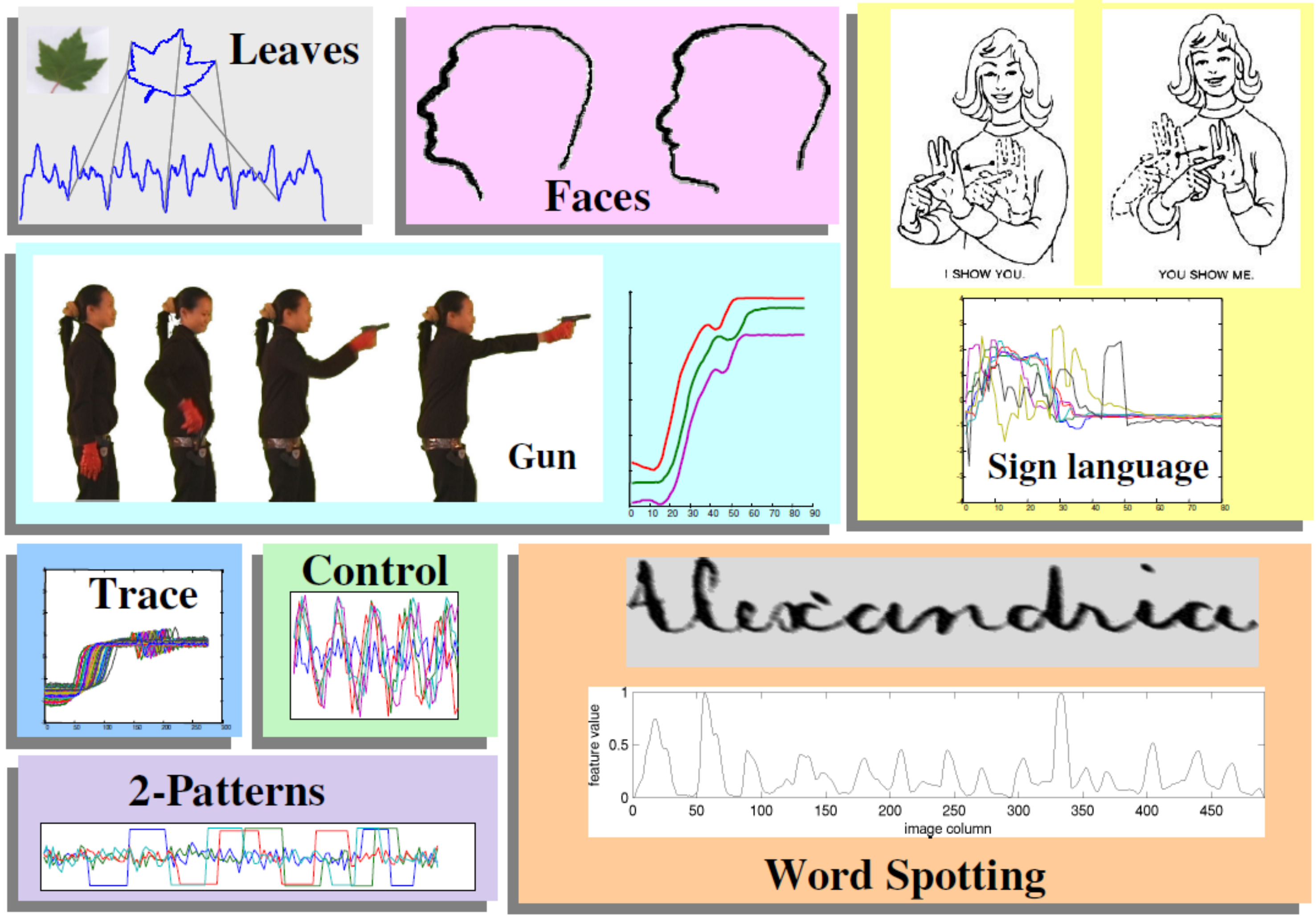
Some real-world example (2/3)



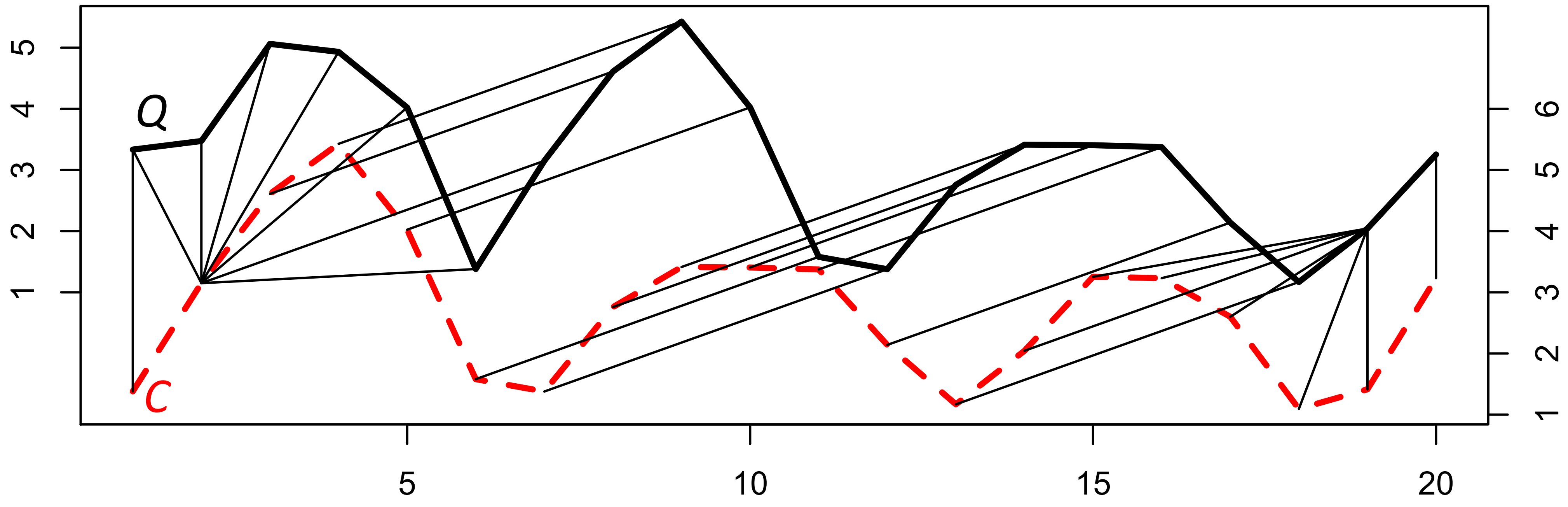




Some real-world example (3/3)



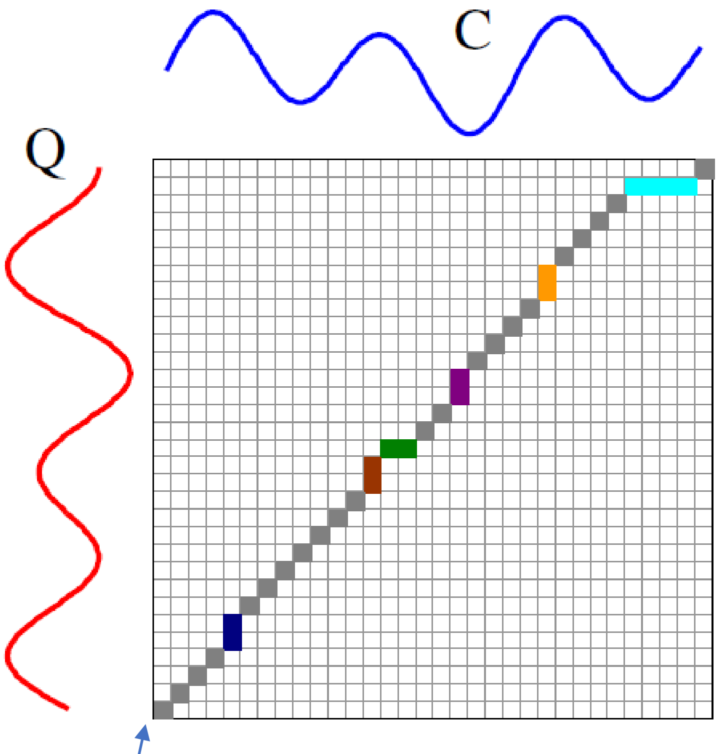
Distance Measure: Dynamic Time Warping (DTW)



Given two time series $Q = q_1 \ldots q_n$ and $C = c_1 \ldots c_n$

We want to find a pairwise points alignment $w \in (\mathbb{N} x \mathbb{N})^n$, wich minimize the pairwise points distance: $\mathrm{DTW}(Q, C) = \underset{P}{\operatorname{argmin}}\left(\sqrt{\sum_{i=1}^{|P|} \left(q_{P_i\,[0]} - c_{P_i\,[1]}\right)^2}\right)$

DTW calculation



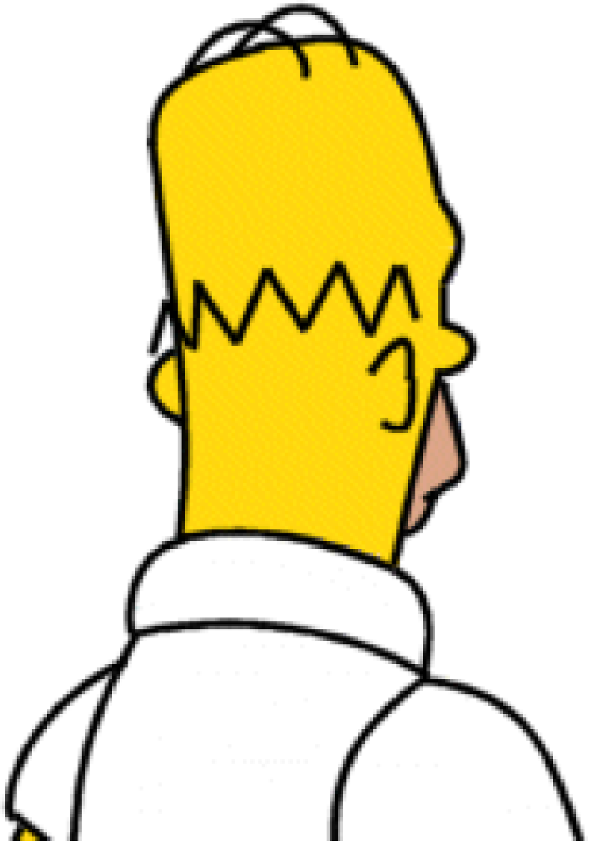
Warping path $P \in (\mathbb{N}x\mathbb{N})^n$

Recursive function of minimum cost path

$$\gamma(P_i) = \sqrt{(q_{P_i\,[0]} - c_{P_i\,[1]})^2 + \min\{\gamma\left(P_{i[0]} - 1\right), \gamma\left(P_{i[1]} - 1\right), \gamma\left(P_{i[0]} - 1\,, P_{i[i]} - 1\right)\}}$$

$P_0 = (0, 0)$ and $P_n = (n, n)$ are always the first and the last element of the warping path rispectively.

Time complexity – $O(n^2)$



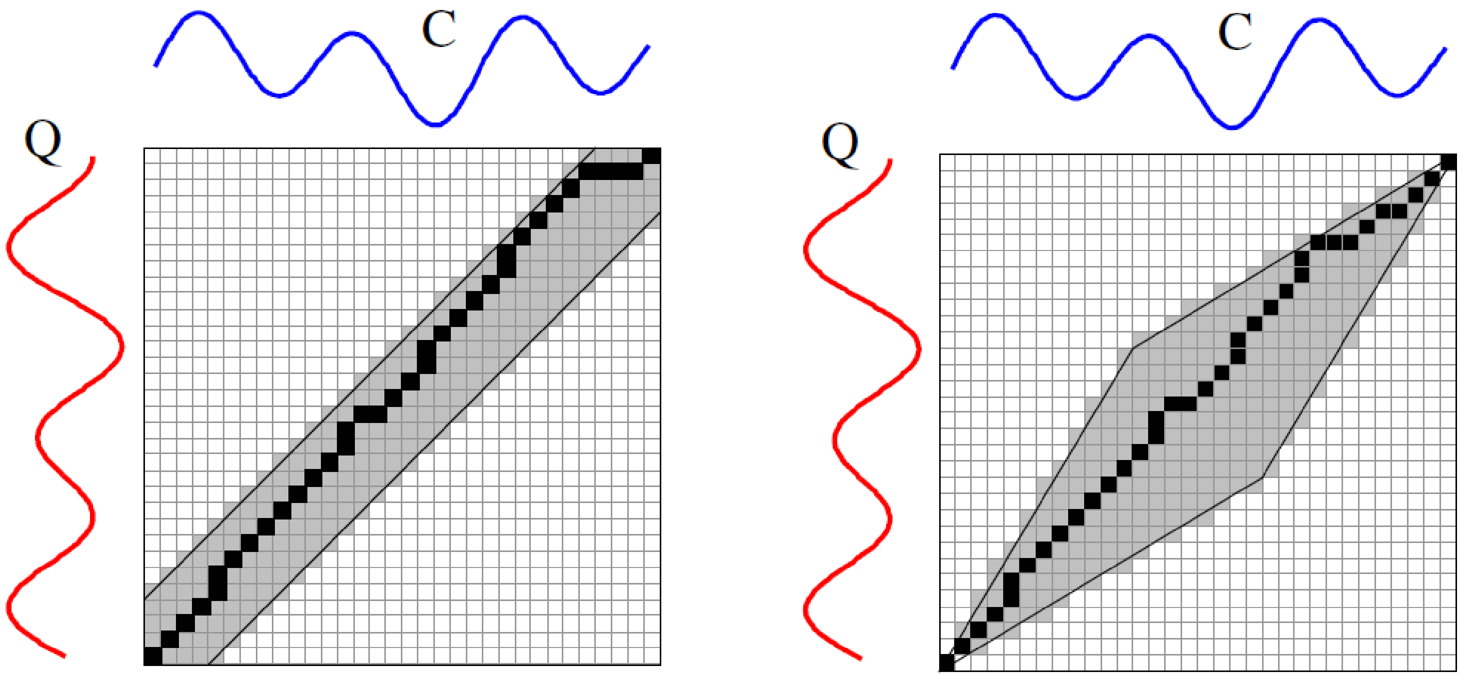
- Dynamic Time Warping gives **much better** results than Euclidean distance on virtually all problems.

- Dynamic Time Warping is very very slow to calculate!

Is there anything we can do to speed up similarity search under DTW?

Global warping path constraints

- Slightly speed up the calculations
- Prevent pathological warpings



Sakoe-Chiba Band

Itakura Parallelogram

Lower bounding

Assume that we have two functions:

- DTW(A,B)
- lower_bound_distance(A,B)

The true DTW function is very slow…

The *lower bound* function is very fast…

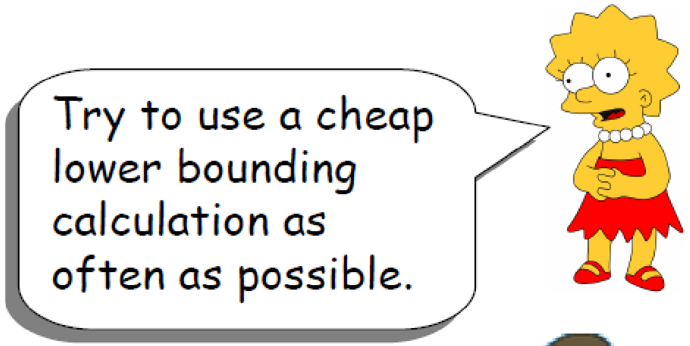
By definition, for all A, B, we have

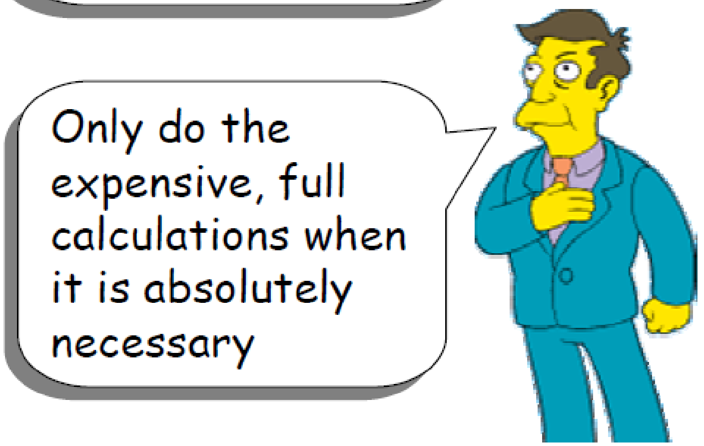
lower_bound_distance(A,B) ≤ DTW(A,B)

Speed up search with lower bounding

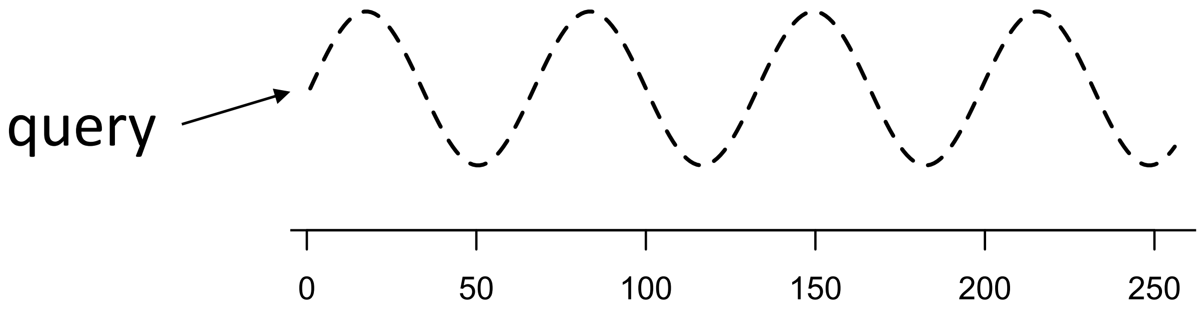
We can speed up similarity search under DTW
by using a lower bounding function

```
Algorithm Lower_Bounding_Sequential_Scan(Q)
best_so_far = infinity;
for all sequences in database
  LB_dist = lower_bound_distance(C_i, Q);
   if LB_dist < best_so_far
     true_dist = DTW(C_i, Q);
     if true_dist < best_so_far
       best_so_far = true_dist;
       index_of_best_match = i;
     endif
   endif
endfor
```

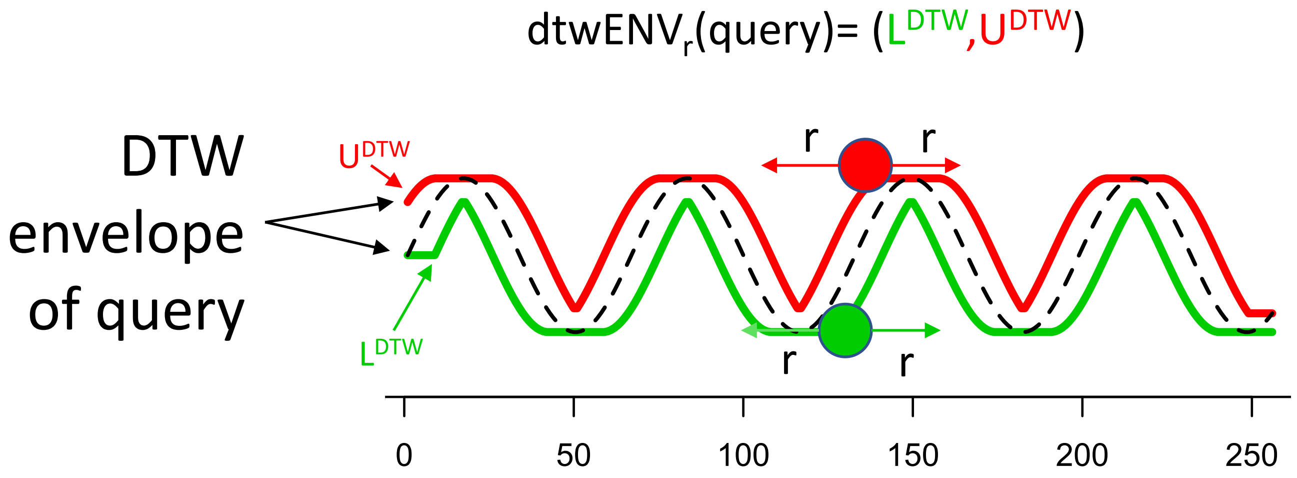




Lower bounding DTW



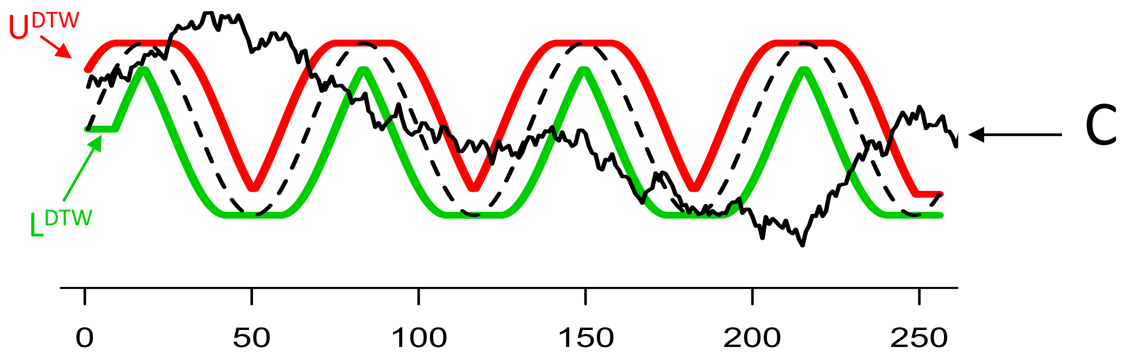
Lower bounding DTW



- For each query position, dtwENV bounds the values that can be aligned to a single point in the data series candidates (warping window r).

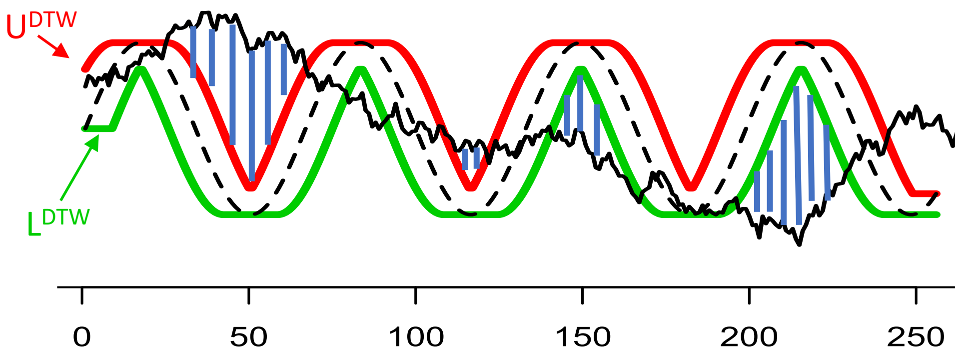
Lower bounding DTW

LB_Keogh(dtwENV$_r$(query), C)



Lower bounding DTW

LB_Keogh(dtwENV$_r$(query), candidate)



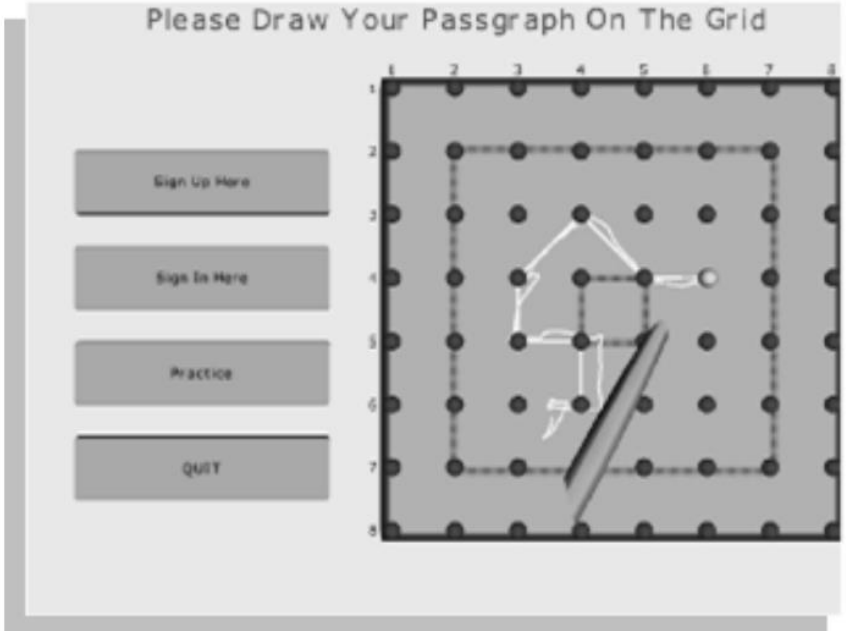
- Lower bounding of the true DTW distance between Query and Candidate

[O(n) time]

Presentation du TD

Haptics DATA

Data are taken from 5 people entering their passgraph (a code to access a system protected by a graphical authentication system) on a touchscreen. The data are the x-axis movement only.



Notebook

- Open the file (Python Notebook):

TS_SimilaritySearch.ipynb

- Instruction are contained in the notebook

References

- Eamonn J. Keogh. A Decade of Progress in Indexing and Mining Large Time Series Databases. VLDB 2006: 1268
- Eamonn J. Keogh, Li Wei, Xiaopeng Xi, Sang-Hee Lee, Michail Vlachos: **LB_Keogh Supports Exact Indexing of Shapes under Rotation Invariance with Arbitrary Representations and Distance Measures. VLDB 2006: 882-893**
- Chotirat Ann Ralanamahatana, Jessica Lin, Dimitrios Gunopulos, Eamonn Keogh, Michail Vlachos & Gautam Das **Mining Time Series Data https://link.springer.com/chapter/10.1007/0-387-25465-x_51#Abs1**