

15-826: Multimedia (Databases) and Data Mining

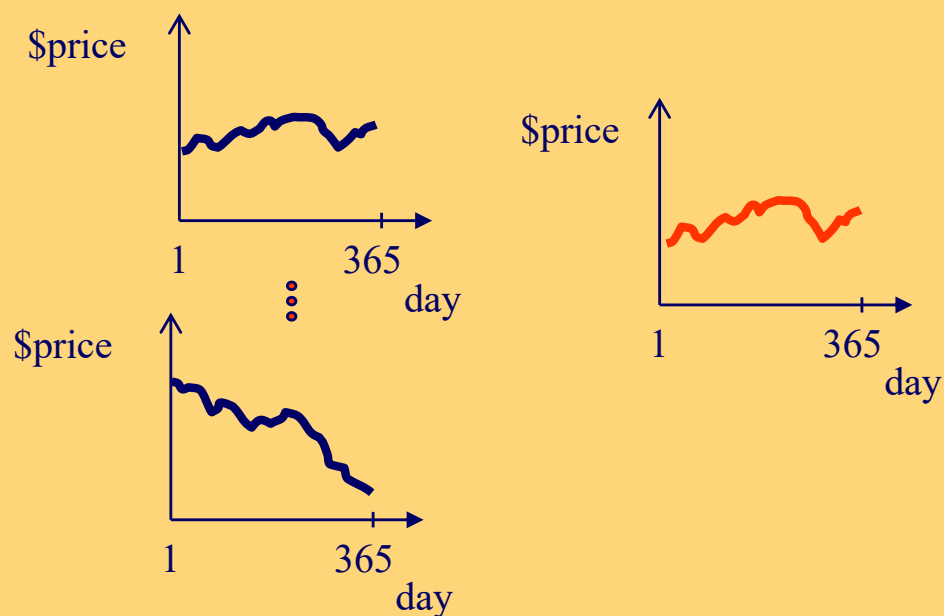
Lecture #23: Multimedia indexing

C. Faloutsos



Problem

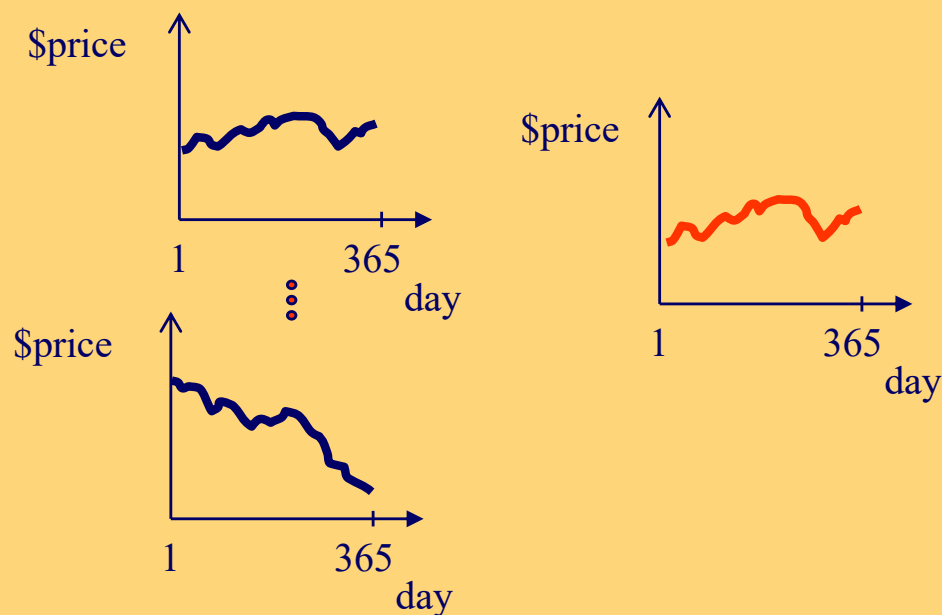
- Q: Find stocks similar to $\langle \text{MSFT} \rangle$





Solution

- Q: Find stocks similar to $\langle \text{MSFT} \rangle$
- A: GEMINI: Extract features + SAM



Must-read Material

- [MM Textbook](#), chapters 7, 8, 9 and 10.
- Myron Flickner, et al: [Query by Image and Video Content: the QBIC System](#) IEEE Computer 28, 9, Sep. 1995, pp. 23-32.
- [Journal of Intelligent Inf. Systems, 3, 3/4, pp. 231-262, 1994](#) (An earlier, more technical version of the IEEE Computer '95 paper.)
- FastMap: [Textbook](#) chapter 11; Also in: C. Faloutsos and K.I. Lin *FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets* ACM SIGMOD 95, pp. 163-174.

Must-read material

- ★ • Laurens van der Maaten, Geoffrey Hinton, [*Visualizing Data using t-SNE*](#), JMLR 9(86):2579–2605, 2008 ([scikit-learn](#) :

```
from sklearn.manifold import TSNE
```

- ★ • Leland McInnes, John Healy, James Melville, [*UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*](#), arxiv, 2018 ([python library](#))



Outline

Goal: ‘Find **similar / interesting** things’

- Intro to DB
- ➔ • Indexing - similarity search
- Data Mining



Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
- fractals
- text
- Singular Value Decomposition (SVD)
- Multimedia
 - DSP
 - indexing



• ...
15-826



Multimedia - Detailed outline

- Multimedia indexing
 - ➔ – Motivation / problem definition
 - Main idea / time sequences
 - images
 - sub-pattern matching
 - automatic feature extraction / FastMap

Problem

Given a large collection of (multimedia)
records (eg. stocks)

Allow fast, similarity queries

Applications

- time series: financial, marketing (click-streams!), ECGs, sound;
- images: medicine, digital libraries, education, art
- higher-d signals: scientific db (eg., astrophysics), medicine (MRI scans), entertainment (video)

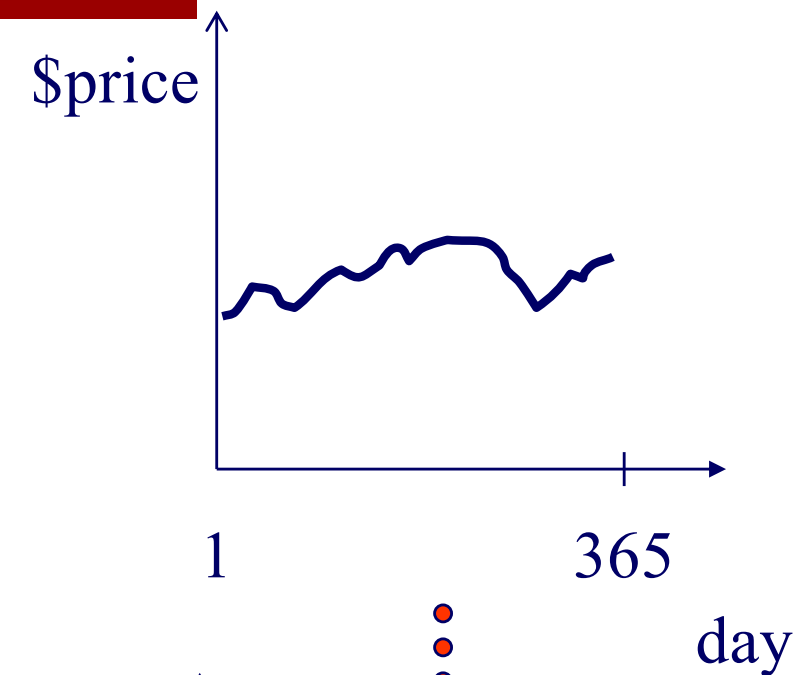
Sample queries

- find medical cases similar to Smith's
- Find pairs of stocks that move in sync
- Find pairs of documents that are similar (plagiarism?)
- find faces similar to 'Tiger Woods'

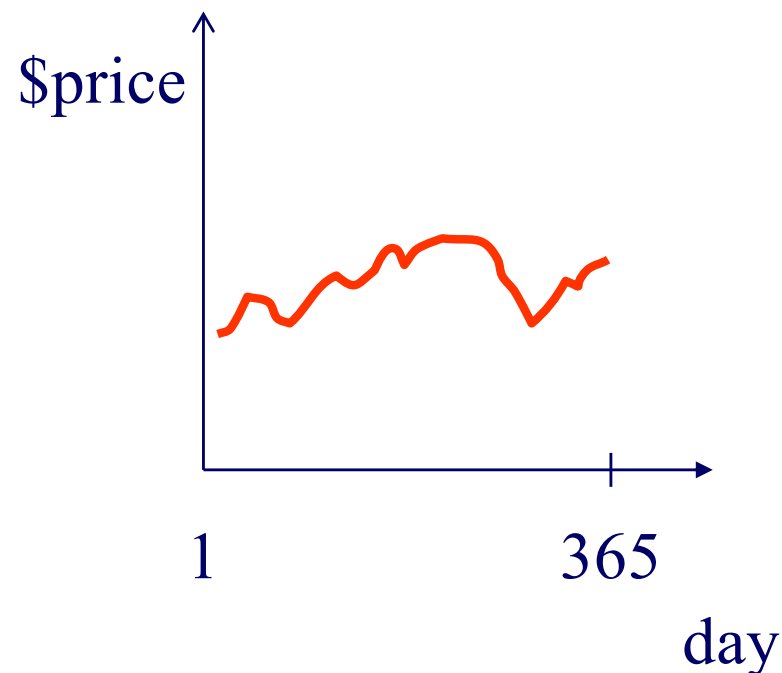
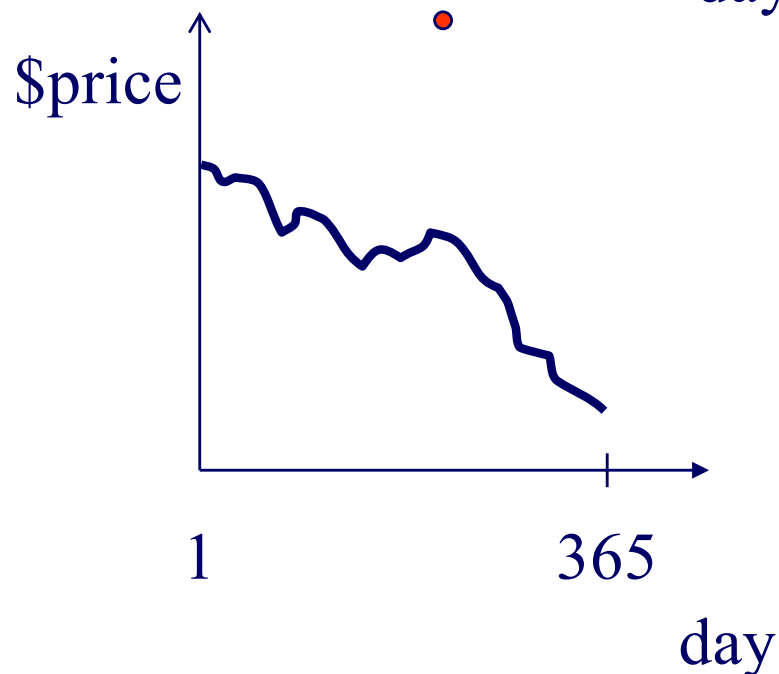
Detailed problem defn.:

Problem:

- given a set of multimedia objects,
 - find the ones similar to a desirable query object
-
- for example:



⋮



distance function: by **expert**
(eg, Euclidean distance)

Types of queries

- whole match vs sub-pattern match
- range query vs nearest neighbors
- all-pairs query

Design goals

- Fast (faster than seq. scan)
- ‘correct’ (ie., no false alarms; no false dismissals)

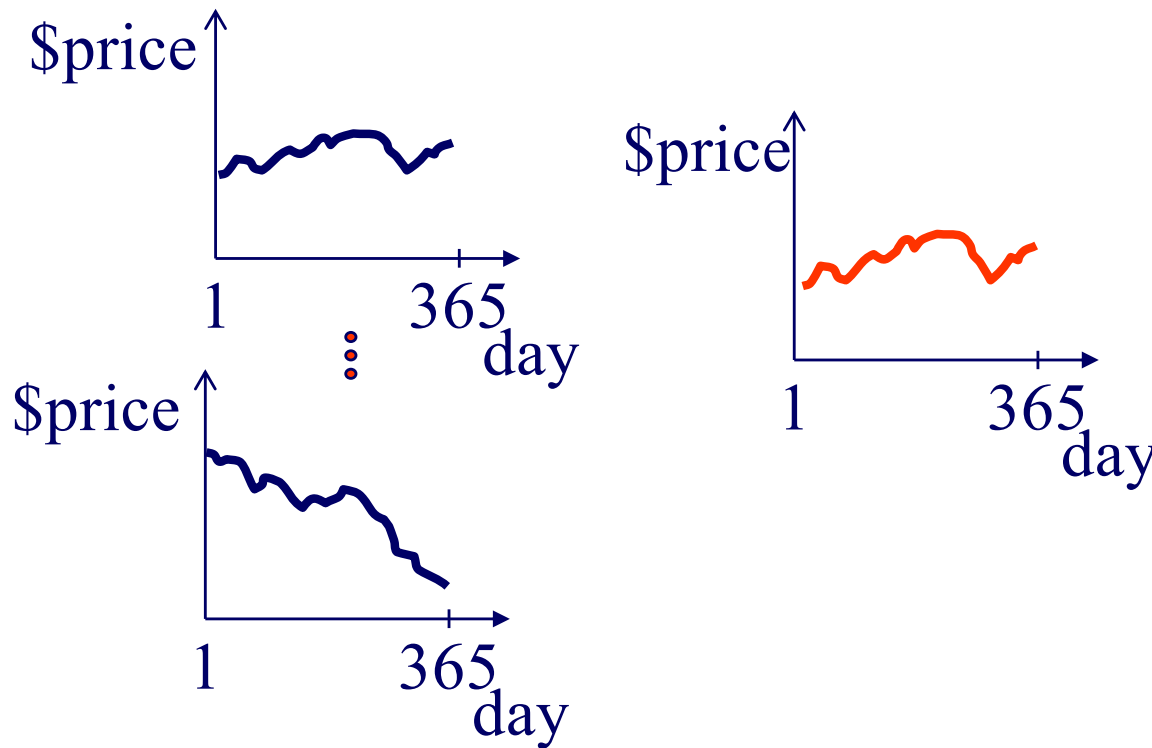


Multimedia - Detailed outline

- multimedia
 - Motivation / problem definition
 - ➔ – Main idea / time sequences
 - images
 - sub-pattern matching
 - automatic feature extraction / FastMap

Main idea

- Eg., time sequences, ‘whole matching’, range queries, Euclidean distance



Main idea

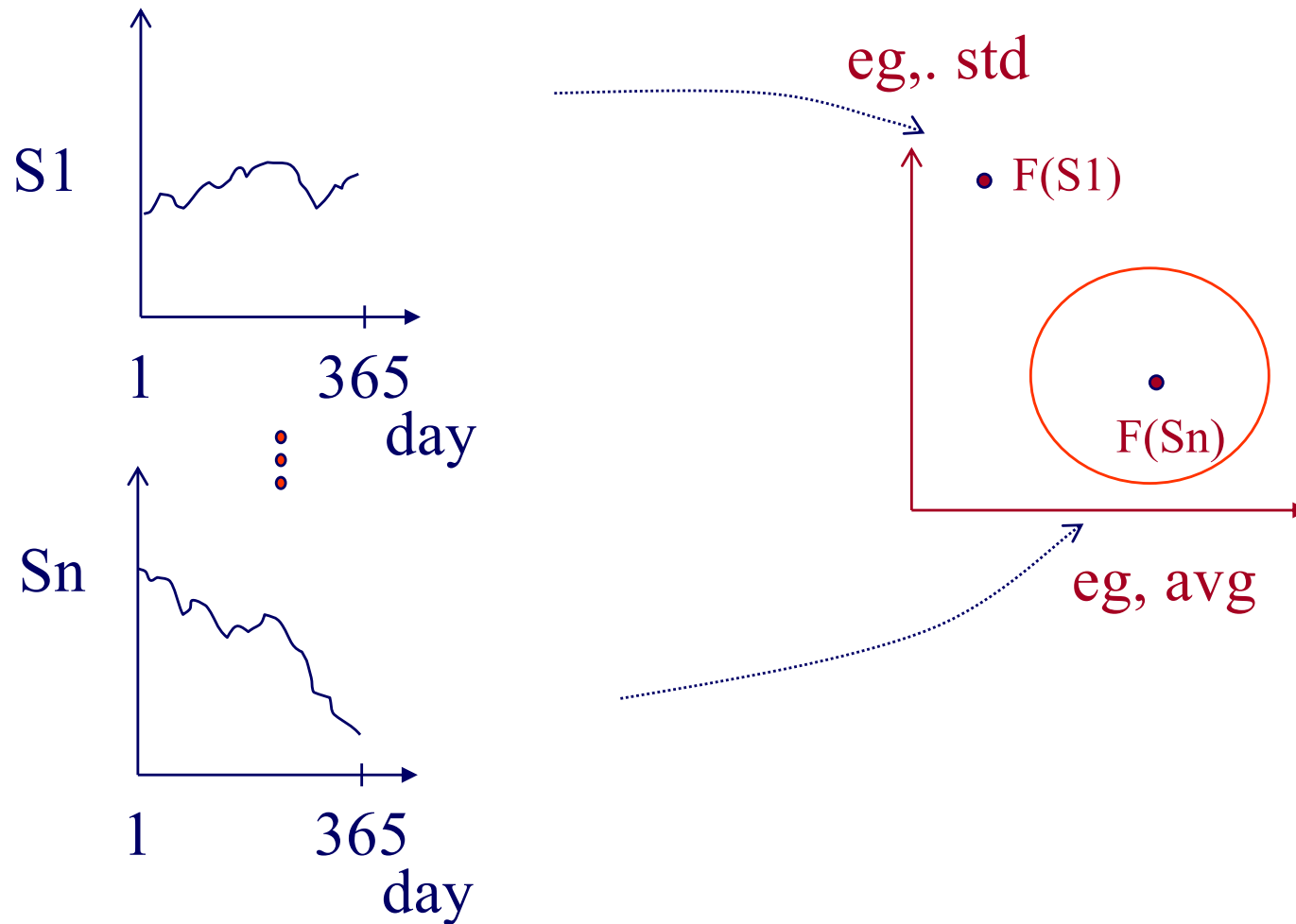
- Seq. scanning works - how to do faster?

Idea: 'GEMINI'

(GEneric Multimedia INdexIng)

Extract a few numerical features, for a 'quick and dirty' test

‘GEMINI’ - Pictorially



GEMINI

Solution: Quick-and-dirty' filter:

- extract n features (numbers, eg., avg., etc.)
- map into a point in n -d feature space
- organize points with off-the-shelf spatial access method ('SAM')
- discard false alarms

GEMINI

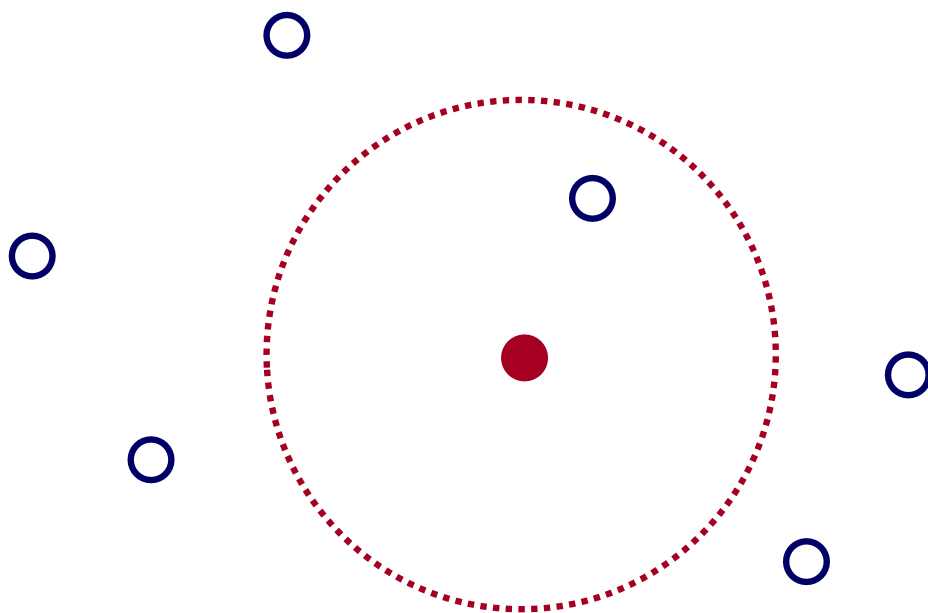
Important: Q: how to guarantee no false dismissals?

A1: preserve distances (but:
difficult/impossible)

A2: **Lower-bounding lemma**: if the mapping
‘makes things look closer’, then there are
no false dismissals

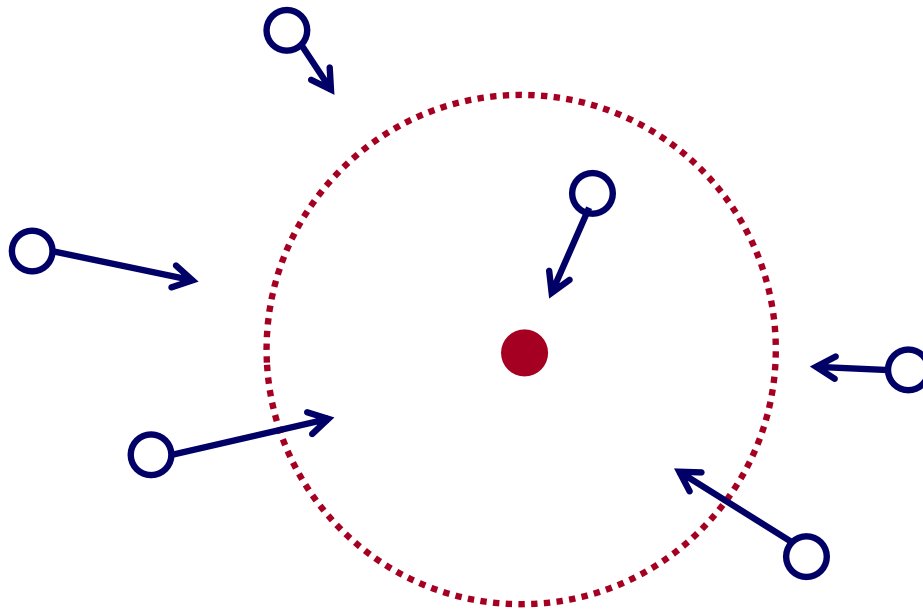
GEMINI

- ‘proof’ of lower-bounding lemma



GEMINI

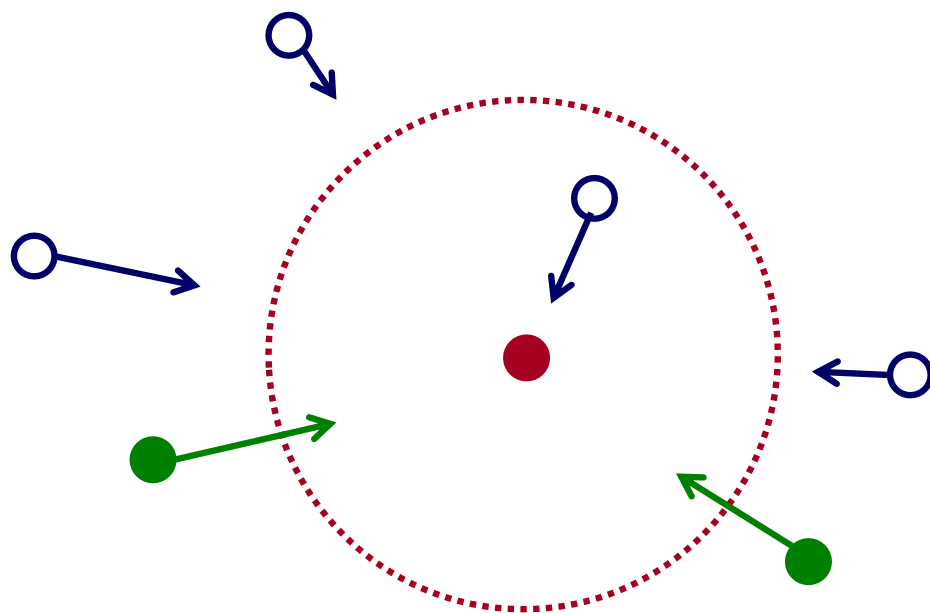
- ‘proof’ of lower-bounding lemma



Lower-bounding:
Makes objects
look closer to each
other (& to query
object)

GEMINI

- ‘proof’ of lower-bounding lemma



Lower-bounding:
Makes objects
look closer to each
other (& to query
object)
-> ONLY **false**
alarms

GEMINI

Important:

Q: how to extract features?

A: *“if I have only one number to describe my object, what should this be?”*

Time sequences

Q: what features?

Time sequences

Q: what features?

A: Fourier coefficients (we'll see them in detail soon)

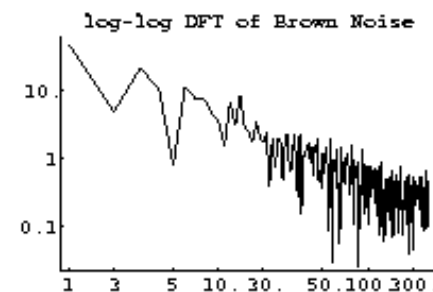
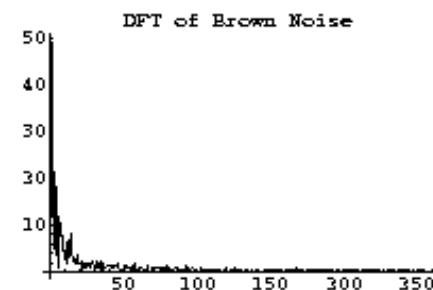
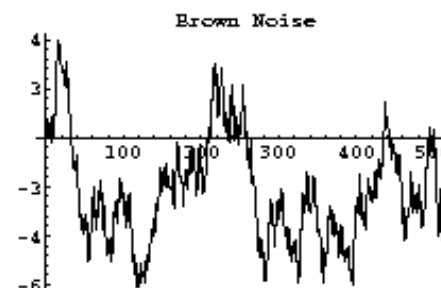
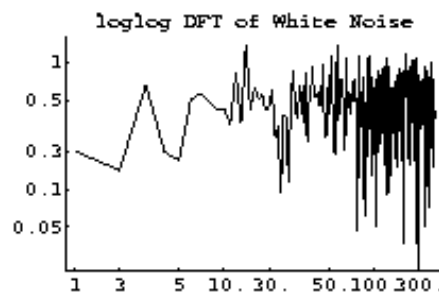
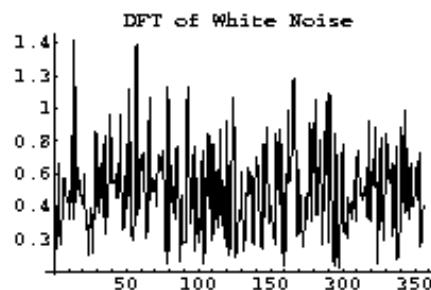
Time sequences

white noise

brown noise

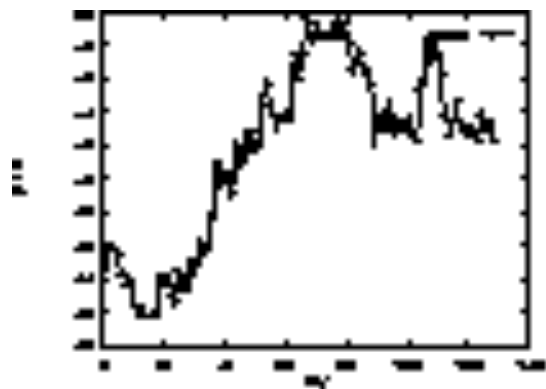
Fourier
spectrum

... in log-log

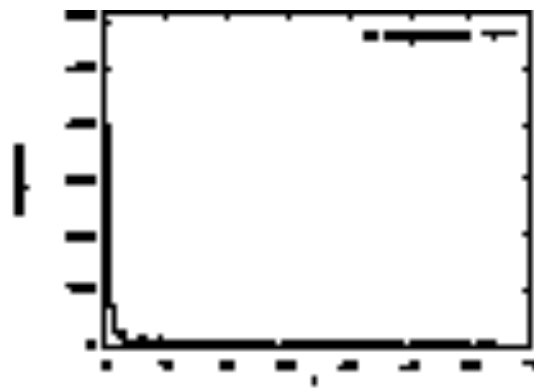


Time sequences

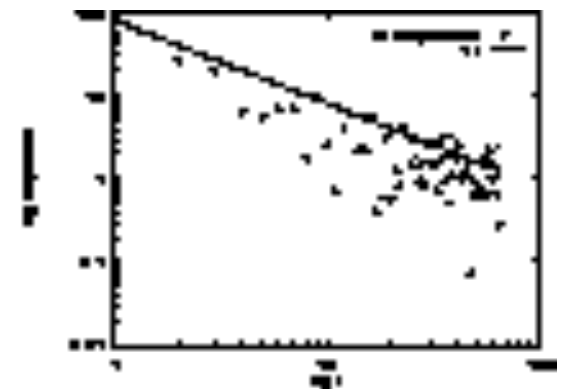
- Eg.:



(a) IBM stock



(b) spectrum
(linear scales)



(c) spectrum
(log scales)

Time sequences

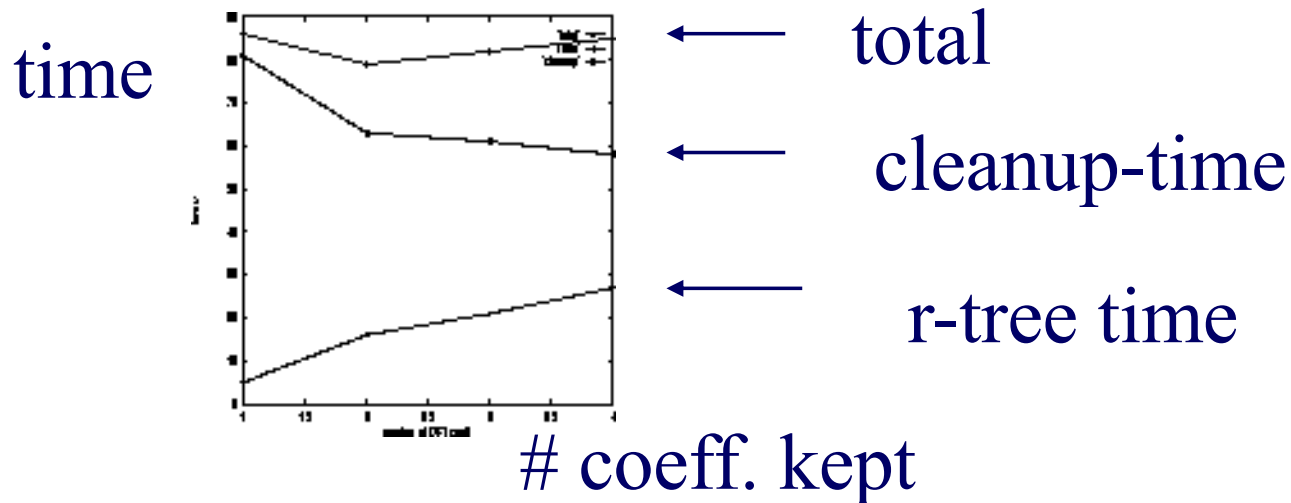
- conclusion: colored noises are well approximated by their first few Fourier coefficients
- colored noises appear in nature:

Time sequences

- brown noise: stock prices ($1/f^2$ energy spectrum)
- pink noise: works of art ($1/f$ spectrum)
- black noises: water reservoirs ($1/f^b$, $b > 2$)
- (slope: related to ‘Hurst exponent’, for self-similar traffic, like, eg. Ethernet/web [Schroeder], [Leland+])

Time sequences - results

- keep the first 2-3 Fourier coefficients
- faster than seq. scan
- NO false dismissals (see book)



Time sequences - improvements:

- improvements/variations:
[Kanellakis+Goldin], [Mendelzon+Rafiei]
- could use Wavelets, or DCT
- could use segment averages [Yi+2000]



Multimedia - Detailed outline

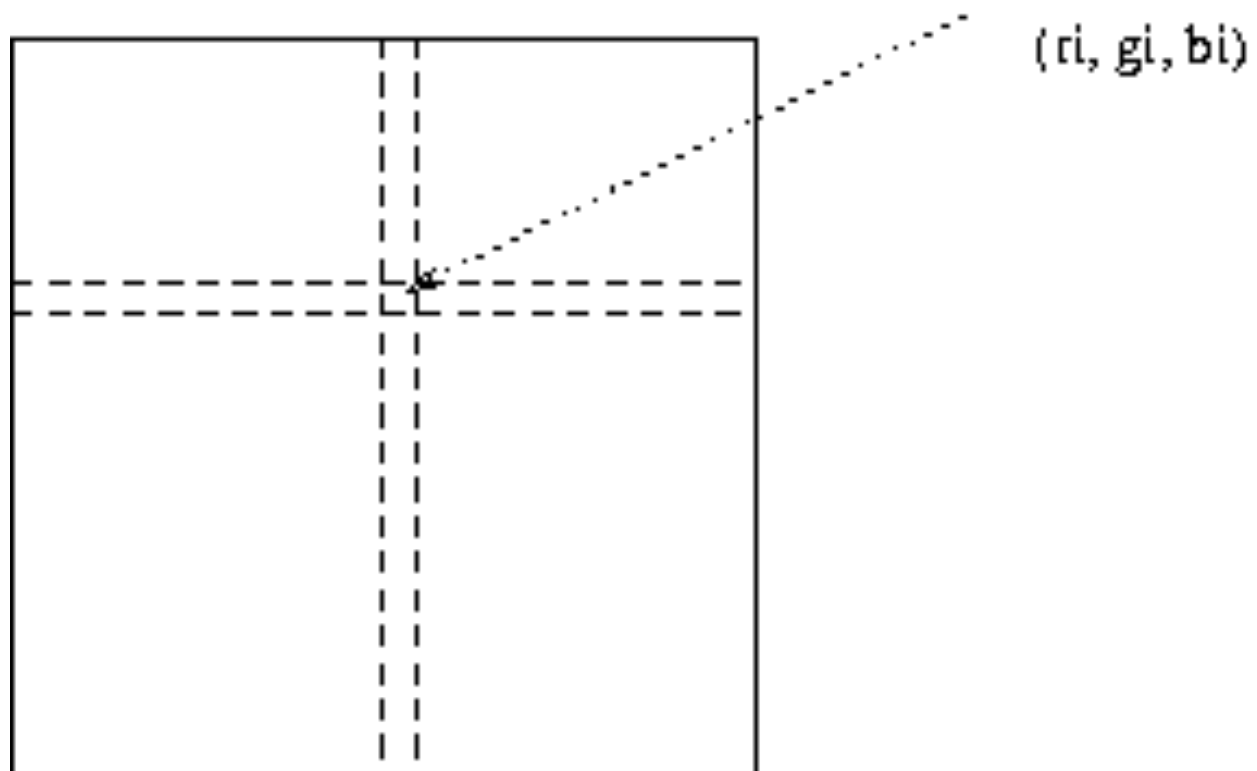
- multimedia
 - Motivation / problem definition
 - Main idea / time sequences
 - ➔ – images (color, shapes)
 - sub-pattern matching
 - automatic feature extraction / FastMap

Images - color

COLOR IMAGE, eg. 256x256

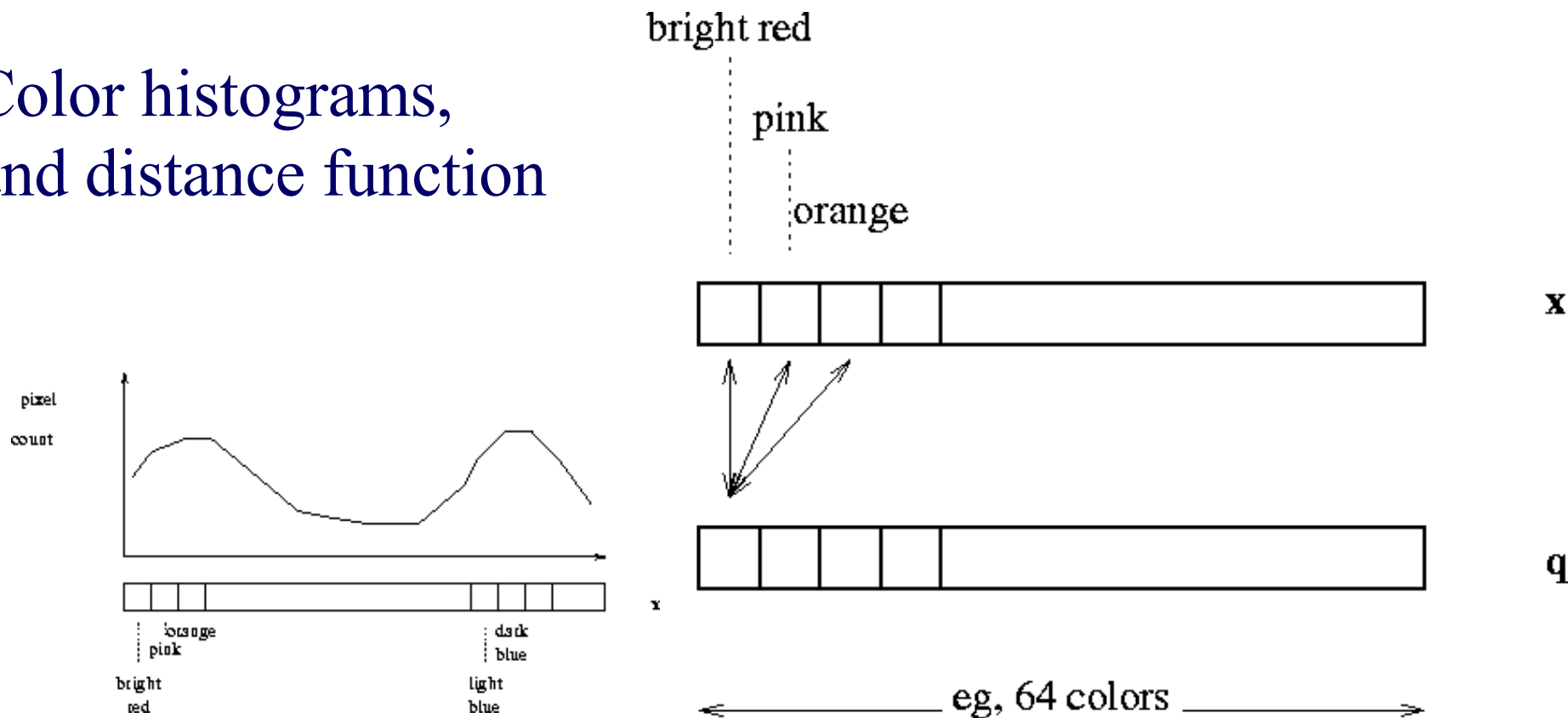
what is an image?

A: 2-d array



Images - color

Color histograms,
and distance function

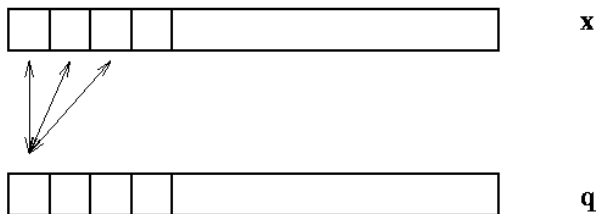


Images - color

Mathematically, the distance function is:

$$distance_{histogram}(\vec{x}, \vec{q}) = (\vec{x} - \vec{q}) \begin{bmatrix} a_{RR} & a_{RP} & \dots \\ a_{PR} & a_{PP} & \dots \\ \dots & \dots & \dots \end{bmatrix} (\vec{x} - \vec{q})^t$$

$$\dots = (\vec{x} - \vec{q}) \mathcal{A} (\vec{x} - \vec{q})^t$$



Images - color

Problem: 'cross-talk' :

- Features are not orthogonal ->
- SAMs will not work properly
- Q: what to do?
- A: feature-extraction question

Images - color

possible answers:

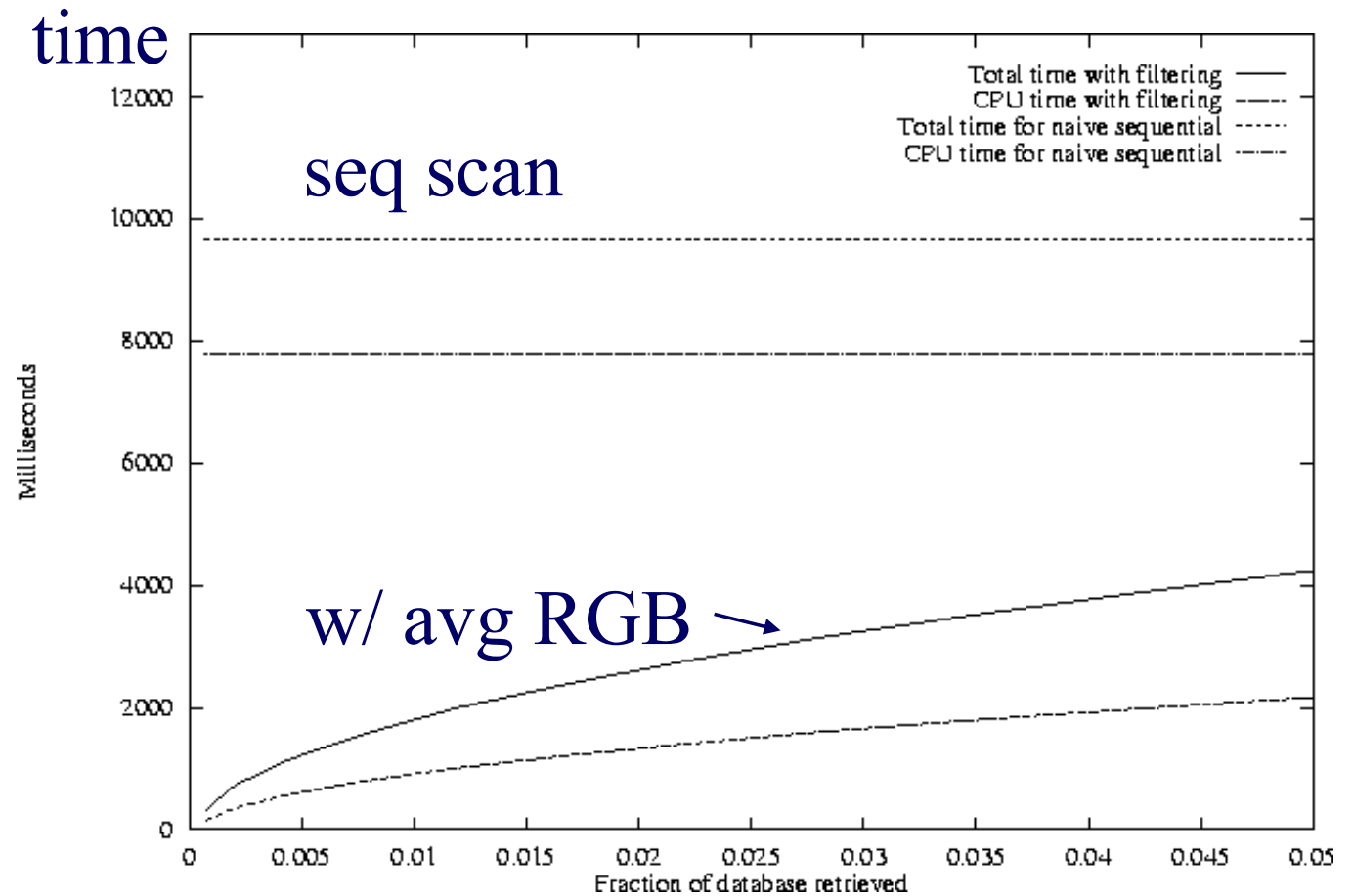
- avg red, avg green, avg blue

it turns out that this lower-bounds the histogram distance ->

- no cross-talk
- SAMs are applicable

Images - color

performance: time



selectivity
#41



Multimedia - Detailed outline

- multimedia
 - Motivation / problem definition
 - Main idea / time sequences
 -  – images (color; shape)
 - sub-pattern matching
 - automatic feature extraction / FastMap

Images - shapes

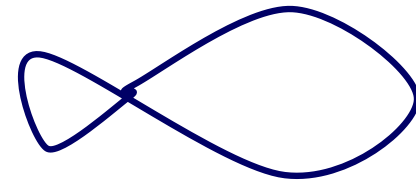
- distance function: Euclidean, on the area, perimeter, and 20 'moments'
- (Q: how to normalize them?)

Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
- (Q: how to normalize them?)
- A: divide by standard deviation)

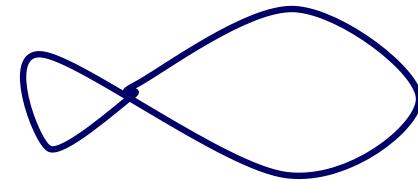
Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
- (Q: other ‘features’ / distance functions?)



Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
- (Q: other ‘features’ / distance functions?
- A1: turning angle
- A2: dilations/erosions
- A3: ...)



Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 'moments'
- Q: how to do dim. reduction?

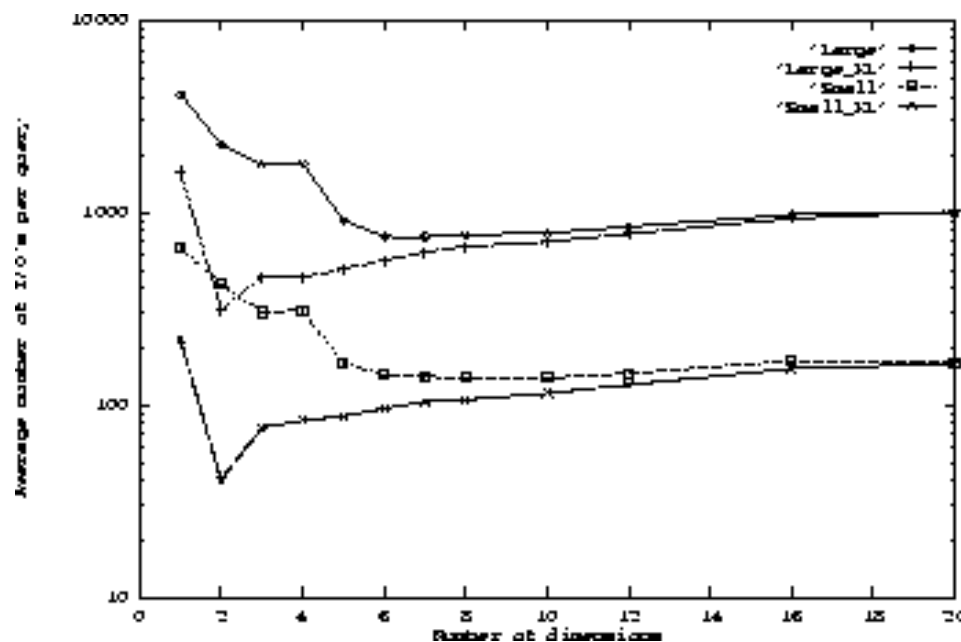
Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
- Q: how to do dim. reduction?
- A: Karhunen-Loeve (= centered PCA/SVD)

Images - shapes

- Performance: $\sim 10\times$ faster

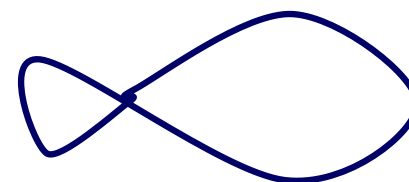
$\log(\# \text{ of I/Os})$



← all kept

of features kept

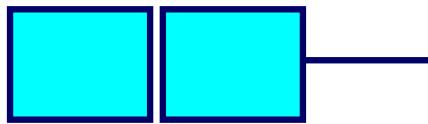
Other shape features?



Other shape features

- Morphology (dilations, erosions, openings, closings) [Korn+, VLDB96]

shape



“structuring
element”

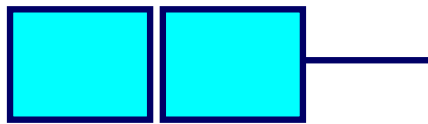
$R=1$



Other shape features

- Morphology (dilations, erosions, openings, closings) [Korn+, VLDB96]

shape



“structuring
element”

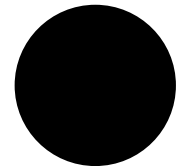
$R=0.5$



$R=1$



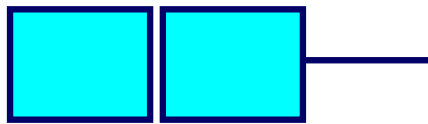
$R=2$



Other shape features

- Morphology (dilations, erosions, openings, closings) [Korn+, VLDB96]

shape



“structuring element”

$R=0.5$



$R=1$

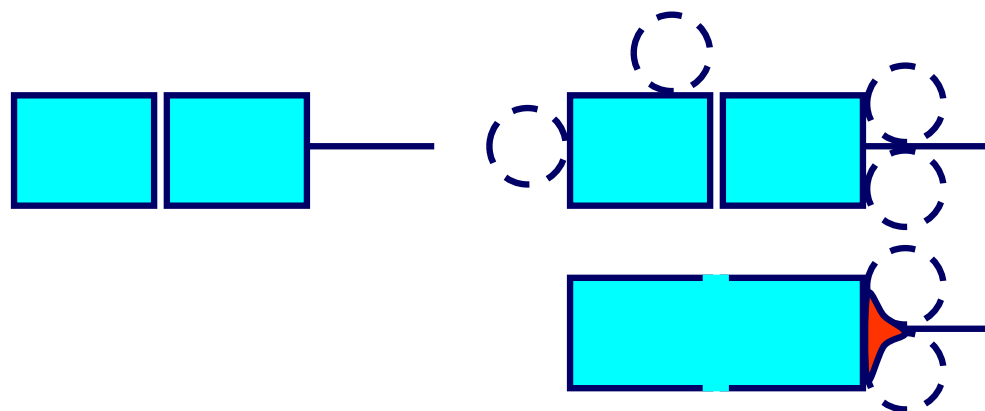


$R=2$



Morphology: closing

- fill in small gaps
- **very similar** to ‘alpha contours’



Morphology: closing

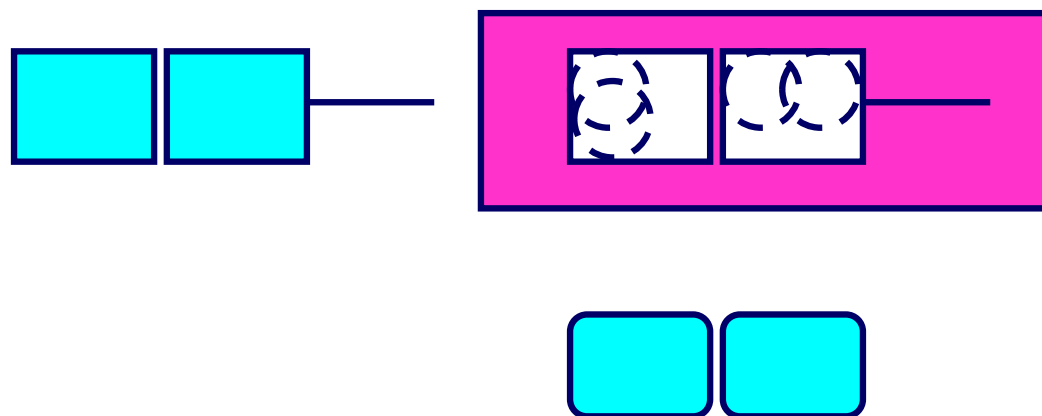
- fill in small gaps



‘closing’,
with $R=1$

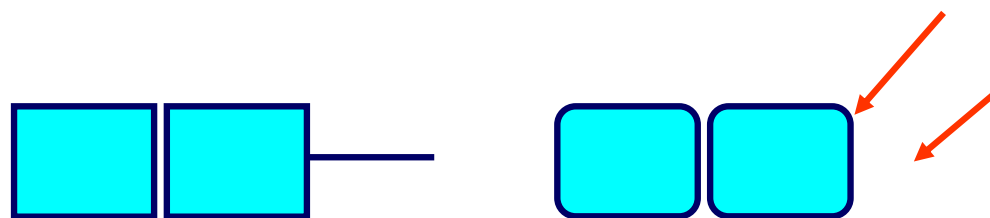
Morphology: opening

- ‘closing’, for the complement =
- trim small extremities



Morphology: opening

- ‘closing’, for the complement =
- trim small extremities



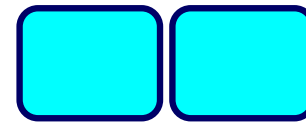
‘opening’
with $R=1$

Morphology

- Closing: fills in gaps



- Opening: trims extremities

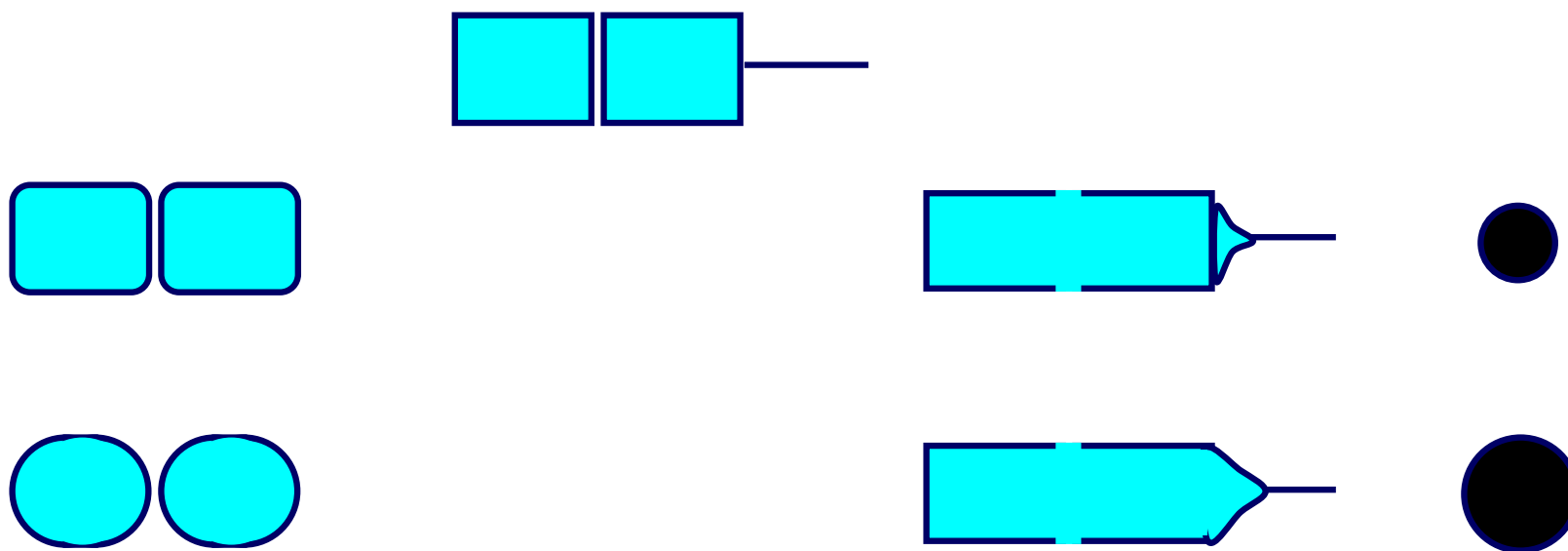


- All wrt a structuring element:



Morphology

- Features: areas of openings ($R=1, 2, \dots$) and closings



Morphology

- Powerful method:
- ‘pattern spectrum’ [Maragos+]
- ‘skeletonization’ of images
- ‘Alpha-shapes’ [Edelsbrunner]
- Book: *An introduction to morphological image processing*, by Edward R. Dougherty

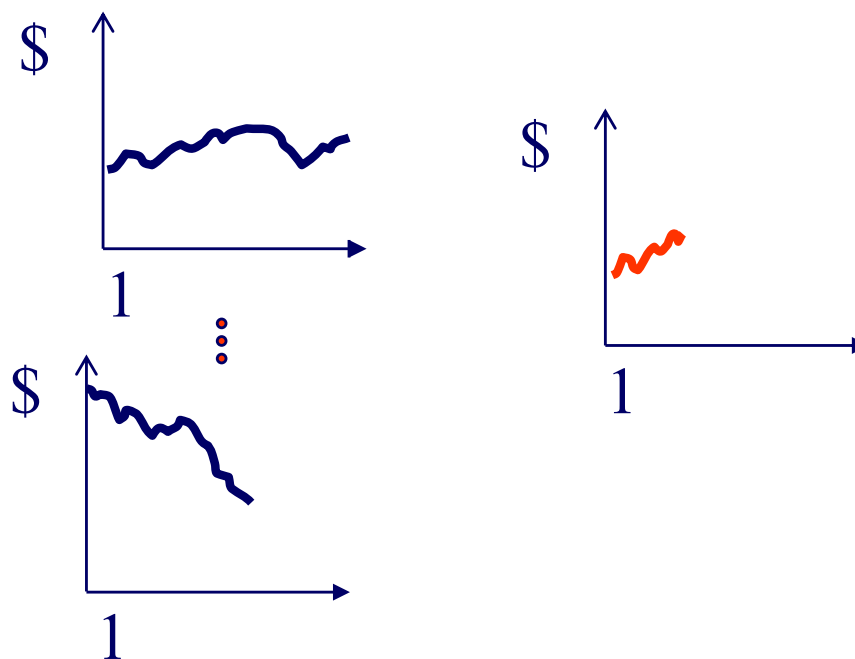


Multimedia - Detailed outline

- multimedia
 - Motivation / problem definition
 - Main idea / time sequences
 - images (color; shape)
 - ➔ – sub-pattern matching
 - automatic feature extraction / FastMap

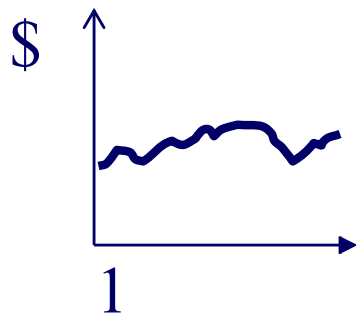
Sub-pattern matching

- Problem: find **sub**-sequences that match the given query pattern



Sub-pattern matching

- Q: how to proceed?
- Hint: try to turn it into a ‘whole-matching’ problem (how?)



Sub-pattern matching

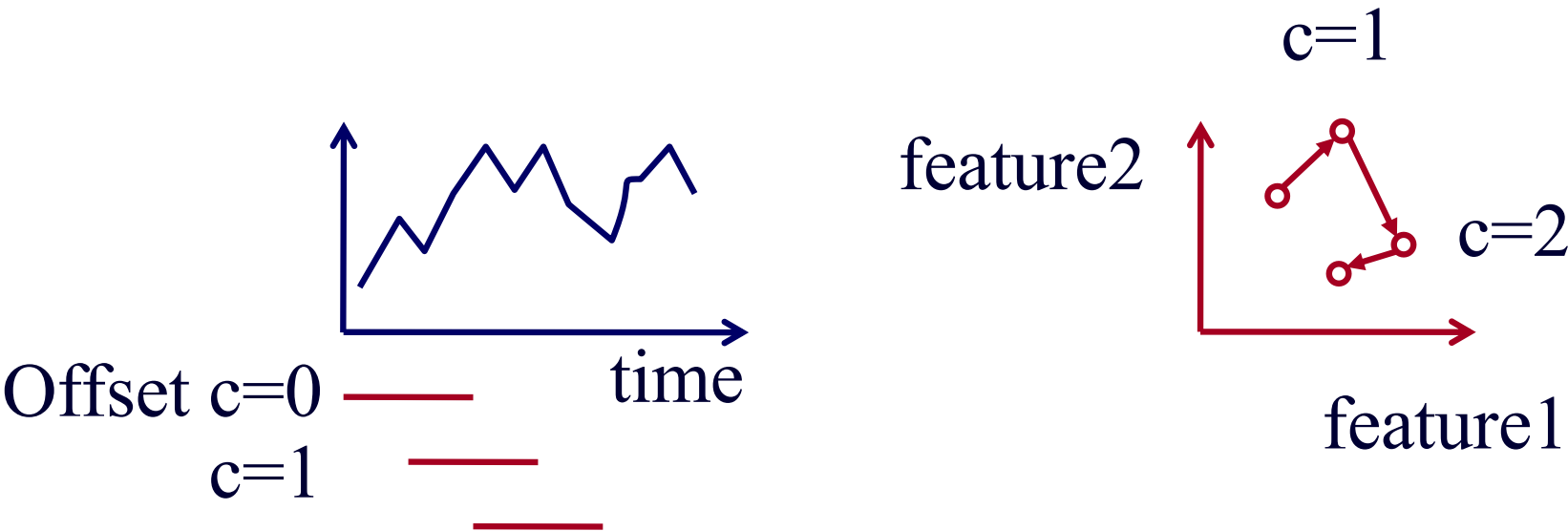
- Assume that queries have minimum duration w ; (eg., $w=7$ days)
- divide data sequences into windows of width w (overlapping, or not?)

Sub-pattern matching

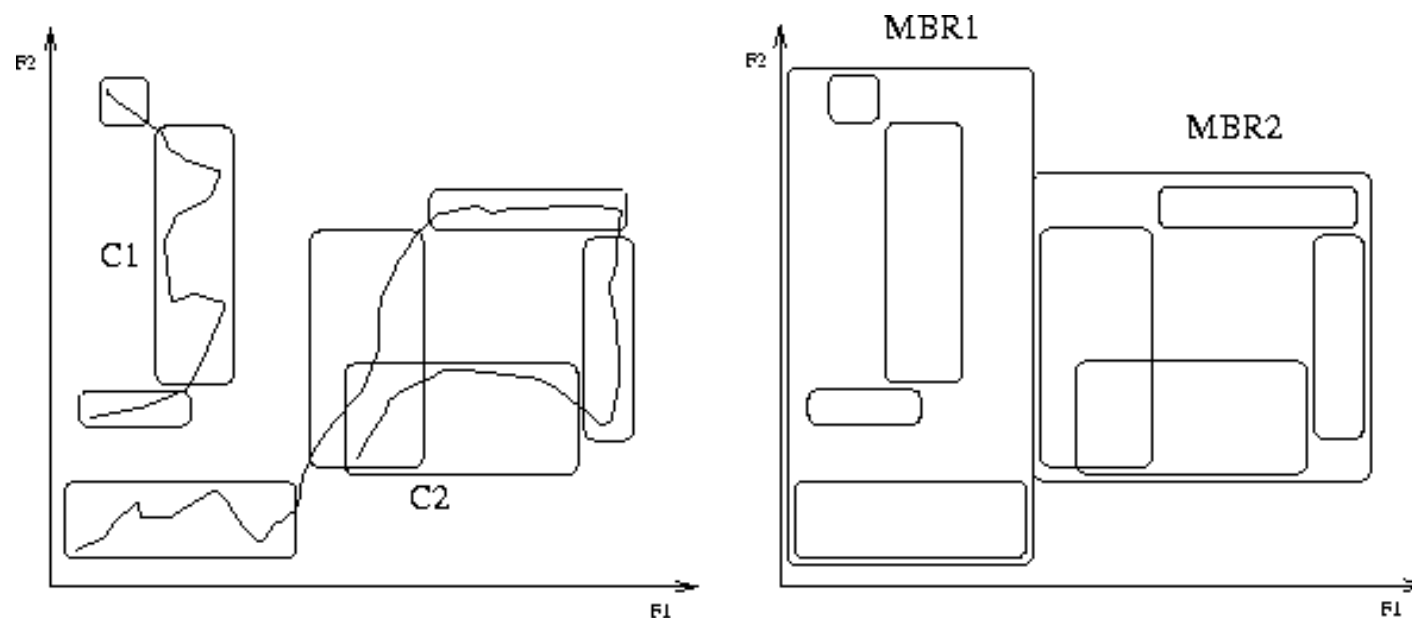
- Assume that queries have minimum duration w ; (eg., $w=7$ days)
- divide data sequences into windows of width w (overlapping, or not?)
- A: sliding, overlapping windows. Thus: trails

Pictorially:

Sub-pattern matching

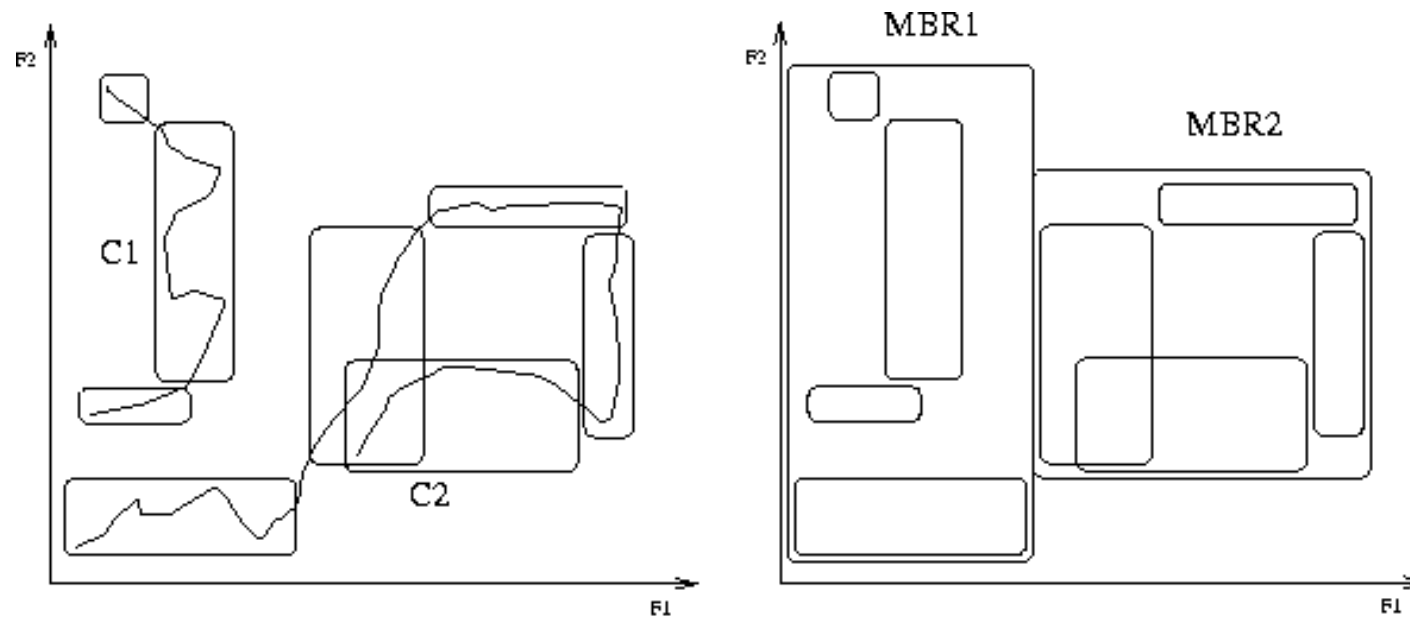


Sub-pattern matching



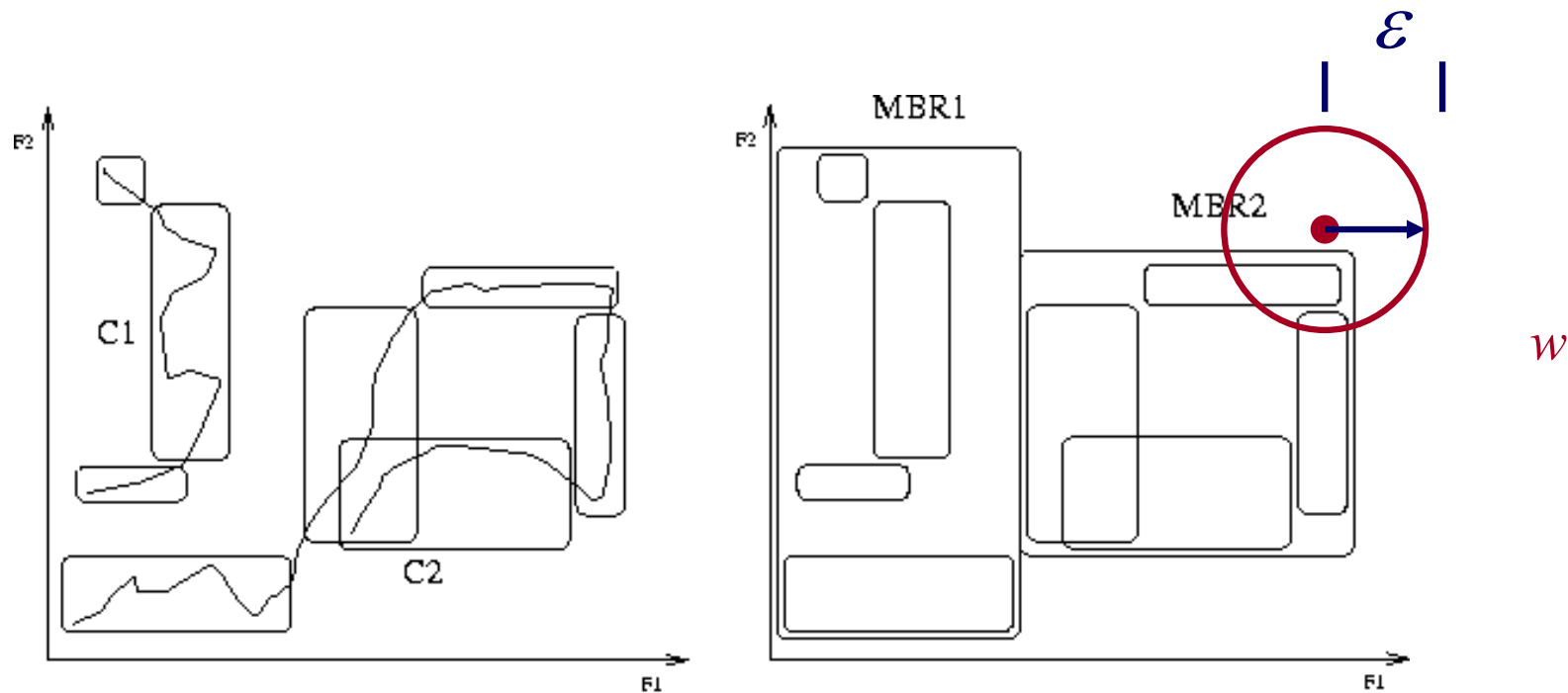
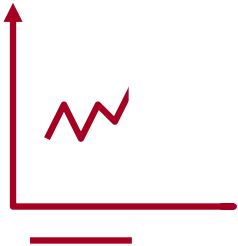
sequences $\rightarrow$ trails $\rightarrow$ MBRs in feature space

Sub-pattern matching



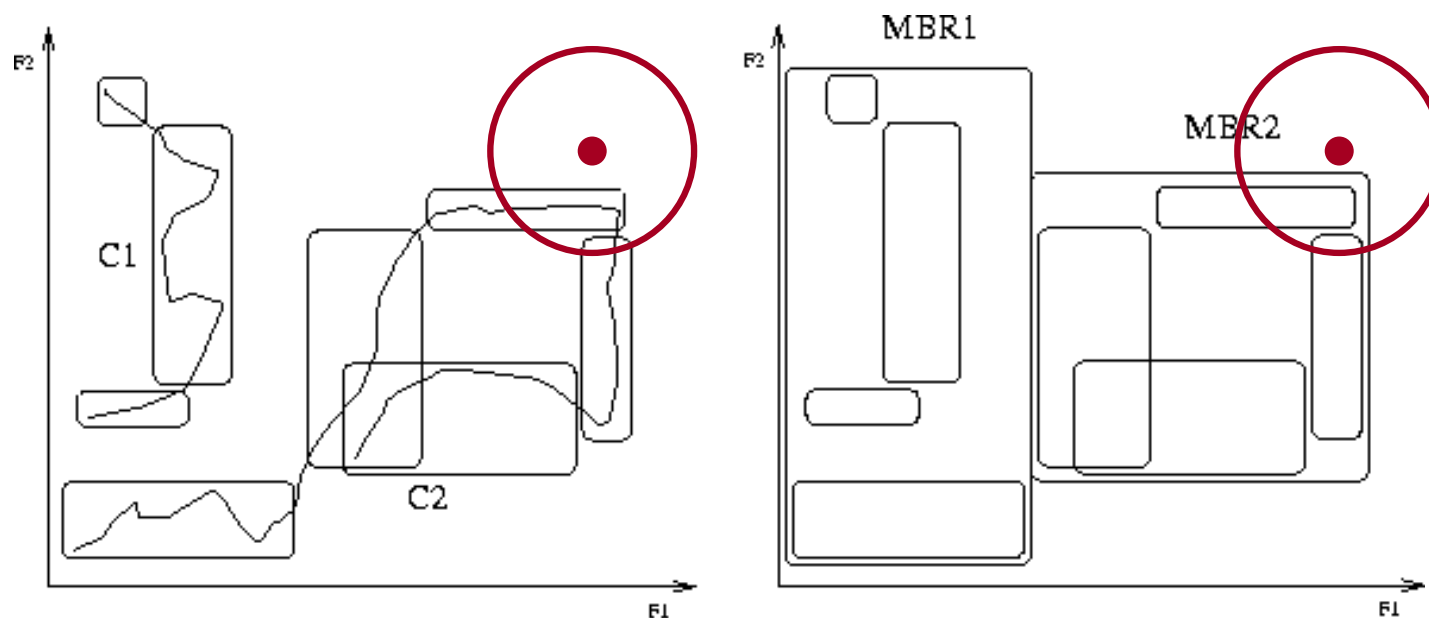
Q: do we store all points? why not?

Sub-pattern matching



Q: how to do range queries of duration w ?

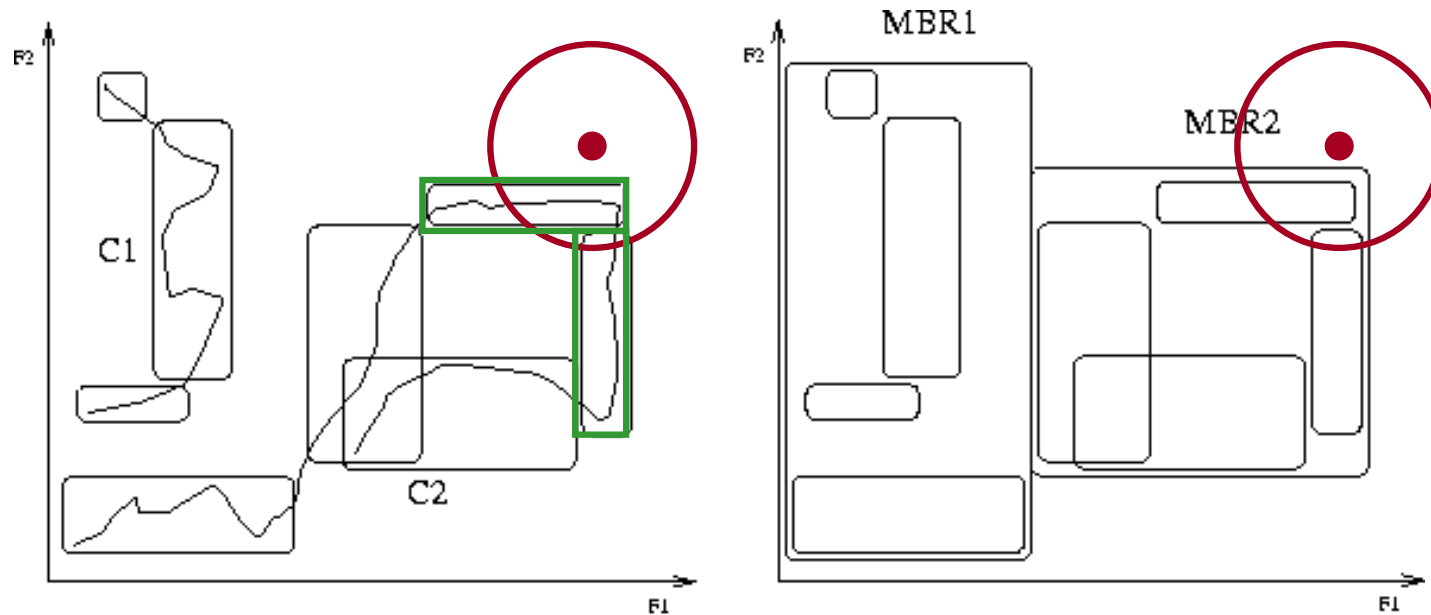
Sub-pattern matching



Q: how to do range queries of duration w ?

A: R-tree; find qualifying stocks and intervals

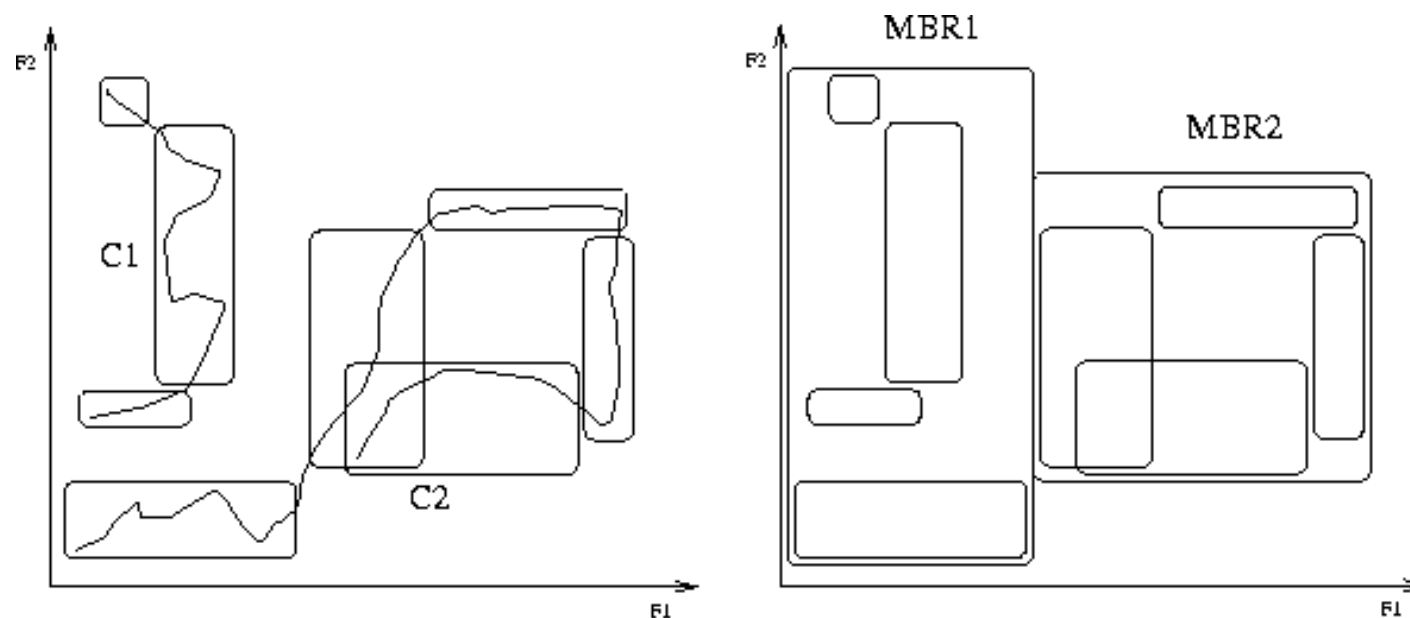
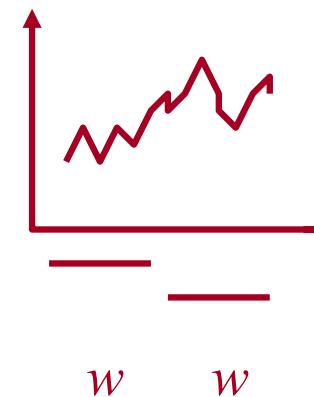
Sub-pattern matching



Q: how to do range queries of duration w ?

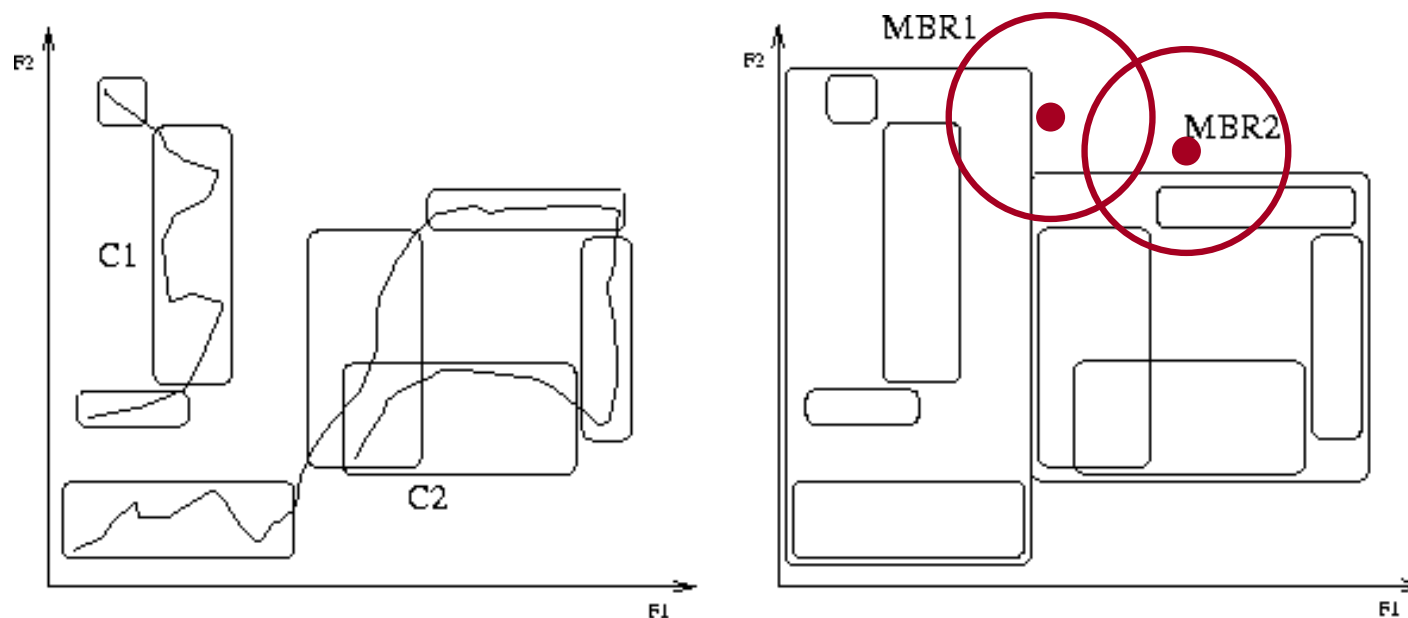
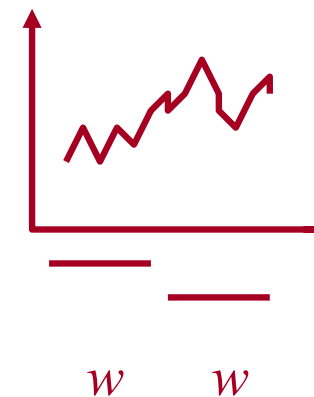
A: R-tree; find qualifying stocks and intervals

Sub-pattern matching



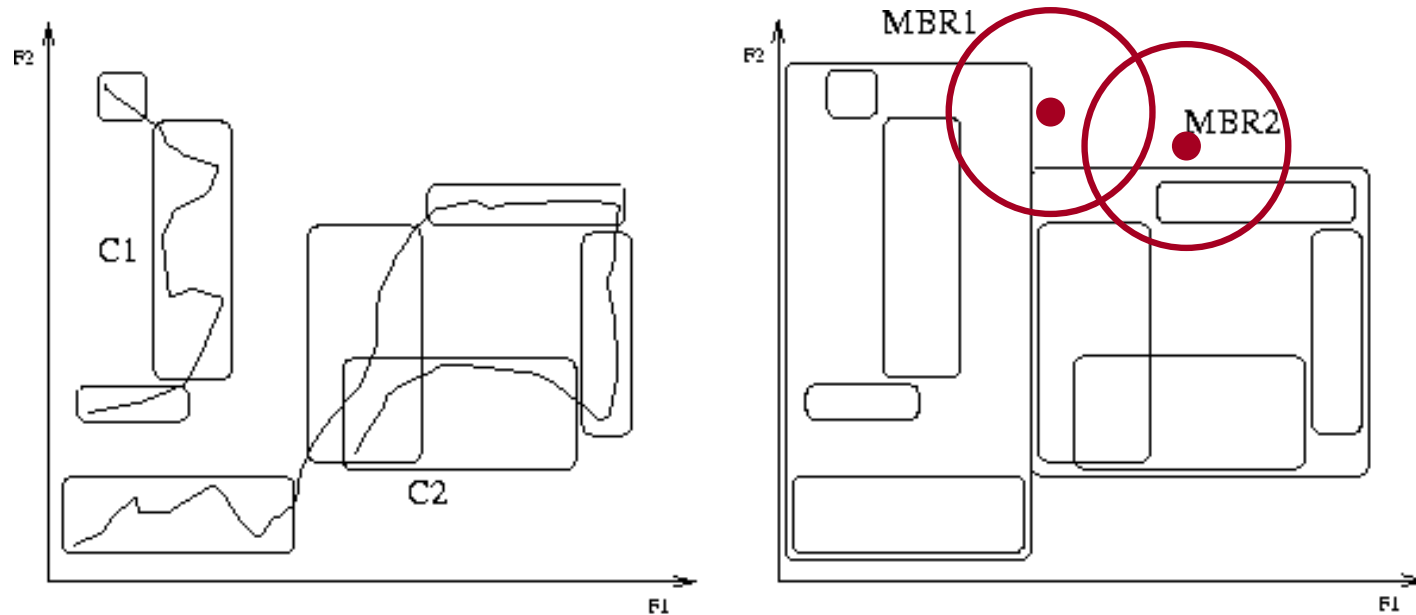
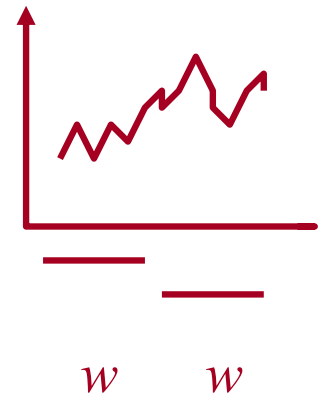
Q: how to do range queries of duration $>w$ (say, $2*w$)?

Sub-pattern matching



Q: how to do range queries of duration $>w$ (say, $2*w$)?

Sub-pattern matching



- Q: how to do range queries of duration $>w$ (say, $2*w$)?
- A: Two range queries of radius epsilon and intersect
(or two queries of smaller radius and union – see paper)

Sub-pattern matching

(improvement [Moon+2001])

- use non-overlapping windows, for data

Conclusions

- GEMINI works for any setting (time sequences, images, etc)
- uses a ‘quick and dirty’ filter
- faster than seq. scan
- (but: how to extract features automatically?)



Multimedia - Detailed outline

- multimedia
 - Motivation / problem definition
 - Main idea / time sequences
 - images (color; shape)
 - sub-pattern matching
 - ➔ – automatic feature extraction / FastMap

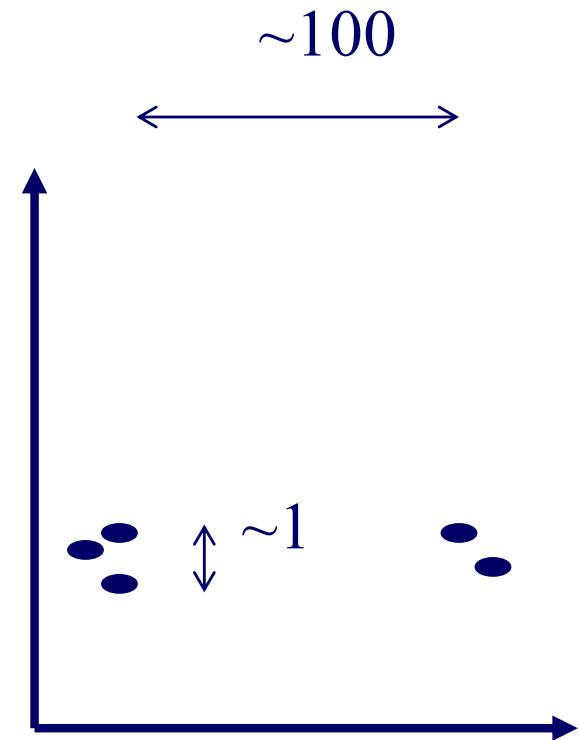
FastMap

Automatic feature extraction:

- Given a dissimilarity function of objects
- Quickly map the objects to a (k-d) 'feature' space.
- (goals: indexing and/or visualization)

FastMap

	O1	O2	O3	O4	O5
O1	0	1	1	100	100
O2	1	0	1	100	100
O3	1	1	0	100	100
O4	100	100	100	0	1
O5	100	100	100	1	0

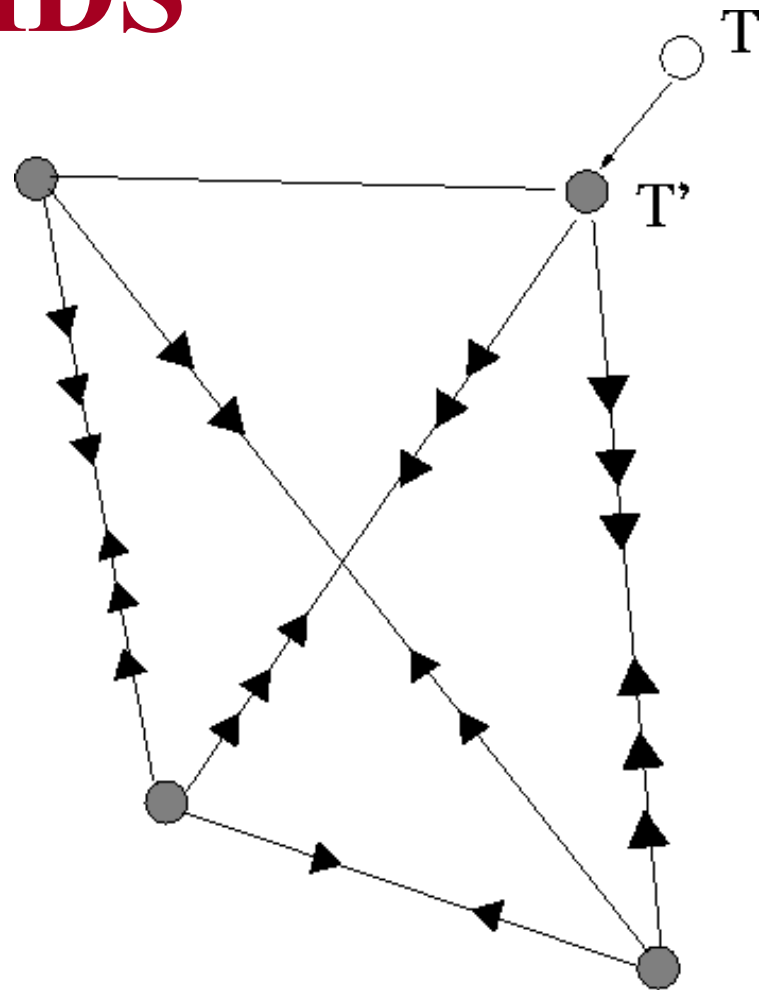


FastMap

- Multi-dimensional scaling (MDS) can do that, but in $O(N^2)$ time

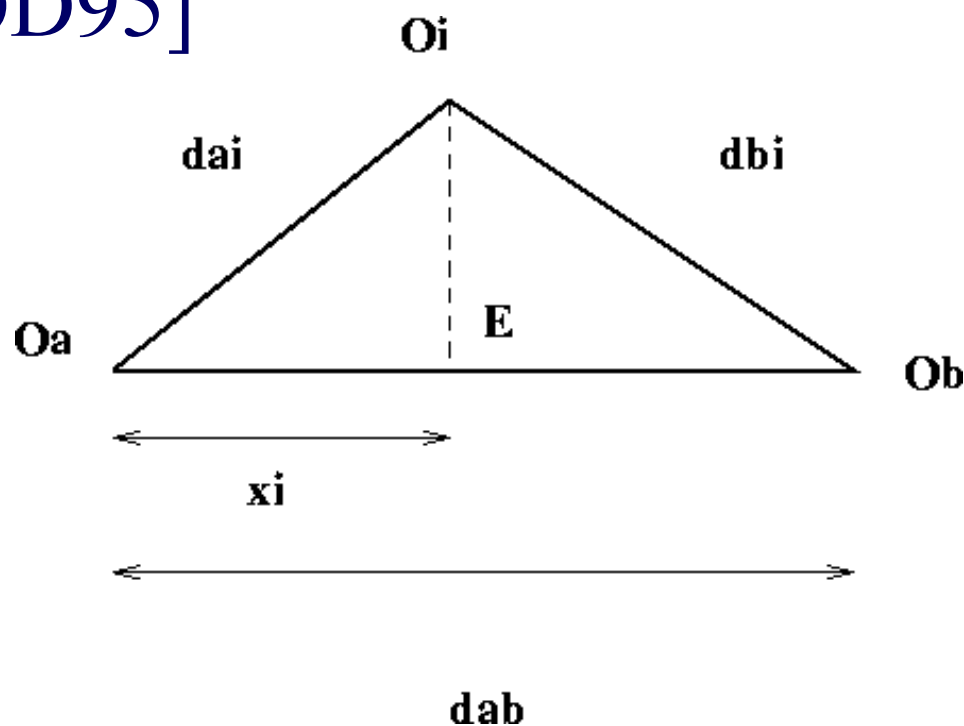
MDS

Multi Dimensional Scaling

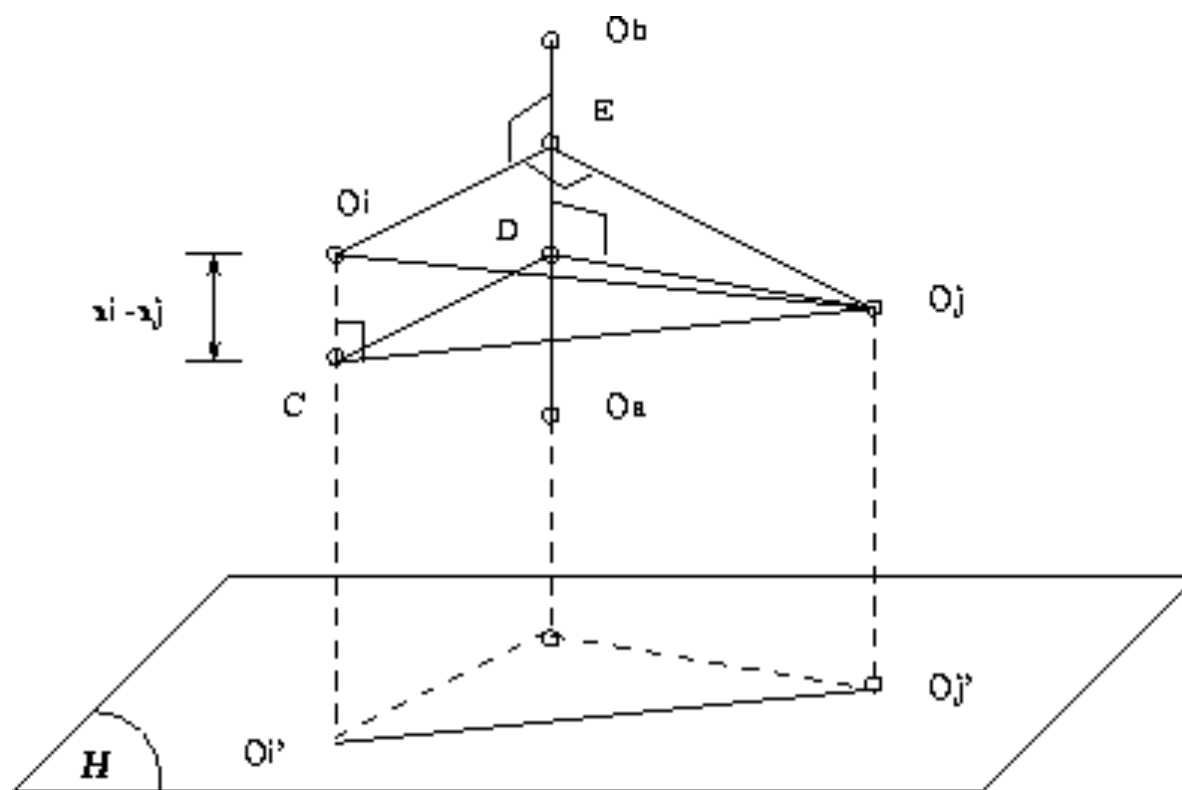


Main idea: projections

We want a **linear** algorithm: FastMap
[SIGMOD95]

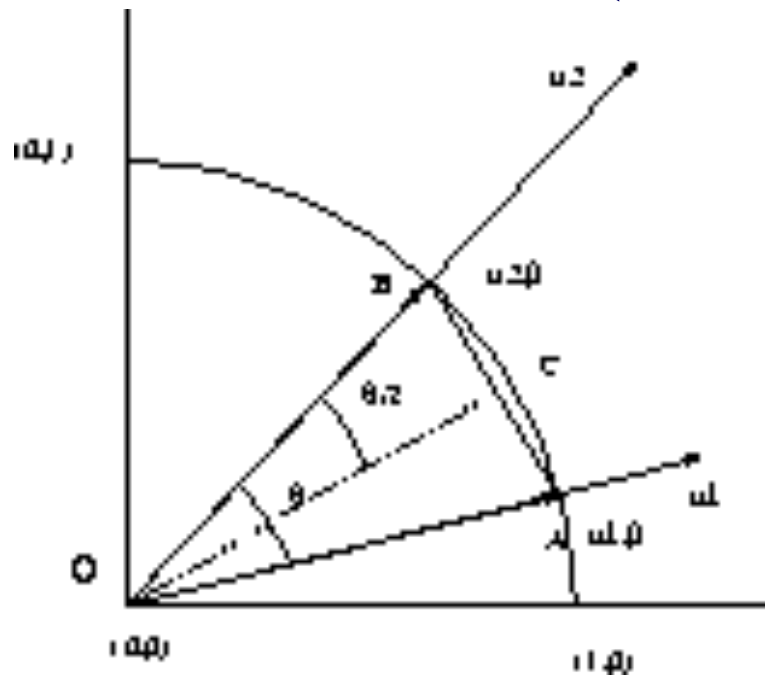


FastMap - next iteration



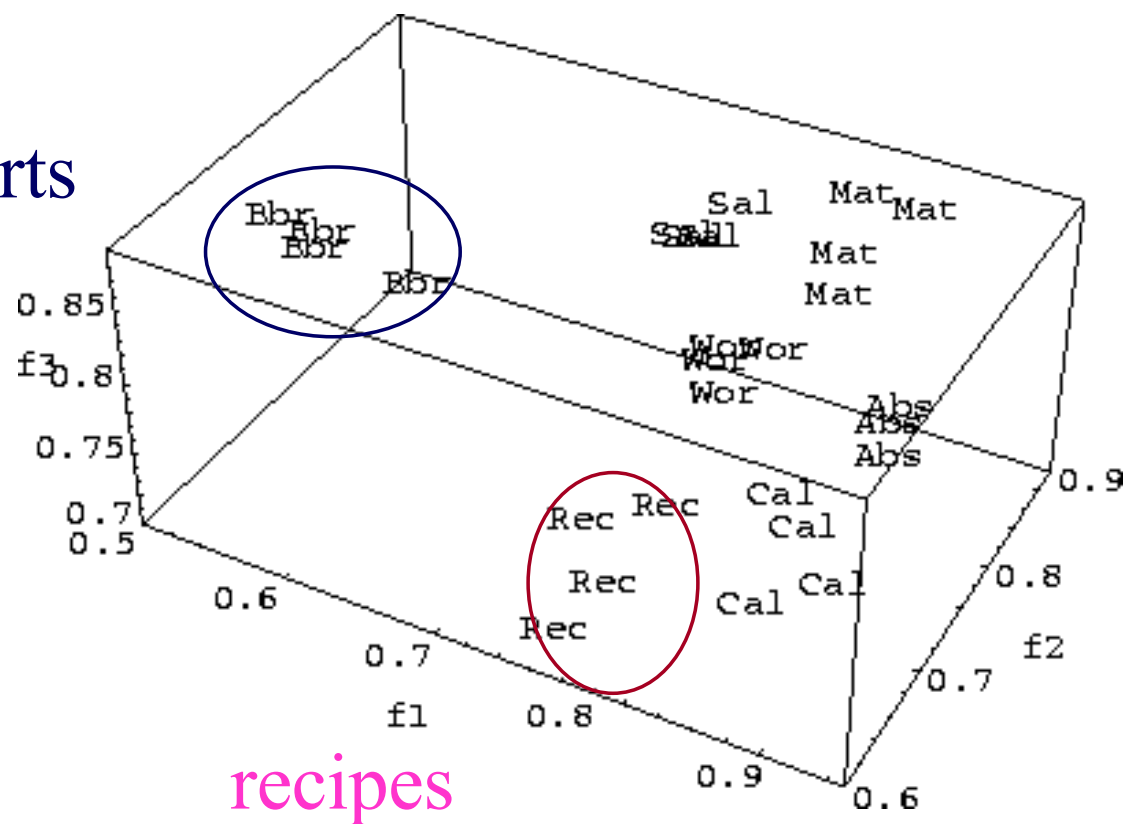
Results

Documents / cosine similarity $\rightarrow$
Euclidean distance (how?)



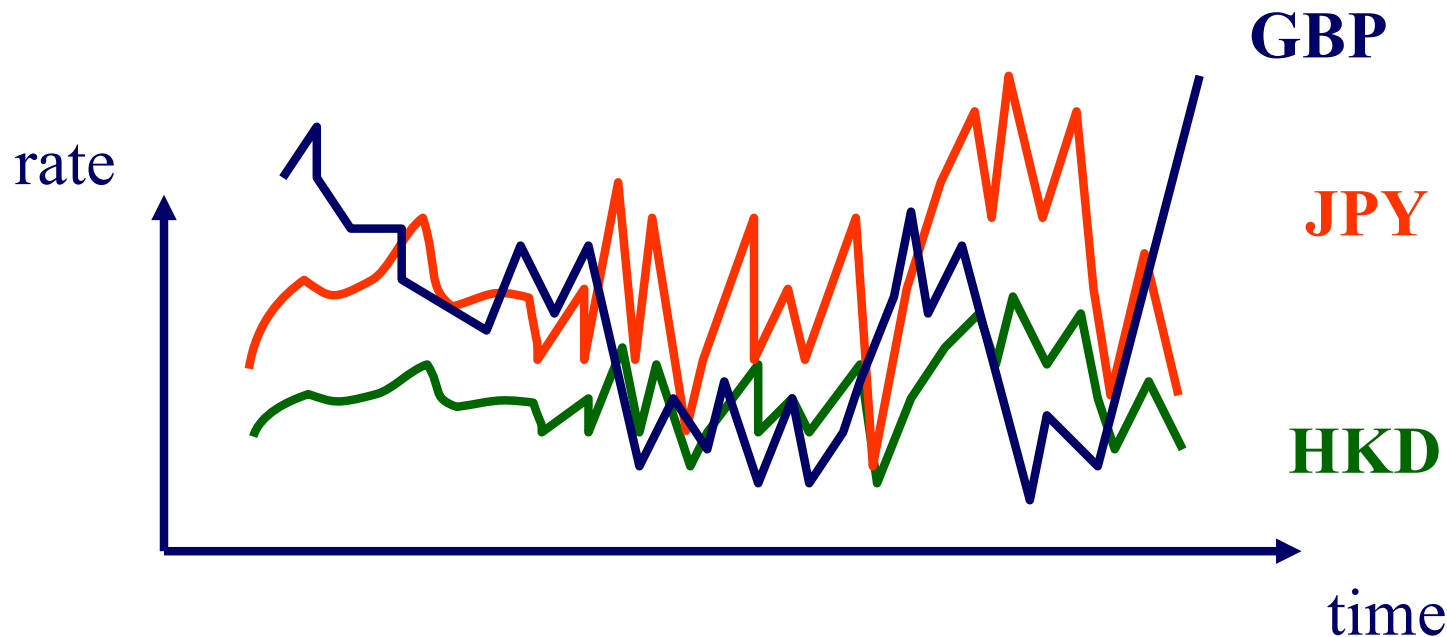
Results

bb reports



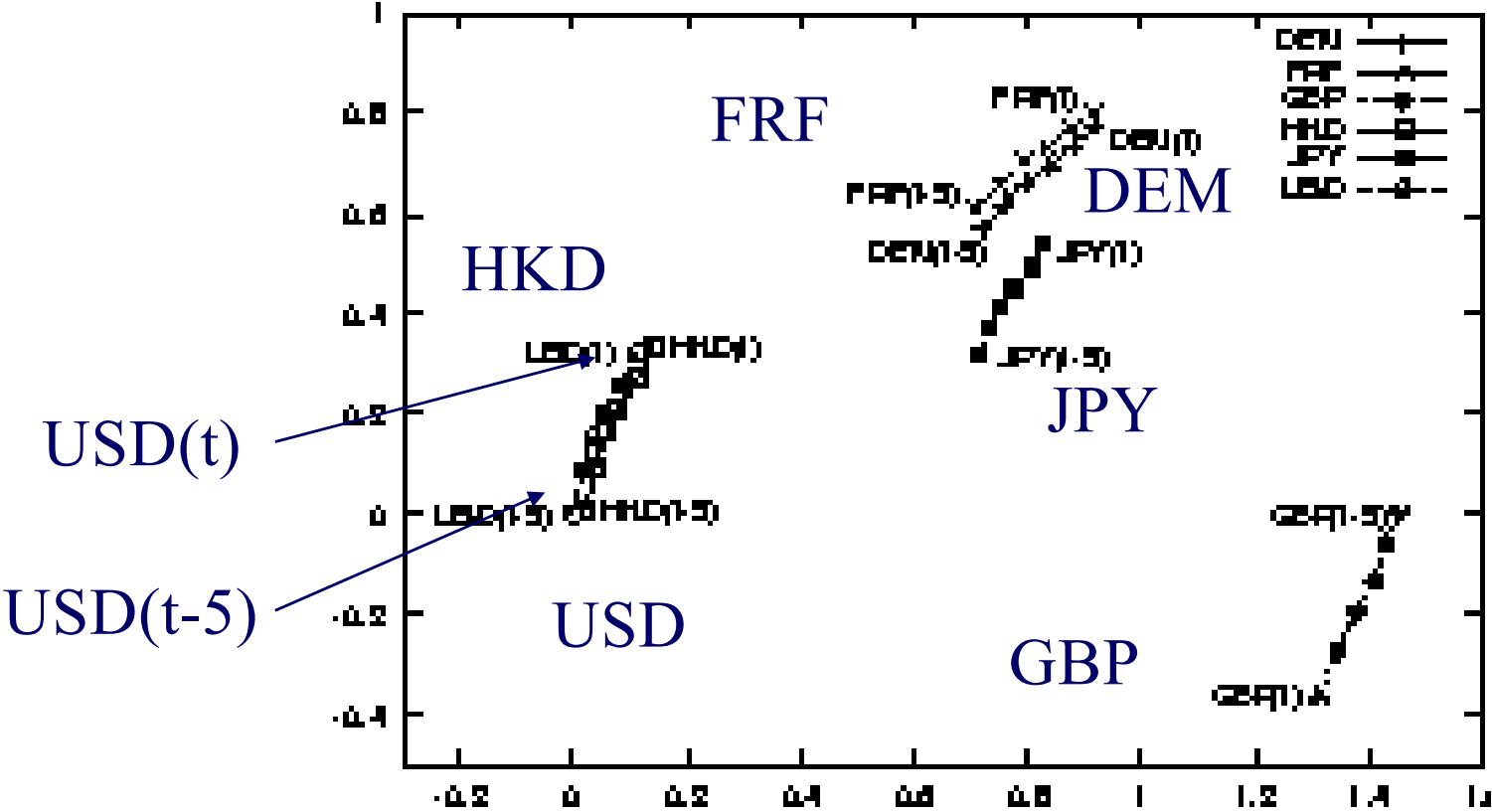
Applications: time sequences

- given n co-evolving time sequences
- visualize them + find rules [ICDE00]



Applications - financial

- currency exchange rates [ICDE00]



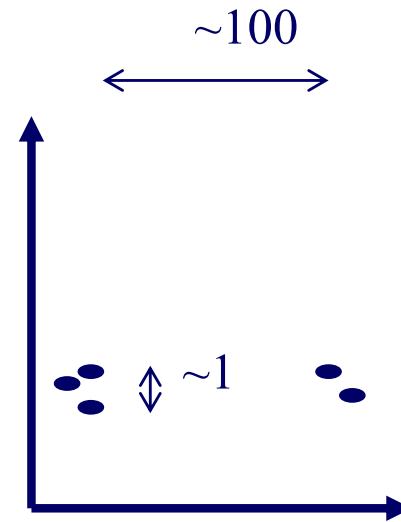
How about SIMILARITY matrix?

D =

	O1	O2	O3	O4	O5
O1	0	1	1	100	100
O2	1	0	1	100	100
O3	1	1	0	100	100
O4	100	100	100	0	1
O5	100	100	100	1	0

S =

	O1	O2	O3	O4	O5
O1	1	0.9	0.9	~0	~0
O2	0.9	1	0.9	~0	~0
O3	0.9	0.9	1	~0	~0
O4	~0	~0	~0	1	0.9
O5	~0	~0	~0	0.9	1



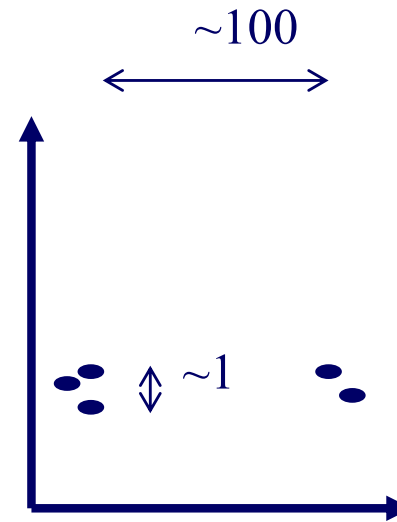
How about SIMILARITY matrix?

D =

	O1	O2	O3	O4	O5
O1	0	1	1	100	100
O2	1	0	1	100	100
O3	1	1	0	100	100
O4	100	100	100	0	1
O5	100	100	100	1	0

S =

	O1	O2	O3	O4	O5
O1	1	0.9	0.9	~0	~0
O2	0.9	1	0.9	~0	~0
O3	0.9	0.9	1	~0	~0
O4	~0	~0	~0	1	0.9
O5	~0	~0	~0	0.9	1



Q: Find 5 (unit) vectors, so that their dot products match S

How about SIMILARITY matrix?

A: rank- r SVD of S ($= U\Lambda U^T$)

keep $U \Lambda^{1/2}$

Q1: why is this correct

Q2: why it may be better than FastMap

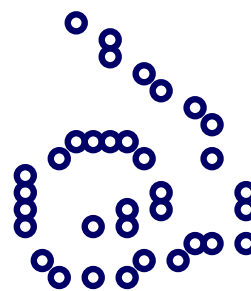
$S =$

	O1	O2	O3	O4	O5
O1	1	0.9	0.9	~0	~0
O2	0.9	1	0.9	~0	~0
O3	0.9	0.9	1	~0	~0
O4	~0	~0	~0	1	0.9
O5	~0	~0	~0	0.9	1

Q: Find 5 (unit) vectors,
so that their dot products
match S

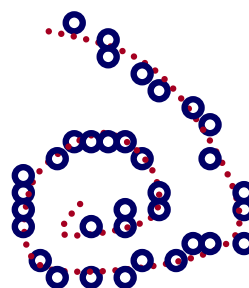
Variations

- Isomap [Tenenbaum, de Silva, Langford, 2000]
- LLE (Local Linear Embedding) [Roweis, Saul, 2000]
- MVE (Minimum Volume Embedding) [Shaw & Jebara, 2007]



Variations

- Isomap [Tenenbaum, de Silva, Langford, 2000]
- LLE (Local Linear Embedding) [Roweis, Saul, 2000]
- MVE (Minimum Volume Embedding) [Shaw & Jebara, 2007]



Variations

- tSNE (see [sklearn](#)) and [JMLR](#)
 - Nearby entities -> nearby points
 - Far apart entities -> "don't care"
- Umap (see [scikit](#)) and [Arxiv](#)
 - Nearby entities -> nearby points
 - Far apart entities -> sort-of far-apart points

Variations

- tSNE: recommended – but: tune ‘perplexity’ (~ 50)

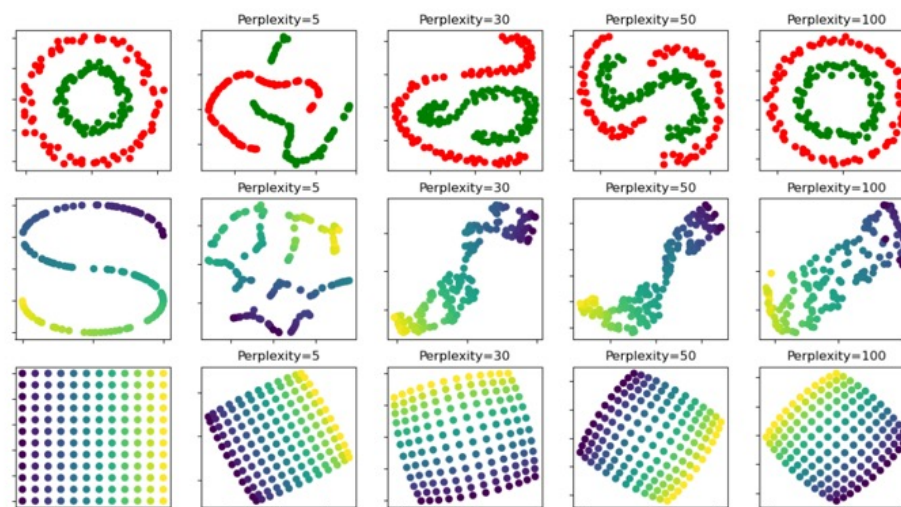


Image: from [scikit-learn](https://scikit-learn.org/stable/auto_examples/t_sne/plot_perplexity.html)

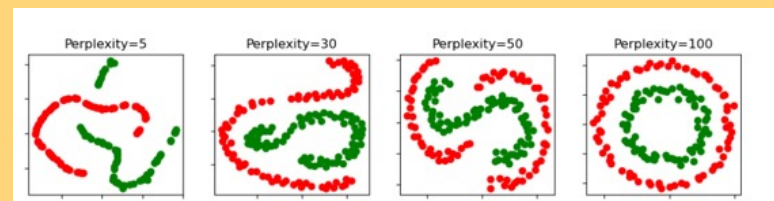
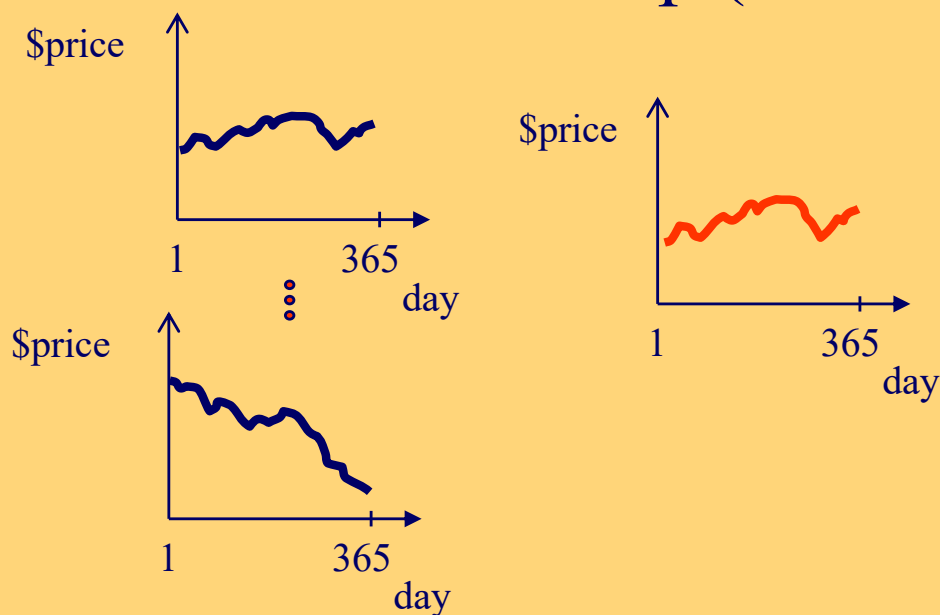
Conclusions

- GEMINI works for multiple settings
- MDS (FastMap/tSNE/UMap) can extract ‘features’ automatically (-> indexing, visual d.m.)



Solution

- Q: Find stocks similar to $\langle \text{MSFT} \rangle$
- A: GEMINI: Extract features + SAM
 - A': and FastMap (tSNE/Umap), for feature extraction



References

- Faloutsos, C., R. Barber, et al. (July 1994). “*Efficient and Effective Querying by Image Content.*” J. of Intelligent Information Systems 3(3/4): 231-262.
- Faloutsos, C. and K.-I. D. Lin (May 1995). *FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets.* Proc. of ACM-SIGMOD, San Jose, CA.
- Faloutsos, C., M. Ranganathan, et al. (May 25-27, 1994). *Fast Subsequence Matching in Time-Series Databases.* Proc. ACM SIGMOD, Minneapolis, MN.

References

- Flickner, M., H. Sawhney, et al. (Sept. 1995). “*Query by Image and Video Content: The QBIC System.*” IEEE Computer 28(9): 23-32.
- Goldin, D. Q. and P. C. Kanellakis (Sept. 19-22, 1995). *On Similarity Queries for Time-Series Data: Constraint Specification and Implementation*. Int. Conf. on Principles and Practice of Constraint Programming (CP95), Cassis, France.
- Flip Korn, Nikolaos Sidiropoulos, Christos Faloutsos, Eliot Siegel, Zenon Protopapas: *Fast Nearest Neighbor Search in Medical Image Databases*. VLDB 1996: 215-226

References

- Leland, W. E., M. S. Taqqu, et al. (Feb. 1994). “*On the Self-Similar Nature of Ethernet Traffic.*” IEEE Transactions on Networking 2(1): 1-15.
- Laurens van der Maaten, Geoffrey Hinton, *Visualizing Data using t-SNE*, JMLR 9(86):2579–2605, 2008.
- Leland McInnes, John Healy, James Melville, *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*, arxiv, 2018

References

- P. Maragos, *Pattern spectrum and multiscale shape representation*, IEEE Trans. PAMI, 11,7, July 1989
- Moon, Y.-S., K.-Y. Whang, et al. (2001). *Duality-Based Subsequence Matching in Time-Series Databases*. ICDE, Heidelberg, Germany.
- Rafiei, D. and A. O. Mendelzon (1997). *Similarity-Based Queries for Time Series Data*. SIGMOD Conference, Tucson, AZ.

References

- Lawrence Saul & Sam Roweis. *An Introduction to Locally Linear Embedding* (draft)
- Sam Roweis & Lawrence Saul. *Nonlinear dimensionality reduction by locally linear embedding*. Science, v.290 [no.5500](#) , Dec.22, 2000. pp.2323--2326.
- Schroeder, M. (1991). *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise*. New York, W.H. Freeman and Company.
- B. Shaw and T. Jebara. "*Minimum Volume Embedding*" . Artificial Intelligence and Statistics, AISTATS, March 2007.

References

- Josh Tenenbaum, Vin de Silva and John Langford. *A Global Geometric Framework for Nonlinear dimensionality Reduction*. Science 290, pp. 2319-2323, 2000.
- Yi, B.-K. and C. Faloutsos (2000). *Fast Time Sequence Indexing for Arbitrary L_p Norms*. VLDB, Cairo, Egypt.

References

- Josh Tenenbaum, Vin de Silva and John Langford. *A Global Geometric Framework for Nonlinear dimensionality Reduction*. Science 290, pp. 2319-2323, 2000.
- Yi, B.-K. and C. Faloutsos (2000). *Fast Time Sequence Indexing for Arbitrary L_p Norms*. VLDB, Cairo, Egypt.

References

- ★ • Laurens van der Maaten, Geoffrey Hinton, [*Visualizing Data using t-SNE*](#), JMLR 9(86):2579–2605, 2008 ([scikit-learn](#) :

```
from sklearn.manifold import TSNE
```

- ★ • Leland McInnes, John Healy, James Melville, [*UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*](#), arxiv, 2018 ([python library](#))