



Mining Time Series Data

Christos Faloutsos
CMU



Thanks

 Deepay Chakrabarti (CMU)

 Prof. Dimitris Gunopulos (UCR)

 Spiros Papadimitriou (CMU)

 Mengzhi Wang (CMU)

 Prof. Byoung-Kee Yi (Pohang U.)

SBBD 05 Copyright: C. Faloutsos, 2005 2



Outline

- ➔ • Motivation
- Similarity Search and Indexing
- DSP (Digital Signal Processing)
- Linear Forecasting
- Bursty traffic - fractals and multifractals
- Non-linear forecasting
- Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 3



Problem definition

- Given: one or more sequences
 $x_1, x_2, \dots, x_t, \dots$
 $(y_1, y_2, \dots, y_p, \dots)$
 $\dots$
- Find
 - similar sequences; forecasts
 - patterns; clusters; outliers

SBBD 05 Copyright: C. Faloutsos, 2005 4



Motivation - Applications

- Financial, sales, economic series
- Medical
 - ECGs +; blood pressure etc monitoring
 - reactions to new drugs
 - elderly care

SBBD 05 Copyright: C. Faloutsos, 2005 5



Motivation - Applications (cont'd)

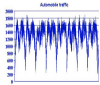
- 'Smart house'
 - sensors monitor temperature, humidity, air quality
- video surveillance

SBBD 05 Copyright: C. Faloutsos, 2005 6

CMU SCS

Motivation - Applications (cont'd)

- civil/automobile infrastructure
 - bridge vibrations [Oppenheim+02]
 - road conditions / traffic monitoring

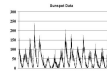


SBBD 05 Copyright: C. Faloutsos, 2005 7

CMU SCS

Motivation - Applications (cont'd)

- Weather, environment/anti-pollution
 - volcano monitoring
 - air/water pollutant monitoring



SBBD 05 Copyright: C. Faloutsos, 2005 8

CMU SCS

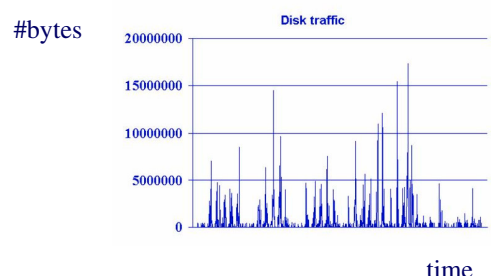
Motivation - Applications (cont'd)

- Computer systems
 - 'Active Disks' (buffering, prefetching)
 - web servers (ditto)
 - network traffic monitoring
 - ...

SBBD 05 Copyright: C. Faloutsos, 2005 9

CMU SCS

Stream Data: Disk accesses



SBBD 05 Copyright: C. Faloutsos, 2005 10

CMU SCS

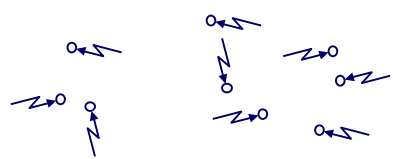
Settings & Applications

- One or more sensors, collecting time-series data

SBBD 05 Copyright: C. Faloutsos, 2005 11

CMU SCS

Settings & Applications

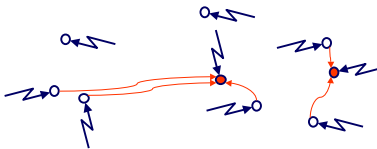


Each sensor collects data $(x_1, x_2, \dots, x_p, \dots)$

SBBD 05 Copyright: C. Faloutsos, 2005 12

CMU SCS

Settings & Applications



Some sensors 'report' to others or to the central site

SBBD 05 Copyright: C. Faloutsos, 2005 13

CMU SCS

Settings & Applications

Goal #1:
Finding patterns
in a single time sequence



SBBD 05 Copyright: C. Faloutsos, 2005 14

CMU SCS

Settings & Applications

Goal #2:
Finding patterns
in many time sequences

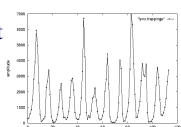


SBBD 05 Copyright: C. Faloutsos, 2005 15

CMU SCS

Problem #1:

Goal: given a signal (eg., #packets over time)
Find: patterns, periodicities, and/or compress



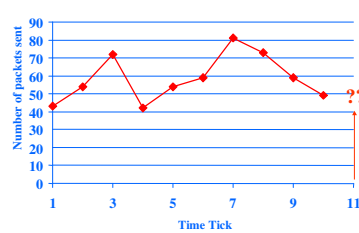
lynx caught per year
(packets per day;
temperature per day)

SBBD 05 Copyright: C. Faloutsos, 2005 16

CMU SCS

Problem#2: Forecast

Given $x_t, x_{t-1}, \dots$, forecast x_{t+1}

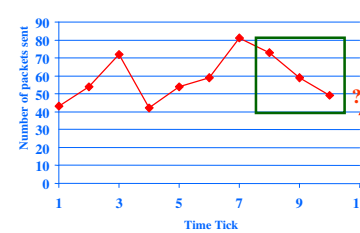


SBBD 05 Copyright: C. Faloutsos, 2005 17

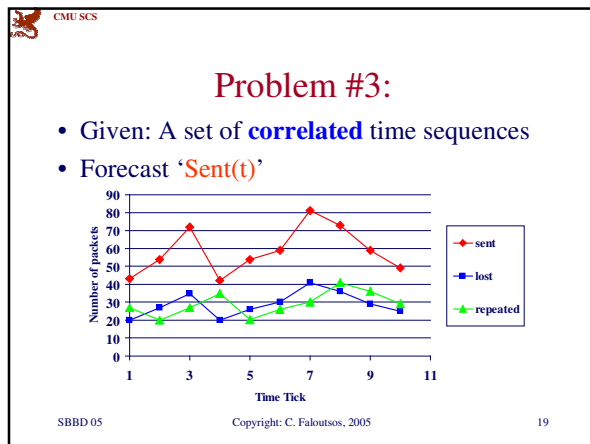
CMU SCS

Problem#2': Similarity search

Eg., Find a 3-tick pattern, similar to the last one



SBBD 05 Copyright: C. Faloutsos, 2005 18



CMU SCS

Differences from DSP/Stat

- Semi-infinite streams
 - we need on-line, 'any-time' algorithms
- Can not afford human intervention
 - need automatic methods
- sensors have limited memory / processing / transmitting power
 - need for (lossy) compression

SBBD 05 Copyright: C. Faloutsos, 2005 20

CMU SCS

Important observations

Patterns, rules, forecasting and similarity indexing are closely related:

- To do forecasting, we need
 - to find patterns/rules
 - to find similar settings in the past
- to find outliers, we need to have forecasts
 - (outlier = too far away from our forecast)

SBBD 05 Copyright: C. Faloutsos, 2005 21

CMU SCS

Important topics NOT in this tutorial:

- Continuous queries
 - [Babu+Widom] [Gehrke+] [Madden+]
- Categorical data streams
 - [Hatonen+96]
- Outlier detection (discontinuities)
 - [Breunig+00]
- Related (see D. Shasha's tutorial)

SBBD 05 Copyright: C. Faloutsos, 2005 22

CMU SCS

Outline

- Motivation
- ➡ Similarity Search and Indexing
- DSP
- Linear Forecasting
- Bursty traffic - fractals and multifractals
- Non-linear forecasting
- Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 23

CMU SCS

Outline

- Motivation
- ➡ Similarity Search and Indexing
 - distance functions: Euclidean; Time-warping
 - indexing
 - feature extraction
- DSP
- ...

SBBD 05 Copyright: C. Faloutsos, 2005 24

CMU SCS

Importance of distance functions

Subtle, but **absolutely necessary**:

- A 'must' for similarity indexing (-> forecasting)
- A 'must' for clustering

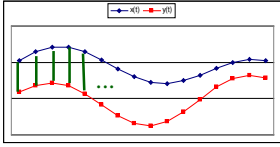
Two major families

- Euclidean and Lp norms
- Time warping and variations

SBBD 05 Copyright: C. Faloutsos, 2005 25

CMU SCS

Euclidean and Lp



$$D(\vec{x}, \vec{y}) = \sum_{i=1}^n (x_i - y_i)^2$$

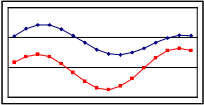
$$L_p(\vec{x}, \vec{y}) = \sum_{i=1}^n |x_i - y_i|^p$$

- L_1 : city-block = Manhattan
- L_2 = Euclidean
- L_∞

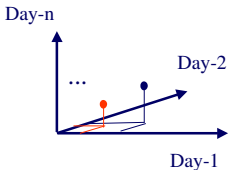
SBBD 05 Copyright: C. Faloutsos, 2005 26

CMU SCS

Observation #1



- Time sequence -> n-d vector



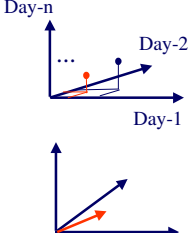
SBBD 05 Copyright: C. Faloutsos, 2005 27

CMU SCS

Observation #2

Euclidean distance is closely related to

- cosine similarity
- dot product
- 'cross-correlation' function



SBBD 05 Copyright: C. Faloutsos, 2005 28

CMU SCS

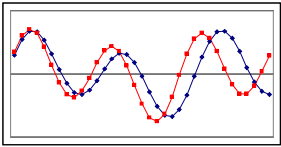
Time Warping

- allow accelerations - decelerations
 - (with or w/o penalty)
- THEN compute the (Euclidean) distance (+ penalty)
- related to the string-editing distance

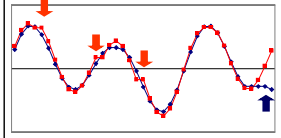
SBBD 05 Copyright: C. Faloutsos, 2005 29

CMU SCS

Time Warping



'stutters':



SBBD 05 Copyright: C. Faloutsos, 2005 30

CMU SCS

Skip

Time warping

Q: how to compute it?
 A: dynamic programming
 $D(i, j)$ = cost to match
 prefix of length i of first sequence x with prefix
 of length j of second sequence y

SBBD 05 Copyright: C. Faloutsos, 2005 31

CMU SCS

Skip

Time warping

Time warping Thus, with no penalty for stutter, for sequences
 $x_1, x_2, \dots, x_i, \quad y_1, y_2, \dots, y_j$

$$D(i, j) = \|x[i] - y[j]\| + \min \begin{cases} D(i-1, j-1) & \text{no stutter} \\ D(i, j-1) & \text{x-stutter} \\ D(i-1, j) & \text{y-stutter} \end{cases}$$

SBBD 05 Copyright: C. Faloutsos, 2005 32

CMU SCS

Skip

Time warping

- Complexity: $O(M*N)$ - quadratic on the length of the strings
- Many variations (penalty for stutters; limit on the number/percentage of stutters; ...)
- popular in voice processing [Rabiner+Juang]

SBBD 05 Copyright: C. Faloutsos, 2005 33

CMU SCS

Other Distance functions

- piece-wise linear/flat approx.; compare pieces [Keogh+01] [Faloutsos+97]
- 'cepstrum' (for voice [Rabiner+Juang])
 – do DFT; take log of amplitude; do DFT again!
- Allow for small gaps [Agrawal+95]

See tutorial by [Gunopulos Das, SIGMOD01]

SBBD 05 Copyright: C. Faloutsos, 2005 34

CMU SCS

Conclusions

Prevailing distances:

- Euclidean and
- time-warping

SBBD 05 Copyright: C. Faloutsos, 2005 35

CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
 - distance functions
 - indexing
 - feature extraction
- DSP
- ...

SBBD 05 Copyright: C. Faloutsos, 2005 36

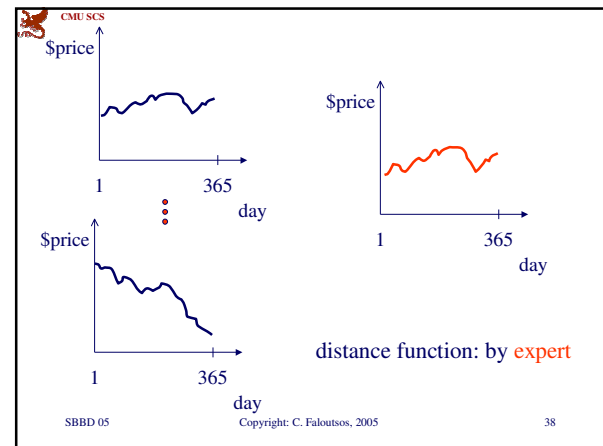
CMU SCS

Indexing

Problem:

- given a set of time sequences,
- find the ones similar to a desirable query sequence

SBBD 05 Copyright: C. Faloutsos, 2005 37



CMU SCS

Idea: 'GEMINI'

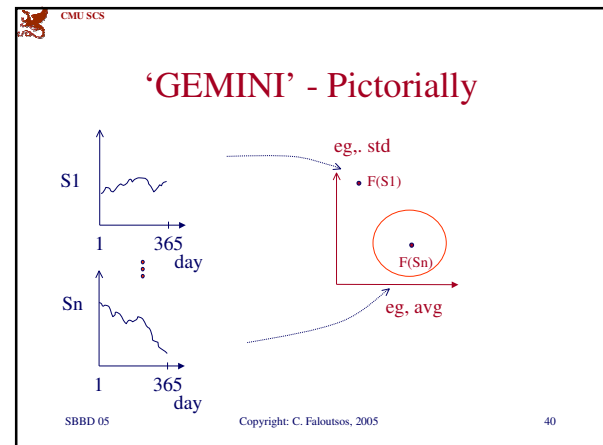
Eg., 'find stocks similar to MSFT'

Seq. scanning: too slow

How to accelerate the search?

[Faloutsos96]

SBBD 05 Copyright: C. Faloutsos, 2005 39



CMU SCS

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

SBBD 05 Copyright: C. Faloutsos, 2005 41

CMU SCS

Examples of GEMINI

- Time sequences: DFT (up to 100 times faster) [SIGMOD94];
- [Kanellakis+], [Mendelzon+]

SBBD 05 Copyright: C. Faloutsos, 2005 42

Examples of GEMINI

Even on other-than-sequence data:

- Images (QBIC) [JIIS94]
- tumor-like shapes [VLDB96]
- video [Informedia + S-R-trees]
- automobile part shapes [Kriegel+97]

SBBD 05

Copyright: C. Faloutsos, 2005

43

Indexing - SAMs

Q: How do Spatial Access Methods (SAMs) work?

A: they group nearby points (or regions) together, on nearby disk pages, and answer spatial queries quickly ('range queries', 'nearest neighbor' queries etc)

For example:

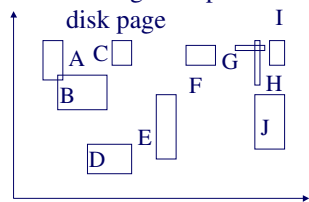
SBBD 05

Copyright: C. Faloutsos, 2005

44

R-trees

- [Guttman84] eg., w/ fanout 4: group nearby rectangles to parent MBRs; each group -> disk page



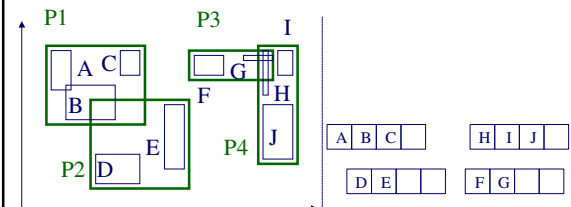
SBBD 05

Copyright: C. Faloutsos, 2005

45

R-trees

- eg., w/ fanout 4:



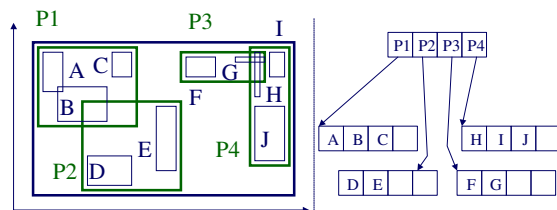
SBBD 05

Copyright: C. Faloutsos, 2005

46

R-trees

- eg., w/ fanout 4:

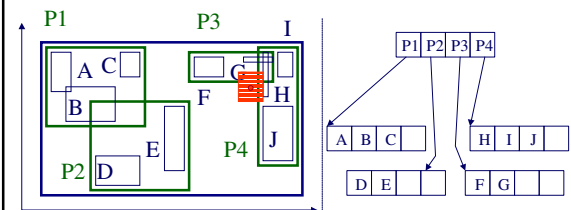


SBBD 05

Copyright: C. Faloutsos, 2005

47

R-trees - range search?



SBBD 05

Copyright: C. Faloutsos, 2005

48

CMU SCS

R-trees - range search?

Skip

SBBD 05 Copyright: C. Faloutsos, 2005 49

CMU SCS

Conclusions

- Fast indexing: through GEMINI
 - feature extraction and
 - (off the shelf) Spatial Access Methods [Gaede+98]

SBBD 05 Copyright: C. Faloutsos, 2005 50

CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
 - distance functions
 - indexing
 - feature extraction
- DSP
- ...

SBBD 05 Copyright: C. Faloutsos, 2005 51

CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
 - distance functions
 - indexing
 - feature extraction
 - DFT, DWT, DCT (data independent)
 - SVD, etc (data dependent)
 - MDS, FastMap

SBBD 05 Copyright: C. Faloutsos, 2005 52

CMU SCS

DFT and cousins

- very good for compressing real signals
- more details on DFT/DCT/DWT: later

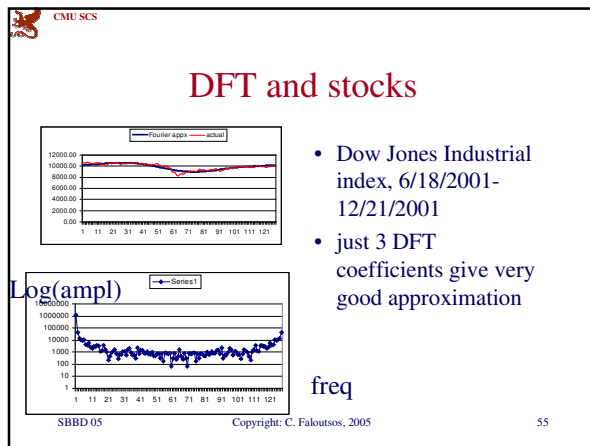
SBBD 05 Copyright: C. Faloutsos, 2005 53

CMU SCS

DFT and stocks

- Dow Jones Industrial index, 6/18/2001-12/21/2001

SBBD 05 Copyright: C. Faloutsos, 2005 54



CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
 - distance functions
 - indexing
 - feature extraction
 - DFT, DWT, DCT (data independent)
 - SVD etc (data dependent)
 - MDS, FastMap

SBBD 05 Copyright: C. Faloutsos, 2005 56

CMU SCS

SVD

- THE optimal method for dimensionality reduction
 - (under the Euclidean metric)

SBBD 05 Copyright: C. Faloutsos, 2005 57

CMU SCS

Singular Value Decomposition (SVD)

- SVD (~LSI ~ KL ~ PCA ~ spectral analysis...)

LSI: S. Dumais; M. Berry
 KL: eg, Duda+Hart
 PCA: eg., Jolliffe
 Details: [Press+], [Faloutsos96]

SBBD 05 Copyright: C. Faloutsos, 2005 58

CMU SCS

SVD

- Extremely useful tool
 - (also behind PageRank/google and Kleinberg's algorithm for hubs and authorities)
- But may be slow: $O(N * M * M)$ if $N > M$
- any approximate, faster method?

SBBD 05 Copyright: C. Faloutsos, 2005 59

CMU SCS

SVD shortcuts

- random projections (Johnson-Lindenstrauss thm [Papadimitriou+ pods98])

SBBD 05 Copyright: C. Faloutsos, 2005 60

CMU SCS

Random projections

- pick 'enough' random directions (will be ~orthogonal, in high-d!!)
- distances are preserved probabilistically, within epsilon
- (also, use as a pre-processing step for SVD [Papadimitriou+ PODS98])

SBBD 05 Copyright: C. Faloutsos, 2005 61

CMU SCS

Feature extraction - w/ fractals

Skip

- Main idea: drop those attributes that don't affect the intrinsic ('fractal') dimensionality [Traina+, SBBD 2000]
- ie., drop attributes that depend on others (linearly or non-linearly!)

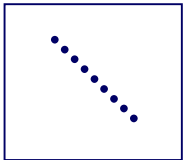
SBBD 05 Copyright: C. Faloutsos, 2005 62

CMU SCS

Fractals

Fractal dimension
= intrinsic dimension
~ degrees of freedom

Real data: often self-similar, with NON-INTEGER intrinsic dimension (!)



Skip

SBBD 05 Copyright: C. Faloutsos, 2005 63

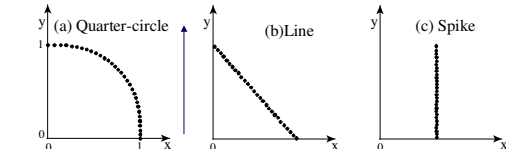
CMU SCS

Feature extraction - w/ fractals

Skip

global FD=1

PFD~1



(a) Quarter-circle (b) Line (c) Spike

PFD~1

SBBD 05 Copyright: C. Faloutsos, 2005 64

CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
 - distance functions
 - indexing
 - feature extraction
 - DFT, DWT, DCT (data independent)
 - SVD (data dependent)
 - MDS, FastMap

➔

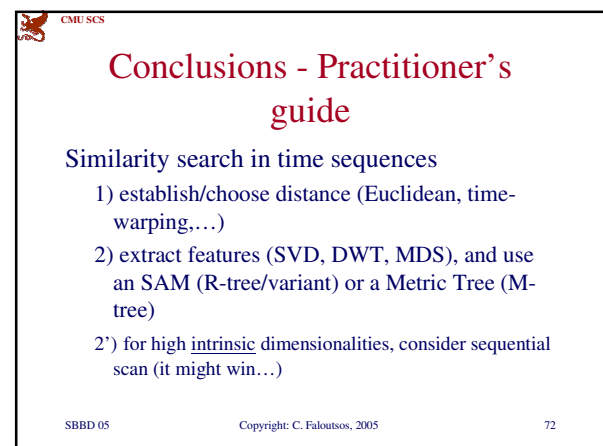
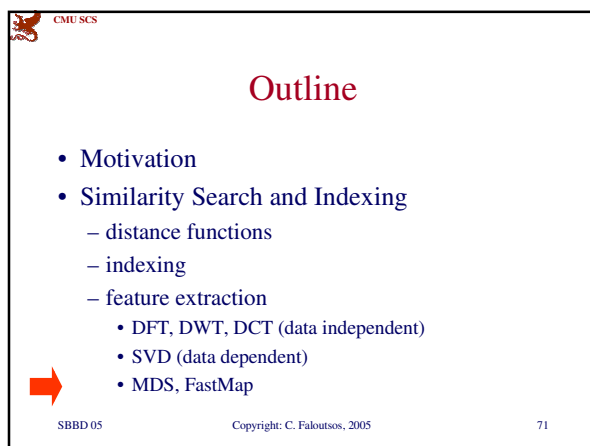
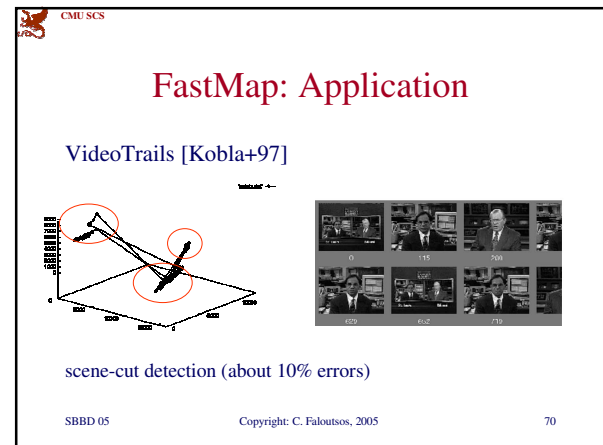
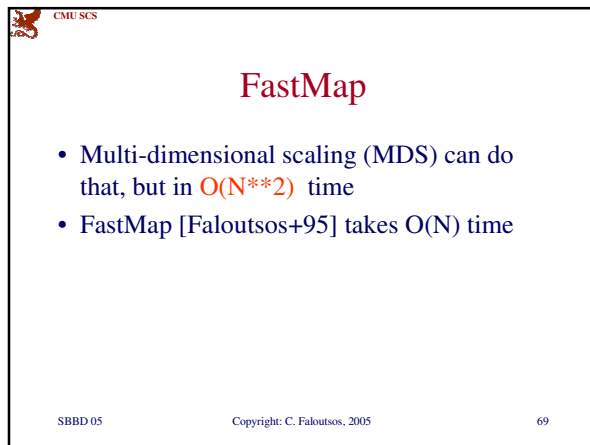
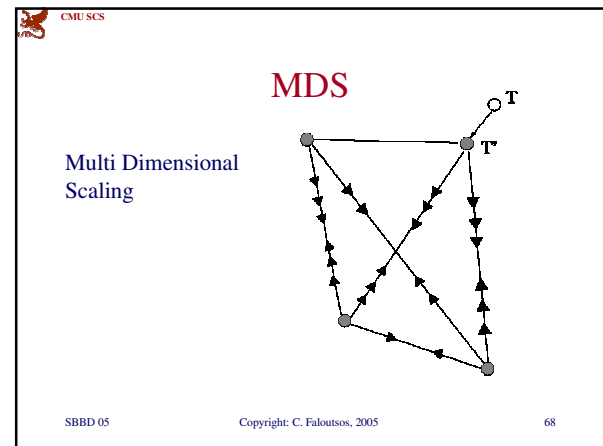
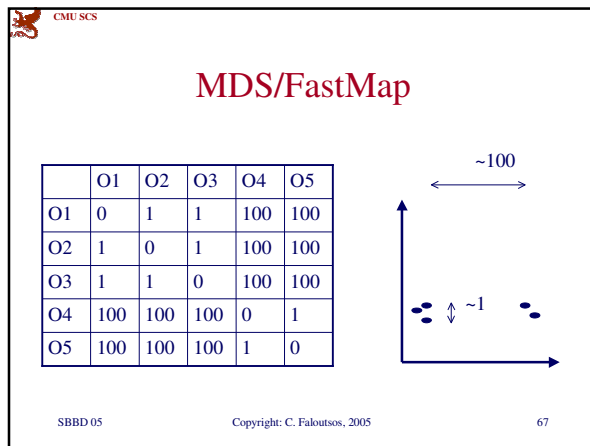
SBBD 05 Copyright: C. Faloutsos, 2005 65

CMU SCS

MDS / FastMap

- but, what if we have NO points to start with? (eg. Time-warping distance)
- A: Multi-dimensional Scaling (MDS) ; FastMap

SBBD 05 Copyright: C. Faloutsos, 2005 66



 CMU SCS

Books

- William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery: *Numerical Recipes in C*, Cambridge University Press, 1992, 2nd Edition. (Great description, intuition and code for SVD)
- C. Faloutsos: *Searching Multimedia Databases by Content*, Kluwer Academic Press, 1996 (introduction to SVD, and GEMINI)

SBBD 05 Copyright: C. Faloutsos, 2005 73

 CMU SCS

References

- Agrawal, R., K.-I. Lin, et al. (Sept. 1995). Fast Similarity Search in the Presence of Noise, Scaling and Translation in Time-Series Databases. Proc. of VLDB, Zurich, Switzerland.
- Babu, S. and J. Widom (2001). "Continuous Queries over Data Streams." SIGMOD Record 30(3): 109-120.
- Breunig, M. M., H.-P. Kriegel, et al. (2000). LOF: Identifying Density-Based Local Outliers. SIGMOD Conference, Dallas, TX.
- Berry, Michael: <http://www.cs.utk.edu/~lsi/>

SBBD 05 Copyright: C. Faloutsos, 2005 74

 CMU SCS

References

- Ciaccia, P., M. Patella, et al. (1997). M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. VLDB.
- Foltz, P. W. and S. T. Dumais (Dec. 1992). "Personalized Information Delivery: An Analysis of Information Filtering Methods." Comm. of ACM (CACM) 35(12): 51-60.
- Guttman, A. (June 1984). R-Trees: A Dynamic Index Structure for Spatial Searching. Proc. ACM SIGMOD, Boston, Mass.

SBBD 05 Copyright: C. Faloutsos, 2005 75

 CMU SCS

References

- Gaede, V. and O. Guenther (1998). "Multidimensional Access Methods." Computing Surveys 30(2): 170-231.
- Gehrke, J. E., F. Korn, et al. (May 2001). On Computing Correlated Aggregates Over Continual Data Streams. ACM Sigmod, Santa Barbara, California.

SBBD 05 Copyright: C. Faloutsos, 2005 76

 CMU SCS

References

- Gunopulos, D. and G. Das (2001). Time Series Similarity Measures and Time Series Indexing. SIGMOD Conference, Santa Barbara, CA.
- Hatonen, K., M. Klemettinen, et al. (1996). Knowledge Discovery from Telecommunication Network Alarm Databases. ICDE, New Orleans, Louisiana.
- Jolliffe, I. T. (1986). Principal Component Analysis, Springer Verlag.

SBBD 05 Copyright: C. Faloutsos, 2005 77

 CMU SCS

References

- Keogh, E. J., K. Chakrabarti, et al. (2001). Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases. SIGMOD Conference, Santa Barbara, CA.
- Kobla, V., D. S. Doermann, et al. (Nov. 1997). VideoTrails: Representing and Visualizing Structure in Video Sequences. ACM Multimedia 97, Seattle, WA.

SBBD 05 Copyright: C. Faloutsos, 2005 78



CMU SCS

References

- Oppenheim, I. J., A. Jain, et al. (March 2002). A MEMS Ultrasonic Transducer for Resident Monitoring of Steel Structures. SPIE Smart Structures Conference SS05, San Diego.
- Papadimitriou, C. H., P. Raghavan, et al. (1998). Latent Semantic Indexing: A Probabilistic Analysis. PODS, Seattle, WA.
- Rabiner, L. and B.-H. Juang (1993). Fundamentals of Speech Recognition, Prentice Hall.

SBBD 05 Copyright: C. Faloutsos, 2005 79



CMU SCS

References

- Traina, C., A. Traina, et al. (October 2000). Fast feature selection using the fractal dimension,. XV Brazilian Symposium on Databases (SBBD), Paraiba, Brazil.

SBBD 05 Copyright: C. Faloutsos, 2005 80



CMU SCS

References

- Dennis Shasha and Yunyue Zhu *High Performance Discovery in Time Series: Techniques and Case Studies* Springer 2004
- Yunyue Zhu, Dennis Shasha ``StatStream: Statistical Monitoring of Thousands of Data Streams in Real Time'' VLDB, August, 2002. pp. 358-369.
- Samuel R. Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong. *The Design of an Acquisitional Query Processor for Sensor Networks*. SIGMOD, June 2003, San Diego, CA.

SBBD 05 Copyright: C. Faloutsos, 2005 81



CMU SCS

Part 2: DSP (Digital Signal Processing)

SBBD 05 Copyright: C. Faloutsos, 2005 82



CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
- ➡ • DSP (DFT, DWT)
- Linear Forecasting
- Bursty traffic - fractals and multifractals
- Non-linear forecasting
- Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 83



CMU SCS

Outline

- ➡ • DFT
 - Definition of DFT and properties
 - how to read the DFT spectrum
- DWT
 - Definition of DWT and properties
 - how to read the DWT scalogram

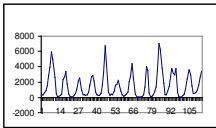
SBBD 05 Copyright: C. Faloutsos, 2005 84

CMU SCS

Introduction - Problem#1

Goal: given a signal (eg., packets over time)
Find: patterns and/or compress

count



lynx caught per year
(packets per day;
automobiles per hour)

year

SBBD 05 Copyright: C. Faloutsos, 2005 85

CMU SCS

What does DFT do?

A: highlights the periodicities



SBBD 05 Copyright: C. Faloutsos, 2005 86

CMU SCS

DFT: definition

Skip

- For a sequence $x_0, x_1, \dots, x_{n-1}$
- the (**n-point**) Discrete Fourier Transform is
- $X_0, X_1, \dots, X_{n-1}$:

$$X_f = 1/\sqrt{n} \sum_{t=0}^{n-1} x_t * \exp(-j2\pi tf/n) \quad f = 0, \dots, n-1$$

($j = \sqrt{-1}$)

$$x_t = 1/\sqrt{n} \sum_{f=0}^{n-1} X_f * \exp(+j2\pi tf/n) \quad \text{inverse DFT}$$

SBBD 05 Copyright: C. Faloutsos, 2005 87

CMU SCS

DFT: definition

- Good news:** Available in **all** symbolic math packages, eg., in ‘mathematica’

```
x = [1,2,1,2];
X = Fourier[x];
Plot[ Abs[X] ];
```

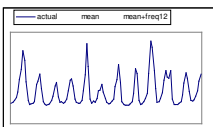
SBBD 05 Copyright: C. Faloutsos, 2005 88

CMU SCS

DFT: Amplitude spectrum

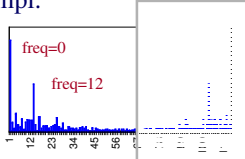
Amplitude: $A_f^2 = \text{Re}^2(X_f) + \text{Im}^2(X_f)$

count



year

Ampl.



freq=0
freq=12

Freq.

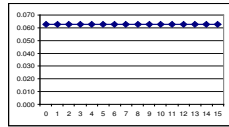
SBBD 05 Copyright: C. Faloutsos, 2005 89

CMU SCS

DFT: examples

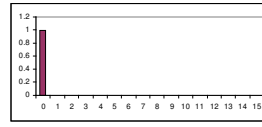
Skip

flat



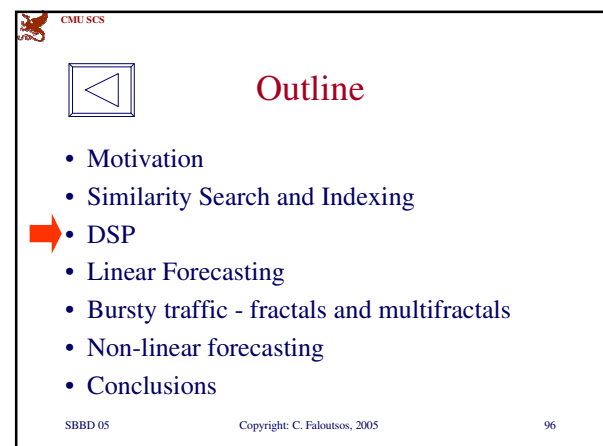
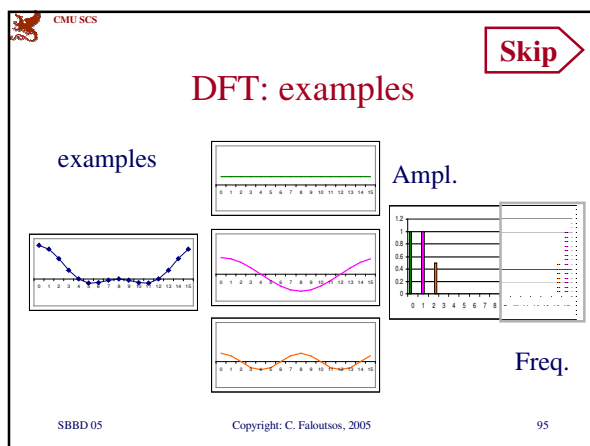
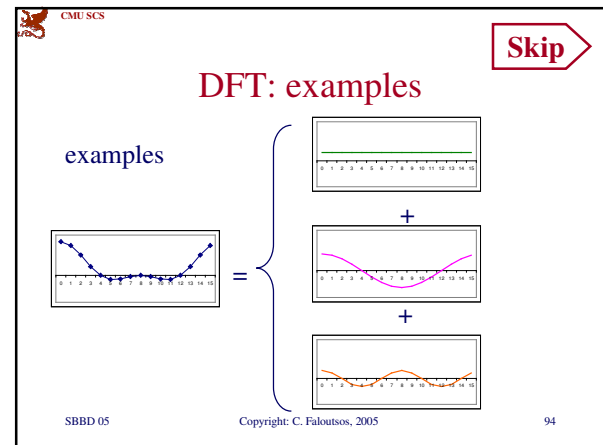
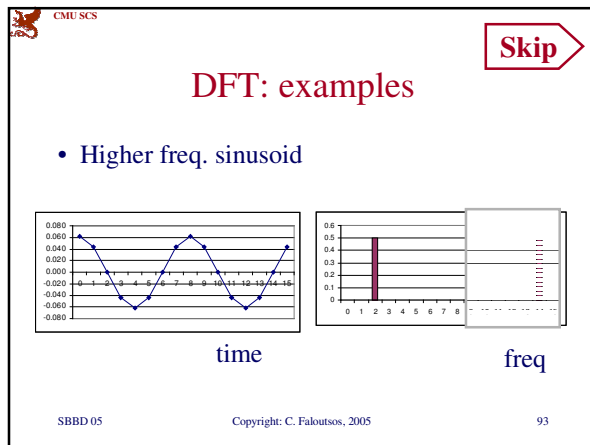
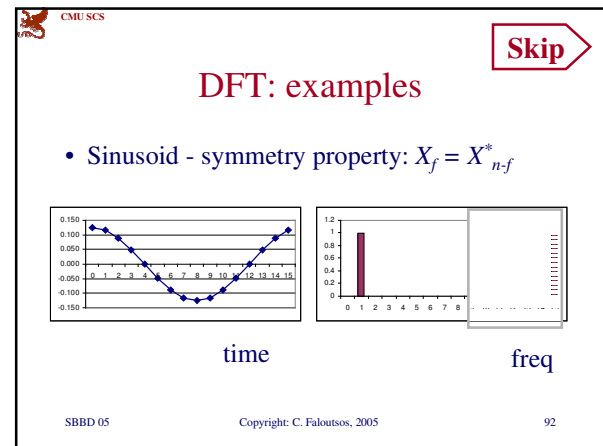
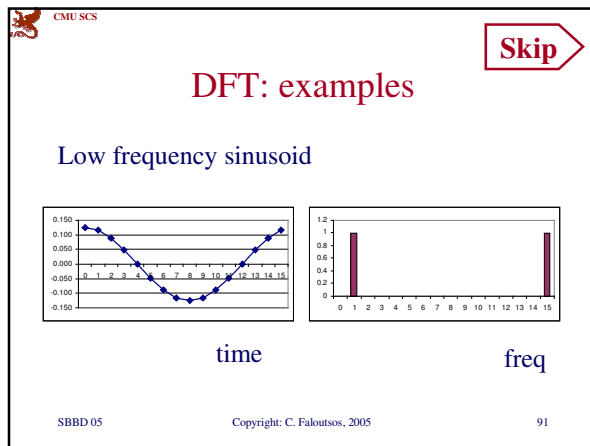
time

Amplitude



freq

SBBD 05 Copyright: C. Faloutsos, 2005 90



CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
- DSP
 - DFT
 - Definition of DFT and properties
 - how to read the DFT spectrum
 - DWT

SBBD 05 Copyright: C. Faloutsos, 2005 97

CMU SCS

DFT: Amplitude spectrum

Amplitude: $A_f^2 = \text{Re}^2(X_f) + \text{Im}^2(X_f)$

count

Ampl.

year

freq=0

freq=12

Freq.

SBBD 05 Copyright: C. Faloutsos, 2005 98

CMU SCS

DFT: Amplitude spectrum

count

Ampl.

year

freq=0

freq=12

Freq.

SBBD 05 Copyright: C. Faloutsos, 2005 99

CMU SCS

DFT: Amplitude spectrum

count

Ampl.

year

freq=0

freq=12

Freq.

SBBD 05 Copyright: C. Faloutsos, 2005 100

CMU SCS

DFT: Amplitude spectrum

- excellent approximation, with only 2 frequencies!
- so what?

actual mean mean+freq12

year

freq=0

freq=12

Freq.

SBBD 05 Copyright: C. Faloutsos, 2005 101

CMU SCS

DFT: Amplitude spectrum

- excellent approximation, with only 2 frequencies!
- so what?
- A1: (lossy) compression
- A2: pattern discovery

actual mean mean+freq12

year

freq=0

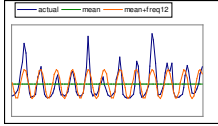
freq=12

Freq.

SBBD 05 Copyright: C. Faloutsos, 2005 102

DFT: Amplitude spectrum

- excellent approximation, with only 2 frequencies!
- so what?
- A1: (lossy) compression
- A2: **pattern discovery**



SBBD 05 Copyright: C. Faloutsos, 2005 103

DFT - Conclusions

- It spots periodicities (with the **'amplitude spectrum'**)
- can be quickly computed ($O(n \log n)$), thanks to the FFT algorithm.
- **standard** tool in signal processing (speech, image etc signals)
- (closely related to DCT and JPEG)

SBBD 05 Copyright: C. Faloutsos, 2005 104

Outline

- Motivation
- Similarity Search and Indexing
- DSP
 - DFT
 - DWT

➔

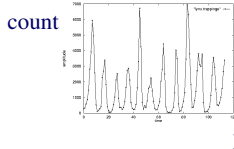
- Definition of DWT and properties
- how to read the DWT scalogram

SBBD 05 Copyright: C. Faloutsos, 2005 105

Problem #1:

Goal: given a signal (eg., #packets over time)

Find: patterns, periodicities, and/or **compress**

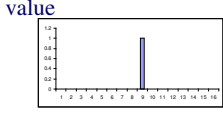


lynx caught per year
(packets per day;
virus infections per month)

SBBD 05 Copyright: C. Faloutsos, 2005 106

Wavelets - DWT

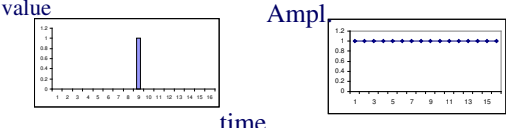
- DFT is great - but, how about compressing a spike?



SBBD 05 Copyright: C. Faloutsos, 2005 107

Wavelets - DWT

- DFT is great - but, how about compressing a spike?
- A: Terrible - all DFT coefficients needed!



SBBD 05 Copyright: C. Faloutsos, 2005 108

Wavelets - DWT

- DFT is great - but, how about compressing a spike?
- A: Terrible - all DFT coefficients needed!

value

time

SBBD 05 Copyright: C. Faloutsos, 2005 109

Wavelets - DWT

- Similarly, DFT suffers on short-duration waves (eg., baritone, silence, soprano)

value

time

SBBD 05 Copyright: C. Faloutsos, 2005 110

Wavelets - DWT

- Solution#1: Short window Fourier transform (SWFT)
- But: how short should be the window?

freq

value

time

SBBD 05 Copyright: C. Faloutsos, 2005 111

Wavelets - DWT

- Answer: **multiple** window sizes! -> DWT

Time domain

DFT

SWFT

DWT

freq

time

SBBD 05 Copyright: C. Faloutsos, 2005 112

Haar Wavelets

- subtract sum of left half from right half
- repeat recursively for quarters, eight-ths, ...

SBBD 05 Copyright: C. Faloutsos, 2005 113

Wavelets - construction

Skip

x0 x1 x2 x3 x4 x5 x6 x7

SBBD 05 Copyright: C. Faloutsos, 2005 114

CMU SCS

Wavelets - construction

Skip

level 1 $d_{1,0}$ $s_{1,0}$ $d_{1,1}$ $s_{1,1}$

x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7

SBBD 05 Copyright: C. Faloutsos, 2005 115

CMU SCS

Wavelets - construction

Skip

level 2 $d_{2,0}$ $s_{2,0}$ $d_{1,0}$ $s_{1,0}$ $d_{1,1}$ $s_{1,1}$

x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7

SBBD 05 Copyright: C. Faloutsos, 2005 116

CMU SCS

Wavelets - construction

Skip

etc ...

$d_{2,0}$ $s_{2,0}$ $d_{1,0}$ $s_{1,0}$ $d_{1,1}$ $s_{1,1}$

x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7

SBBD 05 Copyright: C. Faloutsos, 2005 117

CMU SCS

Wavelets - construction

Q: map each coefficient on the time-freq. plane

f

t

$d_{2,0}$ $s_{2,0}$ $d_{1,0}$ $s_{1,0}$ $d_{1,1}$ $s_{1,1}$

x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7

SBBD 05 Copyright: C. Faloutsos, 2005 118

CMU SCS

Wavelets - construction

Q: map each coefficient on the time-freq. plane

f

t

$d_{2,0}$ $s_{2,0}$ $d_{1,0}$ $s_{1,0}$ $d_{1,1}$ $s_{1,1}$

x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7

SBBD 05 Copyright: C. Faloutsos, 2005 119

CMU SCS

Haar wavelets - code

```
#usr/bin/perl5
# expects a file with numbers
# and prints the dwt transform
# The number of time-ticks should be a power of 2
# USAGE
# haar.pl <fname>

my @vals=();
my @smooth; # the smooth component of the signal
my @diff; # the high-freq. component

# collect the values into the array @val
while(<)>{
    @vals = ( @vals , split );
}

my $len = scalar(@vals);
my $half = int($len/2);
while($half >= 1){
    for(my $i=0; $i< $half; $i++){
        $diff[$i] = ($vals[2*$i] - $vals[2*$i + 1]) / sqrt(2);
        print "d", $diff[$i];
        $smooth[$i] = ($vals[2*$i] + $vals[2*$i + 1]) / sqrt(2);
    }
    print "u";
    @vals = @smooth;
    $half = int($half/2);
}
print "u", $vals[0], "u"; # the final, smooth component
```

SBBD 05 Copyright: C. Faloutsos, 2005 120

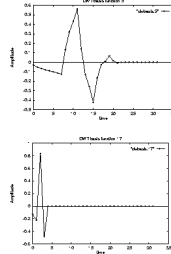
Wavelets - construction

Observation1:
 '+' can be some weighted addition
 '-' is the corresponding weighted difference
 ('Quadrature mirror filters')

Observation2: unlike DFT/DCT,
 there are *many* wavelet bases: Haar, Daubechies-4, Daubechies-6, Coifman, Morlet, Gabor, ...

SBBD 05 Copyright: C. Faloutsos, 2005 121

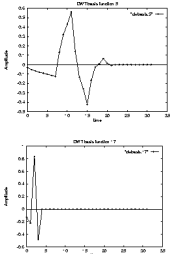
Wavelets - how do they look like?



- E.g., Daubechies-4

SBBD 05 Copyright: C. Faloutsos, 2005 122

Wavelets - how do they look like?

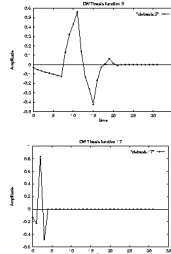


- E.g., Daubechies-4



SBBD 05 Copyright: C. Faloutsos, 2005 123

Wavelets - how do they look like?



- E.g., Daubechies-4



SBBD 05 Copyright: C. Faloutsos, 2005 124

Outline

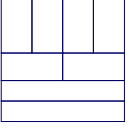
- Motivation
- Similarity Search and Indexing
- DSP
 - DFT
 - DWT
 - Definition of DWT and properties
 - how to read the DWT scalogram

➔

SBBD 05 Copyright: C. Faloutsos, 2005 125

Wavelets - Drill#1:

- Q: baritone/silence/soprano - DWT?



f

t

value



time

SBBD 05 Copyright: C. Faloutsos, 2005 126

Wavelets - Drill#1:

- Q: baritone/soprano - DWT?

SBBD 05 Copyright: C. Faloutsos, 2005 127

Wavelets - Drill#2:

- Q: spike - DWT?

SBBD 05 Copyright: C. Faloutsos, 2005 128

Wavelets - Drill#2:

- Q: spike - DWT?

0.00	0.00	0.71	0.00
0.00	0.50	-0.35	0.35

SBBD 05 Copyright: C. Faloutsos, 2005 129

Wavelets - Drill#3:

- Q: weekly + daily periodicity, + spike - DWT?

SBBD 05 Copyright: C. Faloutsos, 2005 130

Wavelets - Drill#3:

- Q: **weekly** + daily periodicity, + spike - DWT?

SBBD 05 Copyright: C. Faloutsos, 2005 131

Wavelets - Drill#3:

- Q: weekly + **daily** periodicity, + spike - DWT?

SBBD 05 Copyright: C. Faloutsos, 2005 132

CMU SCS

Wavelets - Drill#3:

- Q: weekly + daily periodicity, + **spike** - DWT?

SBBD 05 Copyright: C. Faloutsos, 2005 133

CMU SCS

Wavelets - Drill#3:

- Q: weekly + daily periodicity, + spike - DWT?

SBBD 05 Copyright: C. Faloutsos, 2005 134

CMU SCS

Wavelets - Drill#3:

- Q: DFT?

SBBD 05 Copyright: C. Faloutsos, 2005 135

CMU SCS

Advantages of Wavelets

- Better compression (better RMSE with same number of coefficients - used in JPEG-2000)
- fast to compute (usually: $O(n)!$)
- very good for 'spikes'
- mammalian eye and ear: Gabor wavelets

SBBD 05 Copyright: C. Faloutsos, 2005 136

CMU SCS

Overall Conclusions

- DFT, DCT spot periodicities
- DWT** : multi-resolution - matches processing of mammalian ear/eye better
- All three: powerful tools for **compression**, **pattern detection** in real signals
- All three: included in math packages
 - (matlab, 'R', mathematica, ... - often in spreadsheets!)

SBBD 05 Copyright: C. Faloutsos, 2005 137

CMU SCS

Overall Conclusions

- DWT : very suitable for self-similar traffic
- DWT: used for summarization of streams [Gilbert+01], db histograms etc

SBBD 05 Copyright: C. Faloutsos, 2005 138

 CMU SCS

Resources - software and urls

- <http://www.dsptutor.freeuk.com/jsanalyser/FFTSpectrumAnalyser.html> : Nice java applets for FFT
- <http://www.relisoft.com/freeware/freq.html> voice frequency analyzer (needs microphone)

SBBD 05 Copyright: C. Faloutsos, 2005 139

 CMU SCS

Resources: software and urls

- *xwpl*: open source wavelet package from Yale, with excellent GUI
- <http://monet.me.ic.ac.uk/people/gavin/java/waveletDemos.html> : wavelets and scalograms

SBBD 05 Copyright: C. Faloutsos, 2005 140

 CMU SCS

Books

- William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery: *Numerical Recipes in C*, Cambridge University Press, 1992, 2nd Edition. (Great description, intuition and code for DFT, DWT)
- C. Faloutsos: *Searching Multimedia Databases by Content*, Kluwer Academic Press, 1996 (introduction to DFT, DWT)

SBBD 05 Copyright: C. Faloutsos, 2005 141

 CMU SCS

Additional Reading

- [Gilbert+01] Anna C. Gilbert, Yannis Kotidis and S. Muthukrishnan and Martin Strauss, *Surfing Wavelets on Streams: One-Pass Summaries for Approximate Aggregate Queries*, VLDB 2001

SBBD 05 Copyright: C. Faloutsos, 2005 142

 CMU SCS

BREAK!

SBBD 05 Copyright: C. Faloutsos, 2005 143

 CMU SCS

Indexing and Mining Streams

Christos Faloutsos
CMU

CMU SCS

Part 3: Linear Forecasting

SBBD 05 Copyright: C. Faloutsos, 2005 145

CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
- DSP
- ➔ • Linear Forecasting
 - Bursty traffic - fractals and multifractals
 - Non-linear forecasting
- Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 146

CMU SCS

Forecasting

"Prediction is very difficult, especially about the future." - Nils Bohr

<http://www.hfac.uh.edu/MediaFutures/thoughts.html>

SBBD 05 Copyright: C. Faloutsos, 2005 147

CMU SCS

Outline

- Motivation
- ...
- ➔ • Linear Forecasting
 - Auto-regression: Least Squares; RLS
 - Co-evolving time sequences
 - Examples
 - Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 148

CMU SCS

Problem#2: Forecast

- Example: give $x_{t-1}, x_{t-2}, \dots$, forecast x_t

SBBD 05 Copyright: C. Faloutsos, 2005 149

CMU SCS

Forecasting: Preprocessing

MANUALLY:

remove trends

spot periodicities
7 days

time time

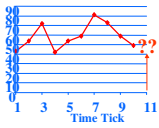
SBBD 05 Copyright: C. Faloutsos, 2005 150

CMU SCS

Problem#2: Forecast

- Solution: try to express x_t as a linear function of the past: $x_{t-2}, x_{t-3}, \dots$ (up to a window of w)
- Formally:

$$x_t \approx a_1 x_{t-1} + \dots + a_w x_{t-w} + \text{noise}$$

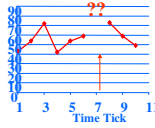


SBBD 05 Copyright: C. Faloutsos, 2005 151

CMU SCS

(Problem: Back-cast; interpolate)

- Solution - interpolate: try to express x_t as a linear function of the past AND the future: $x_{t+1}, x_{t+2}, \dots, x_{t+w_{\text{future}}}; x_{t-1}, \dots, x_{t-w_{\text{past}}}$ (up to windows of $w_{\text{past}}, w_{\text{future}}$)
- EXACTLY the same algo's

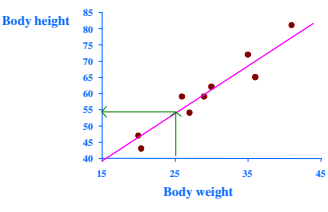


SBBD 05 Copyright: C. Faloutsos, 2005 152

CMU SCS

Linear Regression: idea

patient	weight	height
1	27	43
2	43	54
3	54	72
...	...	...
N	25	??



- express what we don't know (= 'dependent variable')
- as a linear function of what we know (= 'indep. variable(s)')

SBBD 05 Copyright: C. Faloutsos, 2005 153

CMU SCS

Linear Auto Regression:

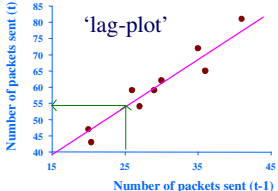
Time	Packets Sent(t)
1	43
2	54
3	72
...	...
N	??

SBBD 05 Copyright: C. Faloutsos, 2005 154

CMU SCS

Linear Auto Regression:

Time	Packets Sent (t-1)	Packets Sent(t)
1	-	43
2	43	54
3	54	72
...	...	...
N	25	??



- lag $w=1$
- Dependent variable = # of packets sent ($S[t]$)
- Independent variable = # of packets sent ($S[t-1]$)

SBBD 05 Copyright: C. Faloutsos, 2005 155

CMU SCS

Outline

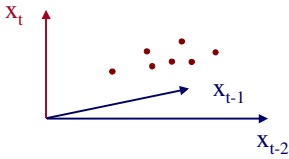
- Motivation
- ...
- Linear Forecasting
 - Auto-regression: **Least Squares; RLS**
 - Co-evolving time sequences
 - Examples
 - Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 156

CMU SCS

More details:

- Q1: Can it work with window $w > 1$?
- A1: YES!

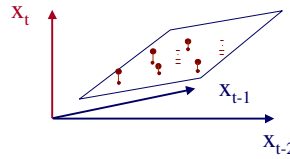


SBBD 05 Copyright: C. Faloutsos, 2005 157

CMU SCS

More details:

- Q1: Can it work with window $w > 1$?
- A1: YES! (we'll fit a hyper-plane, then!)

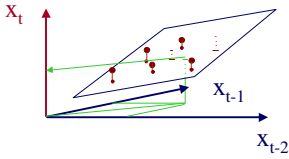


SBBD 05 Copyright: C. Faloutsos, 2005 158

CMU SCS

More details:

- Q1: Can it work with window $w > 1$?
- A1: YES! (we'll fit a hyper-plane, then!)



SBBD 05 Copyright: C. Faloutsos, 2005 159

CMU SCS

More details:

Skip

- Q1: Can it work with window $w > 1$?
- A1: YES! The problem becomes:

$$\mathbf{X}_{[N \times w]} \times \mathbf{a}_{[w \times 1]} = \mathbf{y}_{[N \times 1]}$$

- OVER-CONSTRAINED
 - $\mathbf{a}$ is the vector of the regression coefficients
 - $\mathbf{X}$ has the N values of the w indep. variables
 - $\mathbf{y}$ has the N values of the dependent variable

SBBD 05 Copyright: C. Faloutsos, 2005 160

CMU SCS

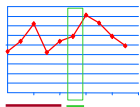
More details:

Skip

- $\mathbf{X}_{[N \times w]} \times \mathbf{a}_{[w \times 1]} = \mathbf{y}_{[N \times 1]}$

Ind-var1 Ind-var-w

time

$$\begin{bmatrix} X_{11} & X_{12} & \dots & X_{1w} \\ X_{21} & X_{22} & \dots & X_{2w} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \dots & X_{Nw} \end{bmatrix} \times \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_w \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$


SBBD 05 Copyright: C. Faloutsos, 2005 161

CMU SCS

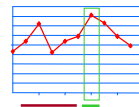
More details:

Skip

- $\mathbf{X}_{[N \times w]} \times \mathbf{a}_{[w \times 1]} = \mathbf{y}_{[N \times 1]}$

Ind-var1 Ind-var-w

time

$$\begin{bmatrix} X_{11} & X_{12} & \dots & X_{1w} \\ X_{21} & X_{22} & \dots & X_{2w} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \dots & X_{Nw} \end{bmatrix} \times \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_w \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$


SBBD 05 Copyright: C. Faloutsos, 2005 162

CMU SCS

More details Skip

- Q2: How to estimate $a_1, a_2, \dots, a_w = \mathbf{a}$?
- A2: with Least Squares fit

$$\mathbf{a} = (\mathbf{X}^T \times \mathbf{X})^{-1} \times (\mathbf{X}^T \times \mathbf{y})$$

- (Moore-Penrose pseudo-inverse)
- $\mathbf{a}$ is the vector that minimizes the RMSE from $\mathbf{y}$

SBBD 05 Copyright: C. Faloutsos, 2005 163

CMU SCS

Even more details Skip

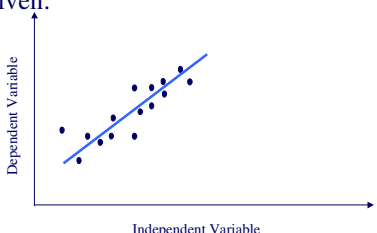
- Q3: Can we estimate $\mathbf{a}$ incrementally?
- A3: Yes, with the brilliant, classic method of 'Recursive Least Squares' (RLS) (see, e.g., [Yi+00], for details) - pictorially:

SBBD 05 Copyright: C. Faloutsos, 2005 164

CMU SCS

Even more details Skip

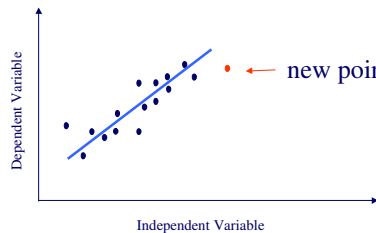
- Given:



SBBD 05 Copyright: C. Faloutsos, 2005 165

CMU SCS

Even more details Skip

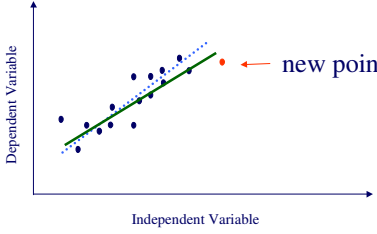


SBBD 05 Copyright: C. Faloutsos, 2005 166

CMU SCS

Even more details Skip

RLS: quickly compute new best fit



SBBD 05 Copyright: C. Faloutsos, 2005 167

CMU SCS

Even more details Skip

• Straightforward Least Squares – Needs huge matrix (growing in size) $O(N \times w)$ – Costly matrix operation $O(N \times w^2)$	• Recursive LS – Need much smaller, fixed size matrix $O(w \times w)$ – Fast, incremental computation $O(1 \times w^2)$
---	---

$N = 10^6, \quad w = 1-100$

SBBD 05 Copyright: C. Faloutsos, 2005 168

CMU SCS

Even more details

Skip

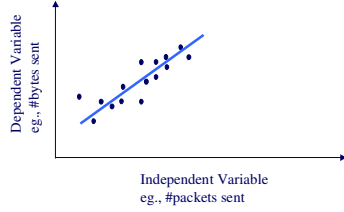
- Q4: can we ‘forget’ the older samples?
- A4: Yes - RLS can easily handle that [Yi+00]:

SBBD 05 Copyright: C. Faloutsos, 2005 169

CMU SCS

Adaptability - ‘forgetting’

Skip

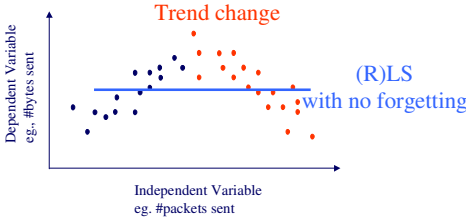


SBBD 05 Copyright: C. Faloutsos, 2005 170

CMU SCS

Adaptability - ‘forgetting’

Skip

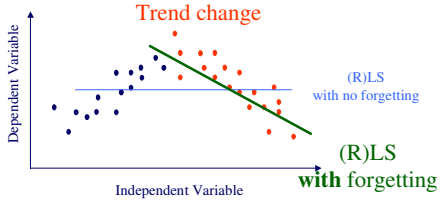


SBBD 05 Copyright: C. Faloutsos, 2005 171

CMU SCS

Adaptability - ‘forgetting’

Skip



- RLS: can *trivially* handle ‘forgetting’

SBBD 05 Copyright: C. Faloutsos, 2005 172

CMU SCS

How to choose ‘w’?

- goal: capture arbitrary periodicities
- with NO human intervention
- on a semi-infinite stream

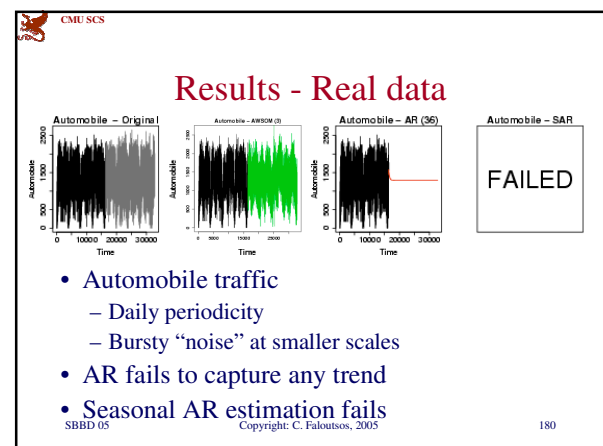
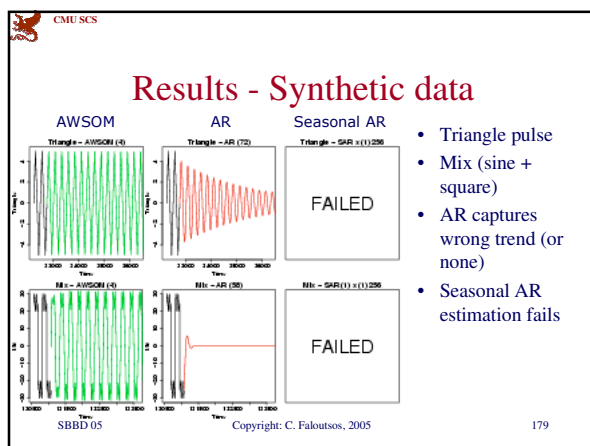
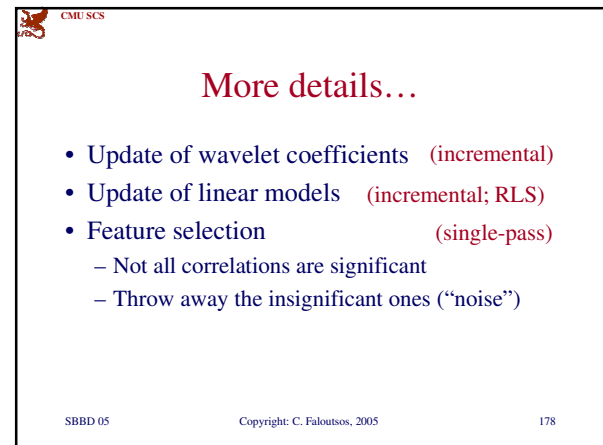
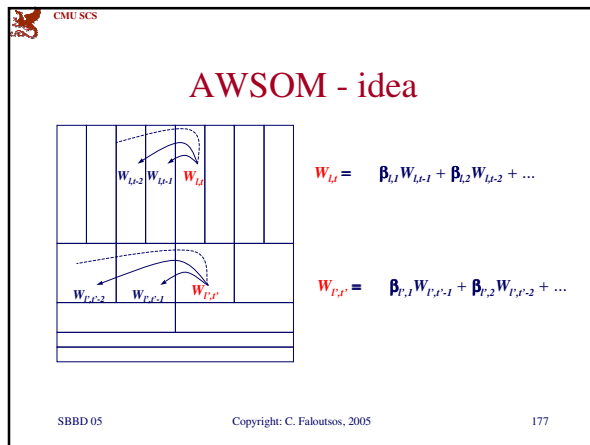
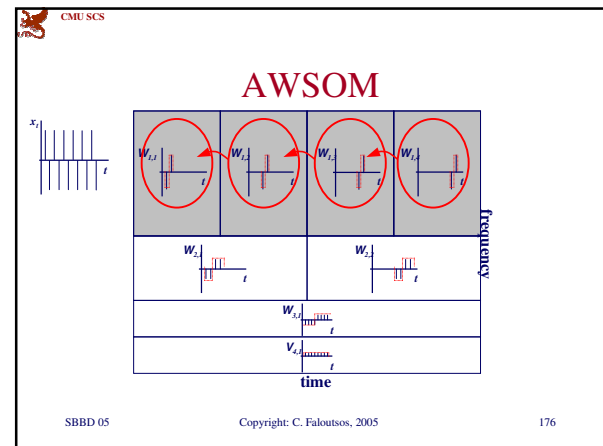
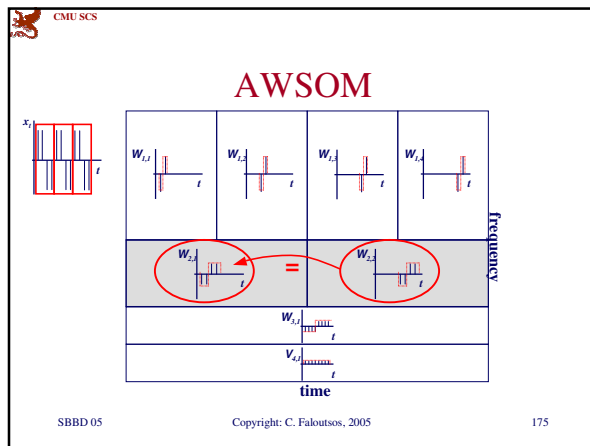
SBBD 05 Copyright: C. Faloutsos, 2005 173

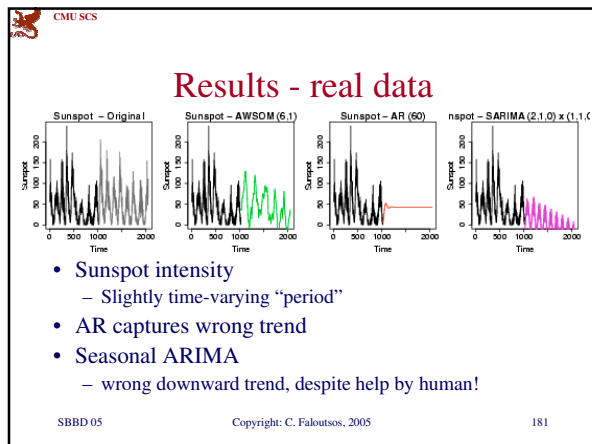
CMU SCS

Answer:

- ‘AWSOM’ (Arbitrary Window Stream forecasting Method) [Papadimitriou+, vldb2003]
- idea: do AR on each wavelet level
- in detail:

SBBD 05 Copyright: C. Faloutsos, 2005 174





CMU SCS

Skip

Complexity

- Model update
 - Space: $O(\lg N + mk^2) \approx O(\lg N)$
 - Time: $O(k^2) \approx O(1)$
- Where
 - N : number of points (so far)
 - k : number of regression coefficients; fixed
 - m : number of linear models; $O(\lg N)$

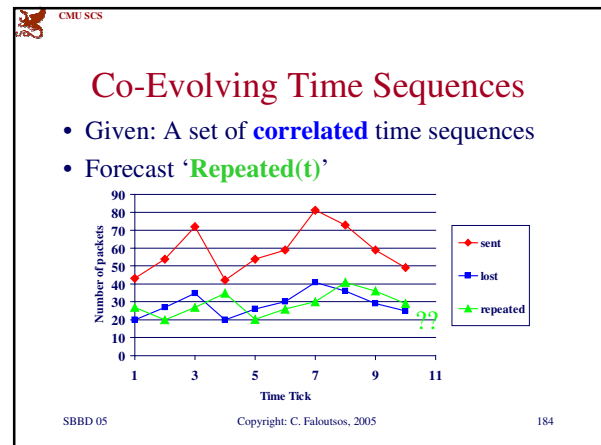
SBBD 05 Copyright: C. Faloutsos, 2005 182

CMU SCS

Outline

- Motivation
- ...
- Linear Forecasting
 - Auto-regression: Least Squares; RLS
 - Co-evolving time sequences
 - Examples
 - Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 183



CMU SCS

Solution:

Q: what should we do?

SBBD 05 Copyright: C. Faloutsos, 2005 185

CMU SCS

Solution:

Least Squares, with

- Dep. Variable: Repeated(t)
- Indep. Variables: Sent(t-1) ... Sent(t-w); Lost(t-1) ... Lost(t-w); Repeated(t-1), ...
- (named: ‘MUSCLES’ [Yi+00])

SBBD 05 Copyright: C. Faloutsos, 2005 186

CMU SCS

B.II - Time Series Analysis Outline

Skip

- Auto-regression
- Least Squares; recursive least squares
- Co-evolving time sequences
- ➔ • Examples
- Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 187

CMU SCS

Examples - Experiments

Skip

- Datasets
 - Modem pool traffic (14 modems, 1500 time-ticks; #packets per time unit)
 - AT&T WorldNet internet usage (several data streams; 980 time-ticks)
- Measures of success
 - Accuracy : Root Mean Square Error (RMSE)

SBBD 05 Copyright: C. Faloutsos, 2005 188

CMU SCS

Accuracy - "Modem"

Skip

RMSE

Modems

AR yesterday MUSCLES

MUSCLES outperforms AR & "yesterday"

SBBD 05 Copyright: C. Faloutsos, 2005 189

CMU SCS

Accuracy - "Internet"

Skip

RMSE

Streams

AR yesterday MUSCLES

MUSCLES consistently outperforms AR & "yesterday"

SBBD 05 Copyright: C. Faloutsos, 2005 190

CMU SCS

B.II - Time Series Analysis Outline

Skip

- Auto-regression
- Least Squares; recursive least squares
- Co-evolving time sequences
- Examples
- ➔ • Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 191

CMU SCS

Conclusions - Practitioner's guide

- AR(IMA) methodology: prevailing method for linear forecasting
- Brilliant method of Recursive Least Squares for fast, incremental estimation.
- See [Box-Jenkins]
- very recently: AWSOM (no human intervention)

SBBD 05 Copyright: C. Faloutsos, 2005 192

 CMU SCS

Resources: software and urls

- MUSCLES: Prof. Byoung-Kee Yi:
<http://www.postech.ac.kr/~bkyi/>
 or christos@cs.cmu.edu
- free-ware: 'R' for stat. analysis
 (clone of Splus)
<http://cran.r-project.org/>

SBBD 05 Copyright: C. Faloutsos, 2005 193

 CMU SCS

Books

- George E.P. Box and Gwilym M. Jenkins and Gregory C. Reinsel, *Time Series Analysis: Forecasting and Control*, Prentice Hall, 1994 (the classic book on ARIMA, 3rd ed.)
- Brockwell, P. J. and R. A. Davis (1987). *Time Series: Theory and Methods*. New York, Springer Verlag.

SBBD 05 Copyright: C. Faloutsos, 2005 194

 CMU SCS

Additional Reading

- [Papadimitriou+ vldb2003] Spiros Papadimitriou, Anthony Brockwell and Christos Faloutsos *Adaptive, Hands-Off Stream Mining* VLDB 2003, Berlin, Germany, Sept. 2003
- [Yi+00] Byoung-Kee Yi et al.: *Online Data Mining for Co-Evolving Time Sequences*, ICDE 2000. (Describes MUSCLES and Recursive Least Squares)

SBBD 05 Copyright: C. Faloutsos, 2005 195

 CMU SCS

Part 4:

Bursty traffic and multifractals

SBBD 05 Copyright: C. Faloutsos, 2005 196

 CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
- DSP
- Linear Forecasting
- ➔ • Bursty traffic - fractals and multifractals
- Non-linear forecasting
- Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 197

 CMU SCS

Outline

- Motivation
- ...
- Linear Forecasting
- ➔ • Bursty traffic - fractals and multifractals
 - Problem
 - Main idea (80/20, Hurst exponent)
 - Results

SBBD 05 Copyright: C. Faloutsos, 2005 198

Recall: Problem #1:

Goal: given a signal (eg., #bytes over time)
Find: patterns, periodicities, and/or compress

#bytes

Bytes per 30'
(packets per day;
earthquakes per year)

time

SBBD 05 Copyright: C. Faloutsos, 2005 199

Problem #1

- model bursty traffic
- generate realistic traces
- (Poisson does not work)

bytes

time

SBBD 05 Copyright: C. Faloutsos, 2005 200

Motivation

- predict queue length distributions (e.g., to give probabilistic guarantees)
- “learn” traffic, for buffering, prefetching, ‘active disks’, web servers

SBBD 05 Copyright: C. Faloutsos, 2005 201

Q: any ‘pattern’?

- Not Poisson
- spike; silence; more spikes; more silence...
- any rules?

bytes

time

SBBD 05 Copyright: C. Faloutsos, 2005 202

solution: self-similarity

bytes

time

time

SBBD 05 Copyright: C. Faloutsos, 2005 203

But:

- Q1: How to generate realistic traces; extrapolate; give guarantees?
- Q2: How to estimate the model parameters?

SBBD 05 Copyright: C. Faloutsos, 2005 204

CMU SCS

Outline

- Motivation
- ...
- Linear Forecasting
- Bursty traffic - fractals and multifractals
 - Problem
 - ➔ – Main idea (80/20, Hurst exponent)
 - Results

SBBD 05 Copyright: C. Faloutsos, 2005 205

CMU SCS

Approach

- Q1: How to generate a sequence, that is
 - bursty
 - self-similar
 - and has similar queue length distributions

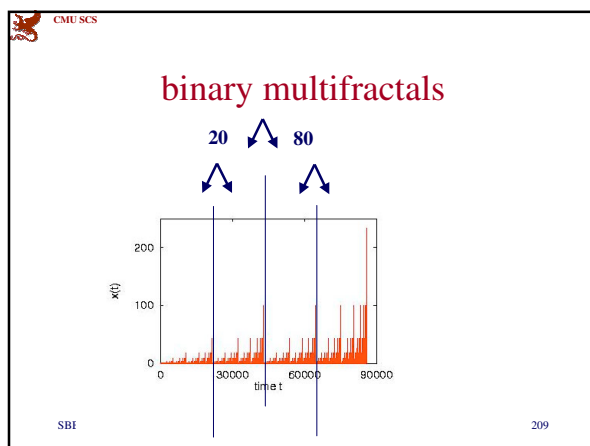
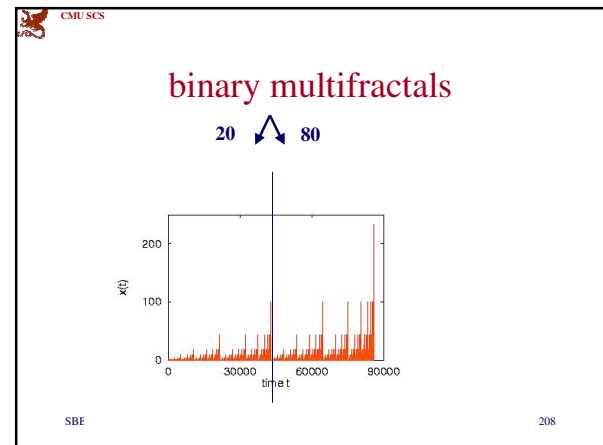
SBBD 05 Copyright: C. Faloutsos, 2005 206

CMU SCS

Approach

- A: ‘binomial multifractal’ [Wang+02]
- ~ 80-20 ‘law’:
 - 80% of bytes/queries etc on first half
 - repeat recursively
- b : bias factor (eg., 80%)

SBBD 05 Copyright: C. Faloutsos, 2005 207



CMU SCS

Parameter estimation

- Q2: How to estimate the bias factor b ?

SBBD 05 Copyright: C. Faloutsos, 2005 210

CMU SCS

Parameter estimation

- Q2: How to estimate the bias factor b ?
- A: MANY ways [Crovetlla+96]
 - Hurst exponent
 - variance plot
 - even DFT amplitude spectrum! ('periodogram')
 - More robust: 'entropy plot' [Wang+02]

SBBD 05 Copyright: C. Faloutsos, 2005 211

CMU SCS

Entropy plot

- Rationale:
 - burstiness: inverse of uniformity
 - entropy measures uniformity of a distribution
 - find entropy at several granularities, to see whether/how our distribution is close to uniform.

SBBD 05 Copyright: C. Faloutsos, 2005 212

CMU SCS

Entropy plot

p1 p2

% of bytes here

- Entropy $E(n)$ after n levels of splits
- $n=1$: $E(1) = -p_1 \log_2(p_1) - p_2 \log_2(p_2)$

SBBD 05 Copyright: C. Faloutsos, 2005 213

CMU SCS

Entropy plot

p_{2,1} p_{2,2} p_{2,3} p_{2,4}

- Entropy $E(n)$ after n levels of splits
- $n=1$: $E(1) = -p_1 \log_2(p_1) - p_2 \log_2(p_2)$
- $n=2$: $E(2) = -\sum_i p_{2,i} \log_2(p_{2,i})$

SBBD 05 Copyright: C. Faloutsos, 2005 214

CMU SCS

Real traffic

Entropy $E(n)$

0.73

Real $0.73 \times n - 0.24$

of levels (n)

- Has linear entropy plot ($\rightarrow$ self-similar)

SBBD 05 Copyright: C. Faloutsos, 2005 215

CMU SCS

Observation - intuition:

Skip

Entropy $E(n)$

0.73

Real $0.73 \times n - 0.24$

of levels (n)

intuition: slope = intrinsic dimensionality = info-bits per coordinate-bit

- unif. Dataset: slope = 1
- multi-point: slope = 0

SBBD 05 Copyright: C. Faloutsos, 2005 216

CMU SCS

Entropy plot - Intuition Skip

- Slope ~ intrinsic dimensionality (in fact, 'Information fractal dimension')
- = info bit per coordinate bit - eg

Dim = 1 

Pick a point;
reveal its coordinate bit-by-bit -
how much info is each bit worth to me?

SBBD 05 Copyright: C. Faloutsos, 2005 217

CMU SCS

Entropy plot Skip

- Slope ~ intrinsic dimensionality (in fact, 'Information fractal dimension')
- = info bit per coordinate bit - eg

Dim = 1 

Is MSB 0?
'info' value = $E(1)$: 1 bit

SBBD 05 Copyright: C. Faloutsos, 2005 218

CMU SCS

Entropy plot Skip

- Slope ~ intrinsic dimensionality (in fact, 'Information fractal dimension')
- = info bit per coordinate bit - eg

Dim = 1 

Is MSB 0?
Is next MSB = 0?

SBBD 05 Copyright: C. Faloutsos, 2005 219

CMU SCS

Entropy plot Skip

- Slope ~ intrinsic dimensionality (in fact, 'Information fractal dimension')
- = info bit per coordinate bit - eg

Dim = 1 

Info value = 1 bit
 $= E(2) - E(1) =$
slope!
Is next MSB = 0?

SBBD 05 Copyright: C. Faloutsos, 2005 220

CMU SCS

Entropy plot Skip

- Repeat, for all points at same position:

Dim=0 

SBBD 05 Copyright: C. Faloutsos, 2005 221

CMU SCS

Entropy plot Skip

- Repeat, for all points at same position:
- we need 0 bits of info, to determine position
- $\rightarrow$ slope = 0 = intrinsic dimensionality

Dim=0 

SBBD 05 Copyright: C. Faloutsos, 2005 222

CMU SCS

Skip

Entropy plot

- Real (and 80-20) datasets can be in-between: bursts, gaps, smaller bursts, smaller gaps, at every scale

Dim = 1 

Dim=0 

$0 < \text{Dim} < 1$ 

SBBD 05 Copyright: C. Faloutsos, 2005 223

CMU SCS

(Fractals, again)

- What set of points could have behavior between point and line?

SBBD 05 Copyright: C. Faloutsos, 2005 224

CMU SCS

Cantor dust

- Eliminate the middle third
- Recursively!

SBBD 05 Copyright: C. Faloutsos, 2005 225

CMU SCS

Cantor dust



SBBD 05 Copyright: C. Faloutsos, 2005 226

CMU SCS

Cantor dust



SBBD 05 Copyright: C. Faloutsos, 2005 227

CMU SCS

Cantor dust



SBBD 05 Copyright: C. Faloutsos, 2005 228

CMU SCS

Cantor dust



SBBD 05 Copyright: C. Faloutsos, 2005 229

CMU SCS

Cantor dust



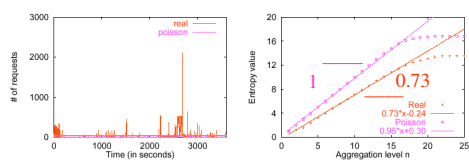
Dimensionality?
(no length; infinite # points!)
Answer: $\log_2 / \log_3 = 0.6$

SBBD 05 Copyright: C. Faloutsos, 2005 230

CMU SCS

Some more entropy plots:

- Poisson vs real

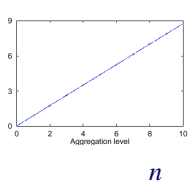


Poisson: slope = ~ 1 $\rightarrow$ uniformly distributed

SBBD 05 Copyright: C. Faloutsos, 2005 231

CMU SCS

B-model



- b-model traffic gives perfectly linear plot
- Lemma: its slope is $\text{slope} = -b \log_2 b - (1-b) \log_2 (1-b)$
- Fitting: do entropy plot; get slope; solve for b

SBBD 05 Copyright: C. Faloutsos, 2005 232

CMU SCS

Outline

- Motivation
- ...
- Linear Forecasting
- Bursty traffic - fractals and multifractals
 - Problem
 - Main idea (80/20, Hurst exponent)
 - ➡ – Experiments - Results

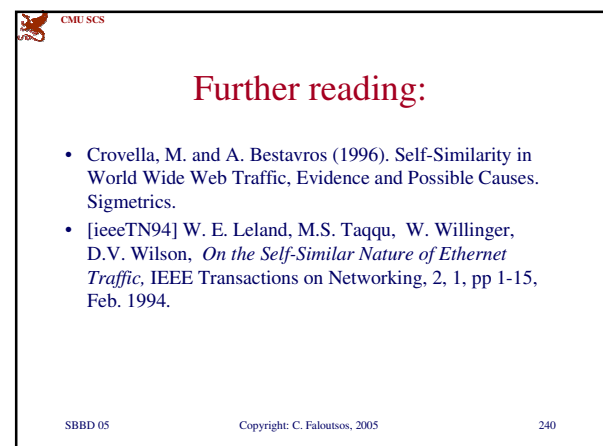
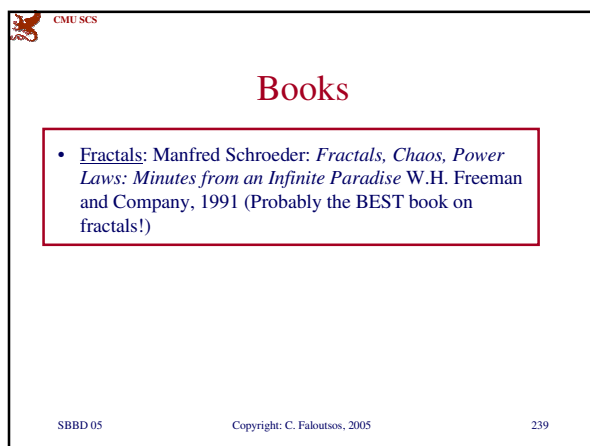
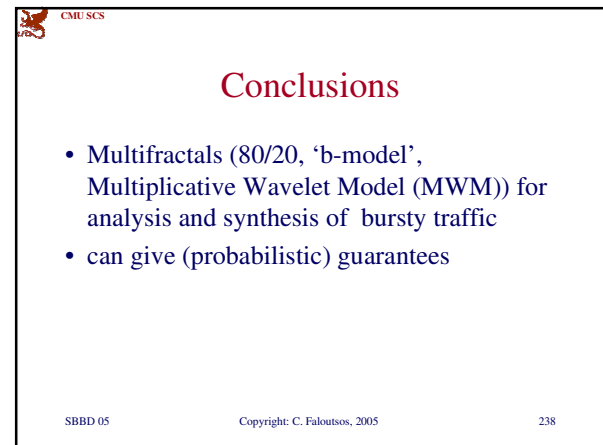
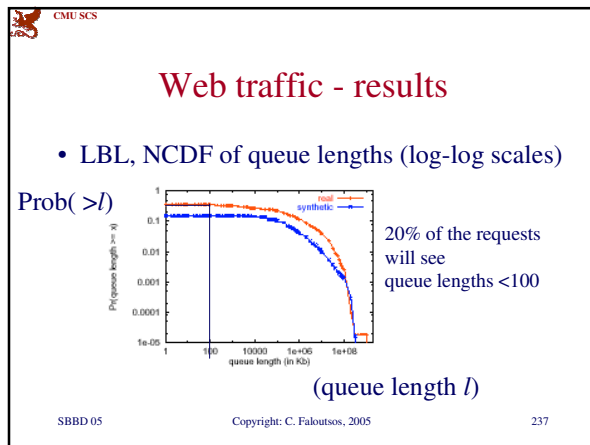
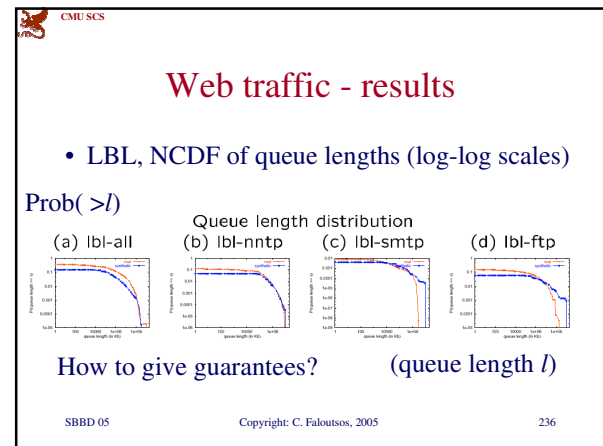
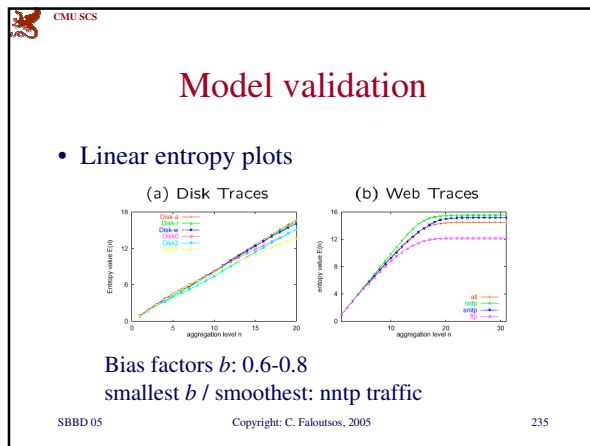
SBBD 05 Copyright: C. Faloutsos, 2005 233

CMU SCS

Experimental setup

- Disk traces (from HP [Wilkes 93])
- web traces from LBL
 - <http://repository.cs.vt.edu/lbl-conn-7.tar.Z>

SBBD 05 Copyright: C. Faloutsos, 2005 234



CMU SCS

Further reading

- [Riedi+99] R. H. Riedi, M. S. Crouse, V. J. Ribeiro, and R. G. Baraniuk, *A Multifractal Wavelet Model with Application to Network Traffic*, IEEE Special Issue on Information Theory, 45. (April 1999), 992-1018.
- [Wang+02] Mengzhi Wang, Tara Madhyastha, Ngai Hang Chang, Spiros Papadimitriou and Christos Faloutsos, *Data Mining Meets Performance Evaluation: Fast Algorithms for Modeling Bursty Traffic*, ICDE 2002, San Jose, CA, 2/26/2002 - 3/1/2002.

SBBD 05 Copyright: C. Faloutsos, 2005 241

CMU SCS

Part 5: chaos and non-linear forecasting

SBBD 05 Copyright: C. Faloutsos, 2005 242

CMU SCS

Outline

- Motivation
- Similarity Search and Indexing
- DSP
- Linear Forecasting
- Bursty traffic - fractals and multifractals
- ➔ • Non-linear forecasting
- Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 243

CMU SCS

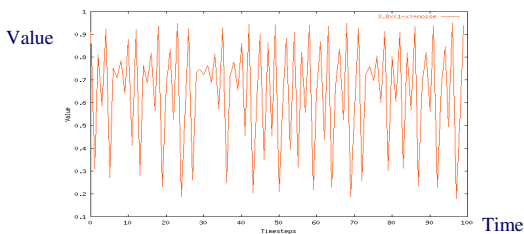
Detailed Outline

- Non-linear forecasting
 - Problem
 - Idea
 - How-to
 - Experiments
 - Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 244

CMU SCS

Recall: Problem #1



Value

Time

Given a time series $\{x_t\}$, predict its future course, that is, x_{t+1} , x_{t+2} , ...

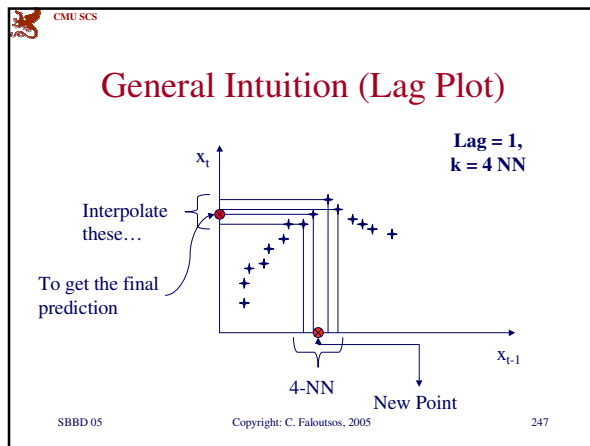
SBBD 05 Copyright: C. Faloutsos, 2005 245

CMU SCS

How to forecast?

- ARIMA - but: linearity assumption
- ANSWER: 'Delayed Coordinate Embedding' = Lag Plots [Sauer92]

SBBD 05 Copyright: C. Faloutsos, 2005 246



CMU SCS

Questions:

- Q1: How to choose lag L ?
- Q2: How to choose k (the # of NN)?
- Q3: How to interpolate?
- Q4: why should this work at all?

SBBD 05 Copyright: C. Faloutsos, 2005 248

CMU SCS

Q1: Choosing lag L

- Manually (16, in award winning system by [Sauer94])

SBBD 05 Copyright: C. Faloutsos, 2005 249

CMU SCS

Q2: Choosing number of neighbors k

- Manually (typically $\sim 1-10$)

SBBD 05 Copyright: C. Faloutsos, 2005 250

CMU SCS

Q3: How to interpolate?

How do we interpolate between the k nearest neighbors?

A3.1: Average

A3.2: Weighted average (weights drop with distance - how?)

SBBD 05 Copyright: C. Faloutsos, 2005 251

CMU SCS

Q3: How to interpolate?

A3.3: Using SVD - seems to perform best ([Sauer94] - first place in the Santa Fe forecasting competition)

x_t

x_{t-1}

SBBD 05 Copyright: C. Faloutsos, 2005 252

CMU SCS

Q4: Any theory behind it?

A4: YES!

SBBD 05 Copyright: C. Faloutsos, 2005 253

CMU SCS

Theoretical foundation

- Based on the “Takens’ Theorem” [Takens81]
- which says that **long enough** delay vectors **can do prediction**, even if there are unobserved variables in the dynamical system (= diff. equations)

SBBD 05 Copyright: C. Faloutsos, 2005 254

CMU SCS

Theoretical foundation

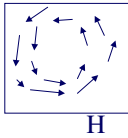
Example: Lotka-Volterra equations

$$\frac{dH}{dt} = rH - aH*P$$

$$\frac{dP}{dt} = bH*P - mP$$

H is count of prey (e.g., hare)
P is count of predators (e.g., lynx)

Suppose only P(t) is observed (t=1, 2, ...).

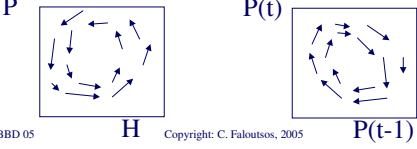


SBBD 05 Copyright: C. Faloutsos, 2005 255

CMU SCS

Theoretical foundation

- But the delay vector space is a faithful reconstruction of the internal system state
- So prediction in **delay vector space** is as good as prediction in **state space**



SBBD 05 Copyright: C. Faloutsos, 2005 256

CMU SCS

Detailed Outline

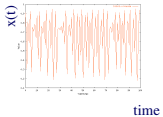
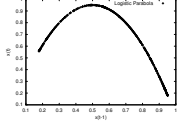
- Non-linear forecasting
 - Problem
 - Idea
 - How-to
 - Experiments
 - Conclusions

SBBD 05 Copyright: C. Faloutsos, 2005 257

CMU SCS

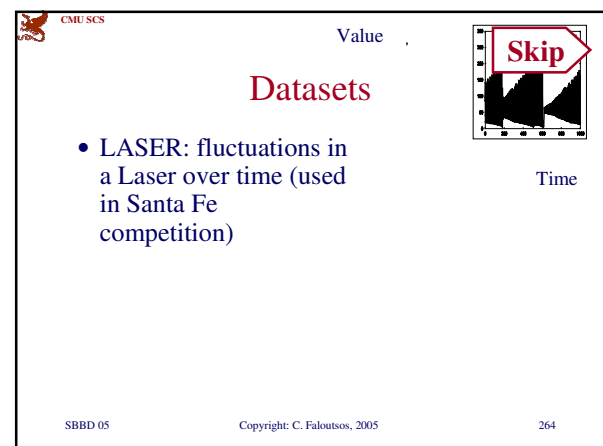
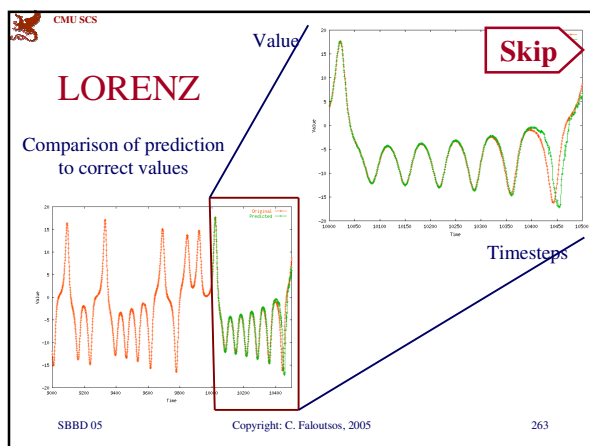
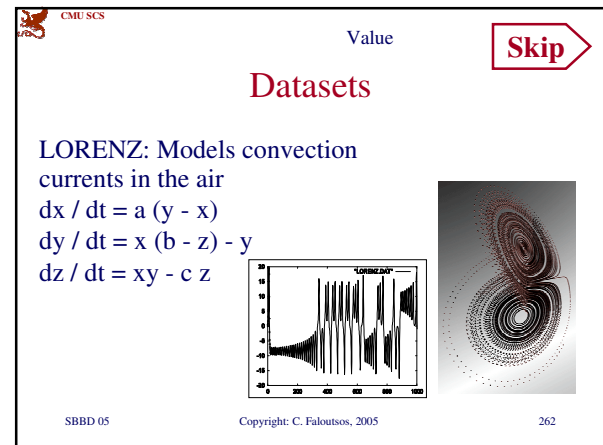
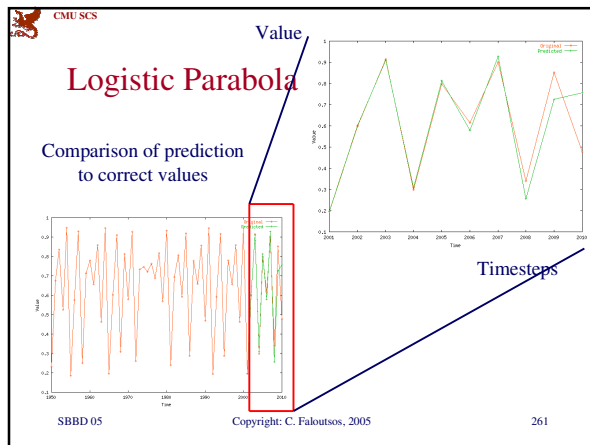
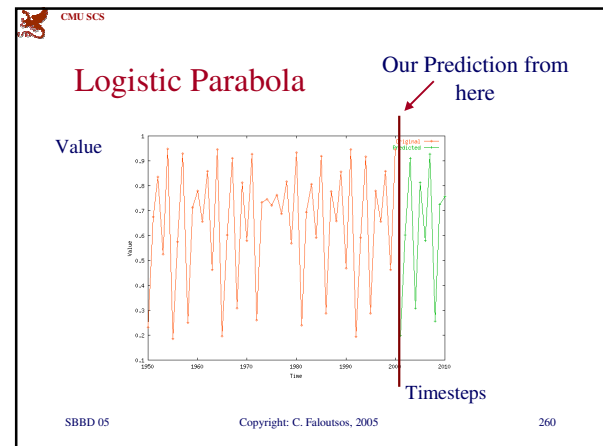
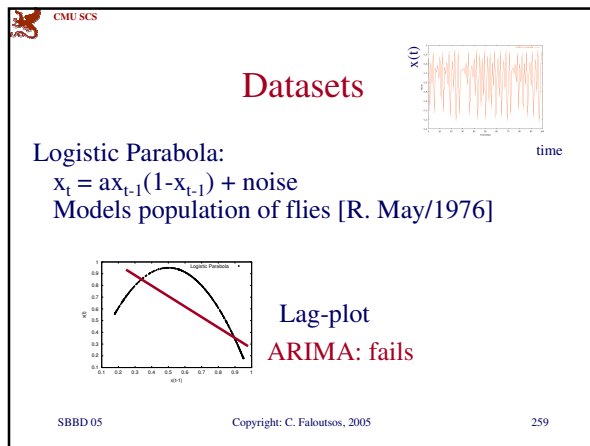
Datasets

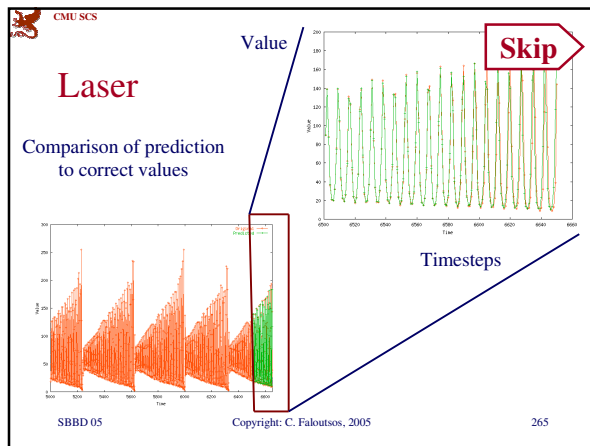
Logistic Parabola:
 $x_t = ax_{t-1}(1-x_{t-1}) + \text{noise}$
 Models population of flies [R. May/1976]

Lag-plot

SBBD 05 Copyright: C. Faloutsos, 2005 258





CMU SCS

Conclusions

- Lag plots for non-linear forecasting (Takens' theorem)
- suitable for 'chaotic' signals

SBBD 05 Copyright: C. Faloutsos, 2005 266

CMU SCS

References

- Deepay Chakrabarti and Christos Faloutsos *F4: Large-Scale Automated Forecasting using Fractals* CIKM 2002, Washington DC, Nov. 2002.
- Sauer, T. (1994). *Time series prediction using delay coordinate embedding*. (in book by Weigend and Gershenfeld, below) Addison-Wesley.
- Takens, F. (1981). *Detecting strange attractors in fluid turbulence*. Dynamical Systems and Turbulence. Berlin: Springer-Verlag.

SBBD 05 Copyright: C. Faloutsos, 2005 267

CMU SCS

References

- Weigend, A. S. and N. A. Gershenfeld (1994). *Time Series Prediction: Forecasting the Future and Understanding the Past*, Addison Wesley. (Excellent collection of papers on chaotic/non-linear forecasting, describing the algorithms behind the winners of the Santa Fe competition.)

SBBD 05 Copyright: C. Faloutsos, 2005 268

CMU SCS

Overall conclusions

- Similarity search: **Euclidean**/time-warping; **feature extraction** and **SAMs**

SBBD 05 Copyright: C. Faloutsos, 2005 269

CMU SCS

Overall conclusions

- Similarity search: **Euclidean**/time-warping; **feature extraction** and **SAMs**
- Signal processing: **DWT** is a powerful tool

SBBD 05 Copyright: C. Faloutsos, 2005 270



Overall conclusions

- Similarity search: **Euclidean**/time-warping; **feature extraction** and **SAMs**
- Signal processing: **DWT** is a powerful tool
- Linear Forecasting: **AR** (Box-Jenkins) methodology

SBBD 05 Copyright: C. Faloutsos, 2005 271



Overall conclusions

- Similarity search: **Euclidean**/time-warping; **feature extraction** and **SAMs**
- Signal processing: **DWT** is a powerful tool
- Linear Forecasting: **AR** (Box-Jenkins) methodology; **AWSOM**
- Bursty traffic: **multifractals** (80-20 'law')

SBBD 05 Copyright: C. Faloutsos, 2005 272



Overall conclusions

- Similarity search: **Euclidean**/time-warping; **feature extraction** and **SAMs**
- Signal processing: **DWT** is a powerful tool
- Linear Forecasting: **AR** (Box-Jenkins) methodology
- Bursty traffic: **multifractals** (80-20 'law')
- Non-linear forecasting: **lag-plots** (Takens)

SBBD 05 Copyright: C. Faloutsos, 2005 273



'Take home' messages

- Hard, but desirable query for sensor data: *'find patterns / outliers'*
- We need **fast**, **automated** such tools
 - Many great tools exist (DWT, ARIMA, ...)
 - some are readily usable; others need to be made scalable / single pass/ automatic

SBBD 05 Copyright: C. Faloutsos, 2005 274



THANK YOU!



christos@cs.cmu.edu
www.cs.cmu.edu/~christos

SBBD 05 Copyright: C. Faloutsos, 2005 275