SCS CMU — Civil and Environmental Engineering

Sensor Mining at work: Principles and a Water Quality Case-Study

Jeanne M. VanBriesen (CIT, CMU)
Christos Faloutsos (SCS, CMU)

SCS CMU — Civil and Environmental Engineering

Thanks


Dr. Jia-Yu (Tim) Pan (CMU)

Dr. Spiros Papadimitriou (IBM)



Dr. Yasushi Sakurai (NTT)

Prof. Byoung-Kee Yi (Pohang U.)


SCS CMU — Civil and Environmental Engineering

Acknowledgements

National Science Foundation
WHERE DISCOVERIES BEGIN

Dept. of Homeland Security

SCS CMU — Civil and Environmental Engineering

Outline

- Motivation
- Traditional tools (wavelets, etc)
- Recent streaming tools
- Intro to water quality [Jeanne]
- Conclusions

SCS CMU — Civil and Environmental Engineering

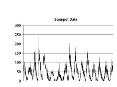
Problem definition

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

SCS CMU — Civil and Environmental Engineering

Motivation - Applications

- Weather, environment/anti-pollution
  - **water quality monitoring**
  - air quality monitoring
  - volcano monitoring



SCS CMU

Civil and Environmental ENGINEERING

Motivation – Applications (cont'd)

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

SCS CMU

Civil and Environmental ENGINEERING

Motivation - Applications (cont'd)

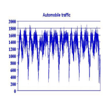
- 'Smart house'
  - sensors monitor temperature, humidity, air quality
- video surveillance

SCS CMU

Civil and Environmental ENGINEERING

Motivation - Applications (cont'd)

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

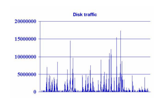


SCS CMU

Civil and Environmental ENGINEERING

Motivation - Applications (cont'd)

- Computer systems
  - data centers (``self-*'')
  - web servers
  - network traffic monitoring
  - ...



SCS CMU

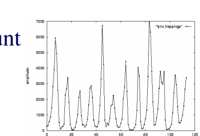
Civil and Environmental ENGINEERING

Problem #1:

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

Find: patterns, periodicities, and/or compress

count



year

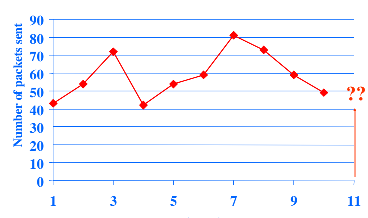
lynx caught per year
(chlorine per minute;
temperature per day)

SCS CMU

Civil and Environmental ENGINEERING

Problem#2: Forecast

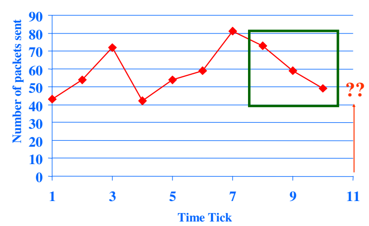
Given x_t, x_{t-1}, …, forecast x_{t+1}



SCS CMU — Civil and Environmental Engineering

Problem#2': Similarity search

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

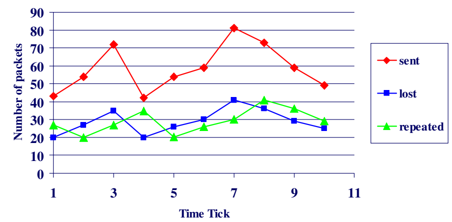


KDD 2006 J. VanBriesen, C. Faloutsos 13

SCS CMU — Civil and Environmental Engineering

Problem #3:

- Given: A set of **correlated** time sequences
- Forecast 'Sent(t)'



KDD 2006 J. VanBriesen, C. Faloutsos 14

SCS CMU — Civil and Environmental Engineering

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)

KDD 2006 J. VanBriesen, C. Faloutsos 16

SCS CMU — Civil and Environmental Engineering

Important topics NOT in this tutorial:

- Continuous queries
  - [Babu+Widom] [Gehrke+] [Madden+]
- Categorical data streams
  - [Hatonen+96]
- Outlier detection (discontinuities)
  - [Breunig+00]

KDD 2006 J. VanBriesen, C. Faloutsos 17

SCS CMU — Civil and Environmental Engineering

Outline

- Motivation
- Traditional tools
  - Similarity Search and Indexing
  - DSP (Digital Signal Processing)
  - Linear Forecasting
  - ICA
- Recent streaming tools
- Intro to water quality [Jeanne]
- Conclusions

KDD 2006 J. VanBriesen, C. Faloutsos 18

SCS CMU — Civil and Environmental Engineering

Importance of distance functions

Subtle, but **absolutely necessary**:

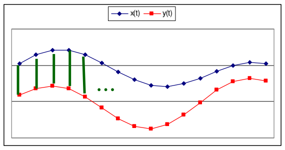
- A 'must' for similarity indexing (-> forecasting)
- A 'must' for clustering

Two major families

- Euclidean and Lp norms
- Time warping and variations

KDD 2006 J. VanBriesen, C. Faloutsos 19

Euclidean and Lp



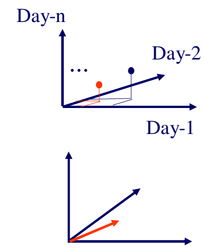
$$D(\vec{x},\vec{y}) = \sum_{i=1}^{n} (x_i - y_i)^2$$

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

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

Observation

Euclidean distance is closely related to
- cosine similarity
- dot product
- 'cross-correlation' function

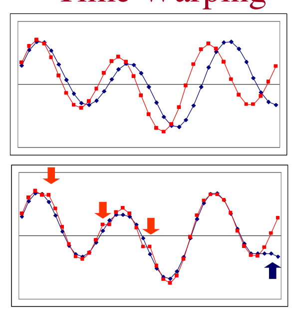


Time Warping

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

Time Warping

'stutters':



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]

Conclusions

Prevailing distances:
- Euclidean and
- time-warping

SCS CMU

Conclusions

Prevailing distances:

- **Euclidean** and ⟵ probably most suitable for our setting
- time-warping

SCS CMU

Outline

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

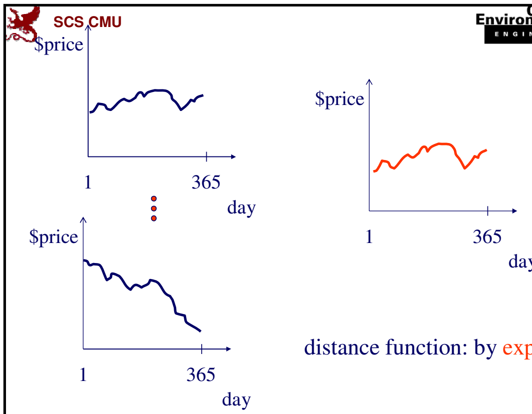
SCS CMU

Indexing

Problem:

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

SCS CMU



distance function: by expert

SCS CMU

Idea: 'GEMINI'

Eg., '*find stocks similar to MSFT*'

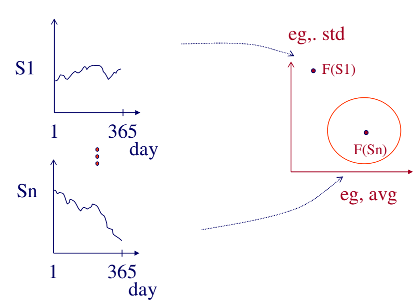
Seq. scanning: too slow

How to accelerate the search?

[Faloutsos96]

SCS CMU

'GEMINI' - Pictorially



SCS CMU

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

KDD 2006 J. VanBriesen, C. Faloutsos 32

SCS CMU

Examples of GEMINI

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

KDD 2006 J. VanBriesen, C. Faloutsos 33

SCS CMU

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]

KDD 2006 J. VanBriesen, C. Faloutsos 34

SCS CMU

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:

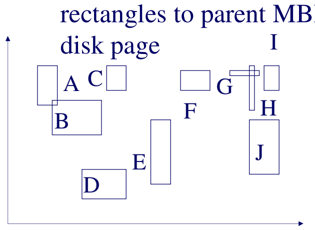
KDD 2006 J. VanBriesen, C. Faloutsos 35

SCS CMU

Skip

R-trees

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



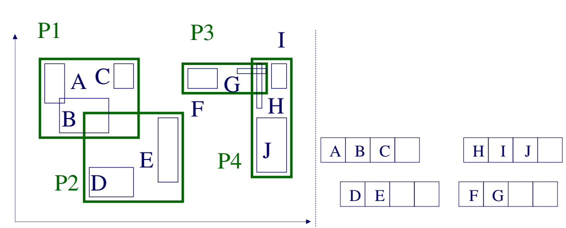
KDD 2006 J. VanBriesen, C. Faloutsos 36

SCS CMU

Skip

R-trees

- eg., w/ fanout 4:



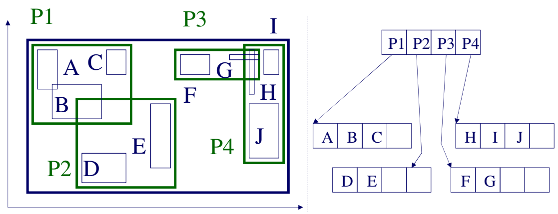
KDD 2006 J. VanBriesen, C. Faloutsos 37

SCS CMU

Skip

R-trees

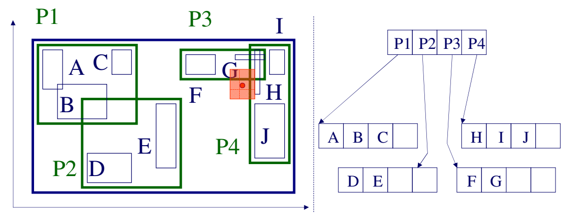
- eg., w/ fanout 4:



SCS CMU

Skip

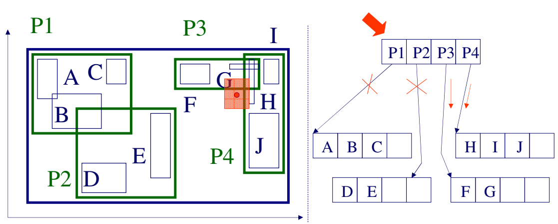
R-trees - range search?



SCS CMU

Skip

R-trees - range search?



SCS CMU

Conclusions

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

SCS CMU

Outline

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

SCS CMU

Outline

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

SCS CMU

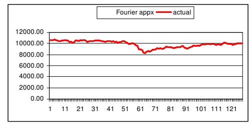
DFT and cousins

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

KDD 2006 J. VanBriesen, C. Faloutsos 44

SCS CMU

DFT and stocks

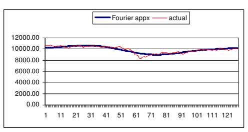


- Dow Jones Industrial index, 6/18/2001-12/21/2001

KDD 2006 J. VanBriesen, C. Faloutsos 45

SCS CMU

DFT and stocks



- Dow Jones Industrial index, 6/18/2001-12/21/2001
- just 3 DFT coefficients give very good approximation

KDD 2006 J. VanBriesen, C. Faloutsos 46

SCS CMU

Outline

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

KDD 2006 J. VanBriesen, C. Faloutsos 47

SCS CMU

SVD

- <u>THE</u> optimal method for dimensionality reduction
  - (under the Euclidean metric)

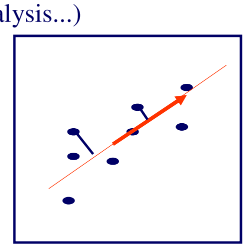
KDD 2006 J. VanBriesen, C. Faloutsos 48

SCS CMU

Singular Value Decomposition (SVD)

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

day2



day1

LSI: S. Dumais; M. Berry

KL: eg, Duda+Hart

PCA: eg., Jolliffe

Details: [Press+], [Faloutsos96]

KDD 2006 J. VanBriesen, C. Faloutsos 49

SCS CMU

SVD

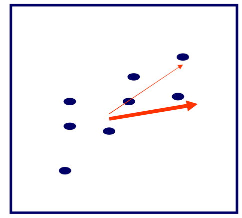
- **Extremely** useful tool
  - (also behind PageRank/google and Kleinberg's algorithm for hubs and authorities)
- But may be slow: O(N * M * M) if N>M
- any approximate, faster method?

KDD 2006 J. VanBriesen, C. Faloutsos 50

SCS CMU

SVD shorcuts

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



KDD 2006 J. VanBriesen, C. Faloutsos 51

SCS CMU

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])

KDD 2006 J. VanBriesen, C. Faloutsos 52

SCS CMU

Variations of R.P.:

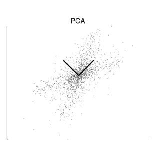
- 'Sketches' in time series analysis [Indyk+, VLDB 2000]

KDD 2006 J. VanBriesen, C. Faloutsos 53

SCS CMU

SVD - quality

- Q: can we find better 'hidden variables'?
- A: yes – with Independent Component Analysis (ICA) – see later

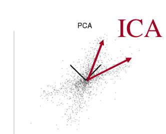


KDD 2006 J. VanBriesen, C. Faloutsos 54

SCS CMU

SVD - quality

- Q: can we find better 'hidden variables'?
- A: yes – with Independent Component Analysis (ICA) – see later



KDD 2006 J. VanBriesen, C. Faloutsos 55

SCS CMU

Conclusions - Practitioner's guide

Similarity search in time sequences

1) establish/choose distance (Euclidean, time-warping,…)

2) extract features (SVD, DWT), and use an SAM (R-tree/variant) or a Metric Tree (M-tree)

2') for high intrinsic dimensionalities, consider sequential scan (it might win…)

KDD 2006 J. VanBriesen, C. Faloutsos 56

SCS CMU

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)

KDD 2006 J. VanBriesen, C. Faloutsos 57

SCS CMU

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/

KDD 2006 J. VanBriesen, C. Faloutsos 58

SCS CMU

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.

KDD 2006 J. VanBriesen, C. Faloutsos 59

SCS CMU

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.

KDD 2006 J. VanBriesen, C. Faloutsos 60

SCS CMU

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.

KDD 2006 J. VanBriesen, C. Faloutsos 61

SCS CMU

References

- P. Indyk, N. Koudas, and S. Muthukrishnan. Identifying Representative Trends in Massive Time Series Data Sets Using Sketches. In Proc. of the 26th Int. Conf. on Very Large Data Bases, Cairo, Egypt, September 2000.
- Jolliffe, I. T. (1986). Principal Component Analysis, Springer Verlag.

KDD 2006 J. VanBriesen, C. Faloutsos 62

SCS CMU

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.

KDD 2006 J. VanBriesen, C. Faloutsos 63

SCS CMU

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.

KDD 2006 J. VanBriesen, C. Faloutsos 64

SCS CMU

References

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

KDD 2006 J. VanBriesen, C. Faloutsos 65

SCS CMU

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.

KDD 2006 J. VanBriesen, C. Faloutsos 66

SCS CMU

Traditional tools: DSP (Digital Signal Processing)

KDD 2006 J. VanBriesen, C. Faloutsos 67

SCS CMU — Civil and Environmental ENGINEERING

Outline

- Motivation
- Traditional tools
  - Similarity Search and Indexing
  - DSP (Digital Signal Processing)
  - Linear Forecasting
  - ICA
- Recent streaming tools
- Intro to water quality [Jeanne]
- Conclusions

KDD 2006 J. VanBriesen, C. Faloutsos 68

SCS CMU — Civil and Environmental ENGINEERING

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

KDD 2006 J. VanBriesen, C. Faloutsos 69

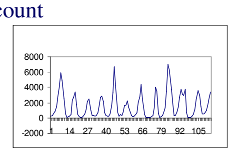
SCS CMU — Civil and Environmental ENGINEERING

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

KDD 2006 J. VanBriesen, C. Faloutsos 70

SCS CMU — Civil and Environmental ENGINEERING

What does DFT do?

A: highlights the periodicities

Powerful tool: Amplitude spectrum

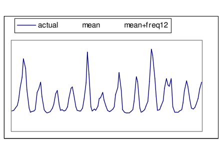
KDD 2006 J. VanBriesen, C. Faloutsos 71

SCS CMU — Civil and Environmental ENGINEERING

DFT: Amplitude spectrum

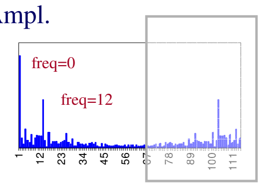
Amplitude: $A_f^{\ 2} = \mathrm{Re}^2(X_f) + \mathrm{Im}^2(X_f)$

count



year

Ampl.



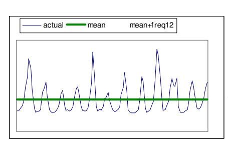
Freq.

KDD 2006 J. VanBriesen, C. Faloutsos 72

SCS CMU — Civil and Environmental ENGINEERING

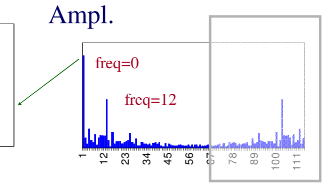
DFT: Amplitude spectrum

count



year

Ampl.

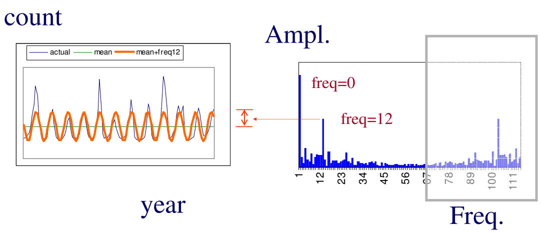


Freq.

KDD 2006 J. VanBriesen, C. Faloutsos 73

SCS CMU — Civil and Environmental Engineering

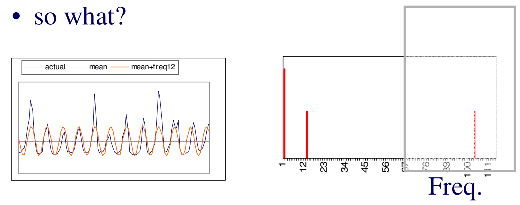
DFT: Amplitude spectrum



SCS CMU — Civil and Environmental Engineering

DFT: Amplitude spectrum

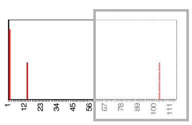
- excellent approximation, with only 2 frequencies!
- so what?



SCS CMU — Civil and Environmental Engineering

DFT: Amplitude spectrum

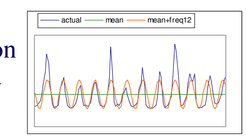
- excellent approximation, with only 2 frequencies!
- so what?
- A1: **(lossy) compression**
- A2: pattern discovery



SCS CMU — Civil and Environmental Engineering

DFT: Amplitude spectrum

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



SCS CMU — Civil and Environmental Engineering

Outline

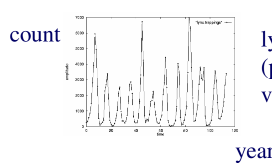
- Motivation
- Similarity Search and Indexing
- DSP
  - DFT
  - DWT
    - Definition of DWT and properties
    - how to read the DWT scalogram

SCS CMU — Civil and Environmental Engineering

Problem #1:

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

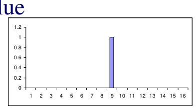
Find: patterns, periodicities, and/or **compress**



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

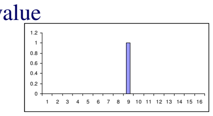
Wavelets - DWT

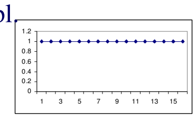
- DFT is great - but, how about compressing a spike?



Wavelets - DWT

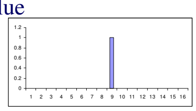
- DFT is great - but, how about compressing a spike?
- A: Terrible - all DFT coefficients needed!

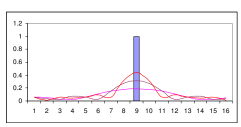




Wavelets - DWT

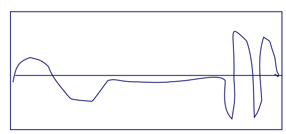
- DFT is great - but, how about compressing a spike?
- A: Terrible - all DFT coefficients needed!





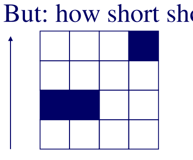
Wavelets - DWT

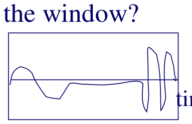
- Similarly, DFT suffers on short-duration waves (eg., baritone, silence, soprano)



Wavelets - DWT

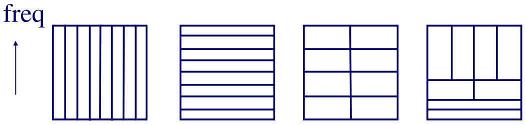
- Solution#1: Short window Fourier transform (SWFT)
- But: how short should be the window?





Wavelets - DWT

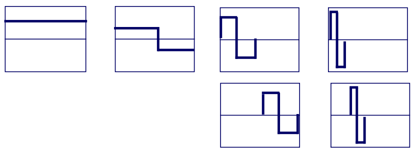
- Answer: **multiple** window sizes! -> DWT



SCS CMU

Haar Wavelets

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



SCS CMU

Skip

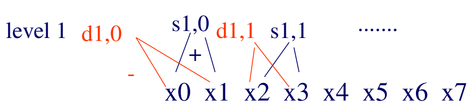
Wavelets - construction

x0 x1 x2 x3 x4 x5 x6 x7

SCS CMU

Skip

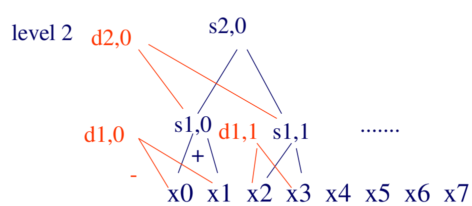
Wavelets - construction



SCS CMU

Skip

Wavelets - construction

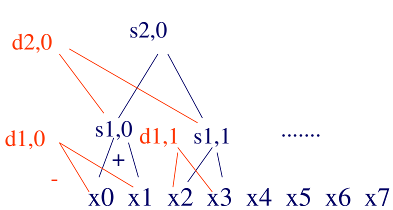


SCS CMU

Skip

Wavelets - construction

etc ...



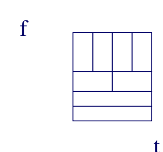
SCS CMU

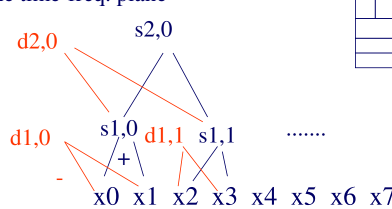
Skip

Wavelets - construction

Q: map each coefficient

on the time-freq. plane





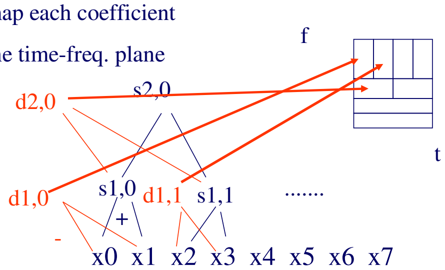
SCS CMU

Skip

Wavelets - construction

Q: map each coefficient

on the time-freq. plane



KDD 2006 J. VanBriesen, C. Faloutsos 92

SCS CMU

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 "\t", $diff[$i];
        $smooth [$i] = ($vals[2*$i] + $vals[2*$i + 1] )/ sqrt(2);
  }
  print "\n";
  @vals = @smooth;
  $half = int($half/2);
}
print "\t", $vals[0], "\n" ;     # the final, smooth component
```

KDD 2006 J. VanBriesen, C. Faloutsos 93

SCS CMU

Skip

Wavelets - construction

Observation1:

- ‘+’ can be some weighted addition
- ‘-’ is the corresponding weighted difference (‘Quadrature mirror filters’)

Observation2: unlike DFT/DCT,

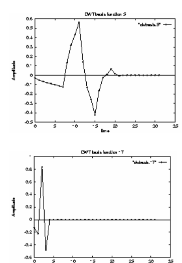
- there are *many* wavelet bases: Haar, Daubechies-4, Daubechies-6, Coifman, Morlet, Gabor, ...

KDD 2006 J. VanBriesen, C. Faloutsos 94

SCS CMU

Skip

Wavelets - how do they look like?



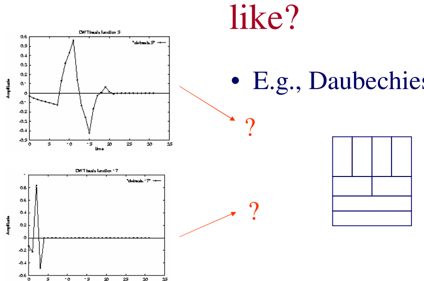
- E.g., Daubechies-4

KDD 2006 J. VanBriesen, C. Faloutsos 95

SCS CMU

Skip

Wavelets - how do they look like?



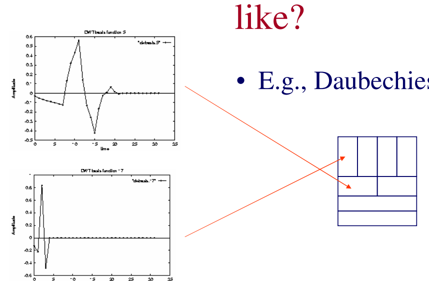
- E.g., Daubechies-4

KDD 2006 J. VanBriesen, C. Faloutsos 96

SCS CMU

Skip

Wavelets - how do they look like?



- E.g., Daubechies-4

KDD 2006 J. VanBriesen, C. Faloutsos 97

SCS CMU — Civil and Environmental Engineering

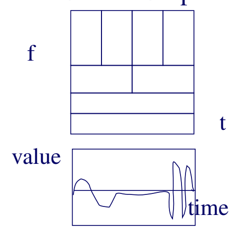
Outline

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

SCS CMU — Civil and Environmental Engineering

Wavelets - Drill#1:

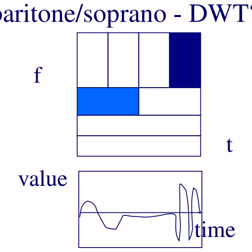
- Q: baritone/silence/soprano - DWT?



SCS CMU — Civil and Environmental Engineering

Wavelets - Drill#1:

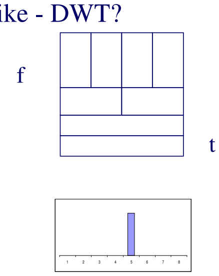
- Q: baritone/soprano - DWT?



SCS CMU — Civil and Environmental Engineering

Wavelets - Drill#2:

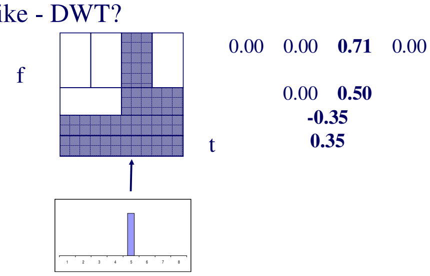
- Q: spike - DWT?



SCS CMU — Civil and Environmental Engineering

Wavelets - Drill#2:

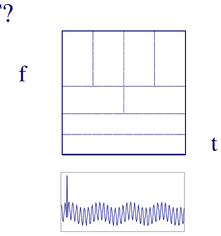
- Q: spike - DWT?



SCS CMU — Civil and Environmental Engineering

Wavelets - Drill#3:

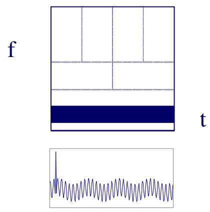
- Q: weekly + daily periodicity, + spike - DWT?



SCS CMU

Wavelets - Drill#3:

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

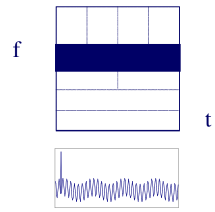


KDD 2006 J. VanBriesen, C. Faloutsos 104

SCS CMU

Wavelets - Drill#3:

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

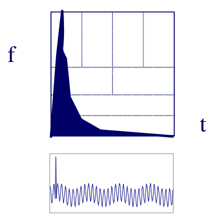


KDD 2006 J. VanBriesen, C. Faloutsos 105

SCS CMU

Wavelets - Drill#3:

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

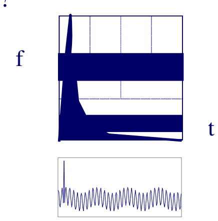


KDD 2006 J. VanBriesen, C. Faloutsos 106

SCS CMU

Wavelets - Drill#3:

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

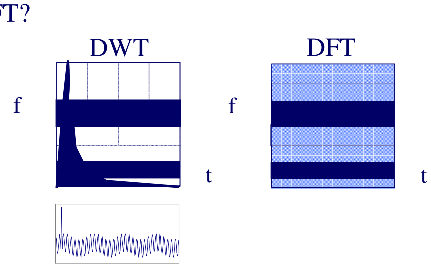


KDD 2006 J. VanBriesen, C. Faloutsos 107

SCS CMU

Wavelets - Drill#3:

- Q: DFT?



KDD 2006 J. VanBriesen, C. Faloutsos 108

SCS CMU

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

KDD 2006 J. VanBriesen, C. Faloutsos 109

SCS CMU

Overall Conclusions

- DFT (and 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!)

KDD 2006 J. VanBriesen, C. Faloutsos 110

SCS CMU

Overall Conclusions

- DWT: used for summarization of streams [Gilbert+01], db histograms etc

KDD 2006 J. VanBriesen, C. Faloutsos 111

SCS CMU

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)

KDD 2006 J. VanBriesen, C. Faloutsos 112

SCS CMU

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

KDD 2006 J. VanBriesen, C. Faloutsos 113

SCS CMU

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)

KDD 2006 J. VanBriesen, C. Faloutsos 114

SCS CMU

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

KDD 2006 J. VanBriesen, C. Faloutsos 115

SCS CMU

Traditional tools: Linear Forecasting

SCS CMU

Forecasting

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

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

SCS CMU

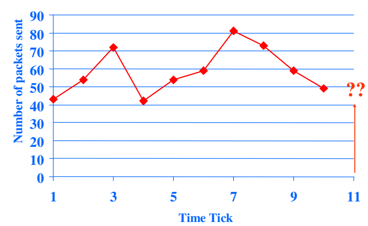
Outline

- Motivation
- Traditional tools
  - Similarity Search and Indexing
  - DSP (Digital Signal Processing)
  - Linear Forecasting
  - ICA
- Recent streaming tools
- Intro to water quality [Jeanne]
- Conclusions

SCS CMU

Problem#2: Forecast

- Example: give x_{t-1}, x_{t-2}, …, forecast x_t



SCS CMU

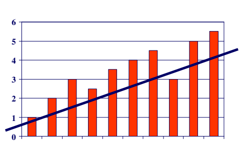
Forecasting: Preprocessing

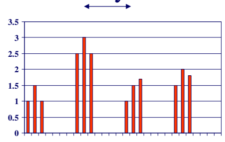
MANUALLY:

remove trends

spot periodicities

7 days





SCS CMU

Problem#2: Forecast

- Solution: try to express

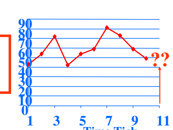
  x_t

  as a linear function of the past: x_{t-2}, x_{t-2}, …,

  (up to a window of w)

Formally:

$$\boxed{x_t \approx a_1 x_{t-1} + \ldots + a_w x_{t-w} + noise}$$



SCS CMU

Skip

(Problem: Back-cast; interpolate)

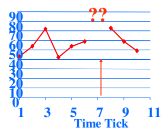
- Solution - interpolate: try to express

  x_t

  as a linear function of the past AND the future:

  $x_{t+1}, x_{t+2}, \ldots x_{t+wfuture}; x_{t-1}, \ldots x_{t-wpast}$

  (up to windows of w_{past} , w_{future})
- EXACTLY the same algo's



KDD 2006 J. VanBriesen, C. Faloutsos 122

SCS CMU

Linear Auto Regression:

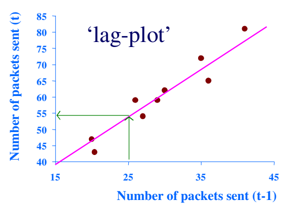
Time	Packets Sent(t)
1	43
2	54
3	72
…	…
N	??

KDD 2006 J. VanBriesen, C. Faloutsos 123

SCS CMU

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])

KDD 2006 J. VanBriesen, C. Faloutsos 124

SCS CMU

Outline

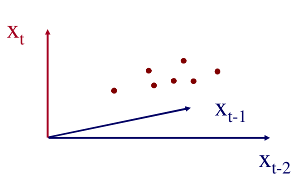
- Motivation
- ...
- Linear Forecasting
  - Auto-regression: **Least Squares; RLS**
  - Co-evolving time sequences
  - Examples
  - Conclusions

KDD 2006 J. VanBriesen, C. Faloutsos 125

SCS CMU

More details:

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

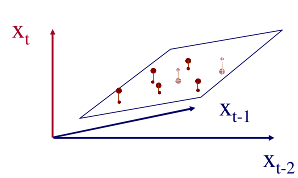


KDD 2006 J. VanBriesen, C. Faloutsos 126

SCS CMU

More details:

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

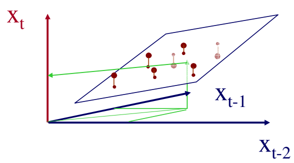


KDD 2006 J. VanBriesen, C. Faloutsos 127

SCS CMU — Civil and Environmental Engineering

More details:

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



SCS CMU — Civil and Environmental Engineering

Skip

More details:

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

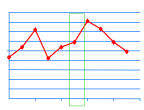
$$\mathbf{X}_{[N \times w]} \times \mathbf{a}_{[w \times 1]} = \mathbf{y}_{[N \times 1]}$$

- OVER-CONSTRAINED
  - **a** is the vector of the regression coefficients
  - **X** has the *N* values of the *w* indep. variables
  - **y** has the N values of the dependent variable

SCS CMU — Civil and Environmental Engineering

Skip

More details:



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

Ind-var1 Ind-var-w

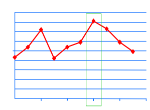
time

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

SCS CMU — Civil and Environmental Engineering

Skip

More details:



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

Ind-var1 Ind-var-w

time

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

SCS CMU — Civil and Environmental Engineering

Skip

More details

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

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

- (Moore-Penrose pseudo-inverse)
- **a** is the vector that minimizes the RMSE from **y**

SCS CMU — Civil and Environmental Engineering

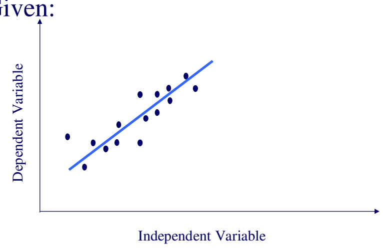
Even more details

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

SCS CMU · Civil and Environmental ENGINEERING

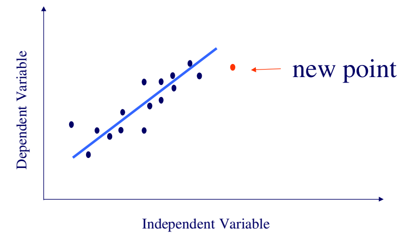
Even more details

- Given:



SCS CMU · Civil and Environmental ENGINEERING

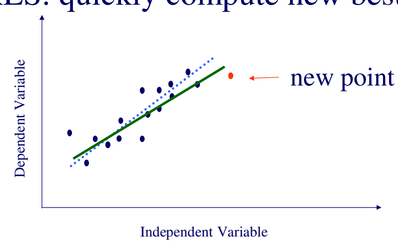
Even more details



SCS CMU · Civil and Environmental ENGINEERING

Even more details

RLS: quickly compute new best fit



SCS CMU · Civil and Environmental ENGINEERING

Even more details

- 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 $O(1 \times w^2)$

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

SCS CMU · Civil and Environmental ENGINEERING

Skip

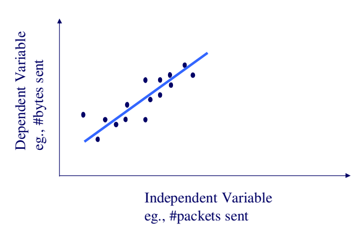
Even more details

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

SCS CMU · Civil and Environmental ENGINEERING

Skip

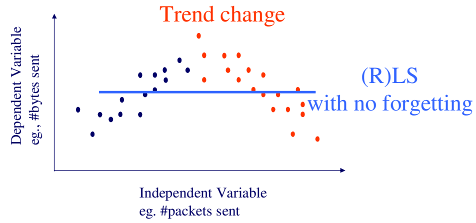
Adaptability - 'forgetting'



SCS CMU

Skip

Adaptability - 'forgetting'

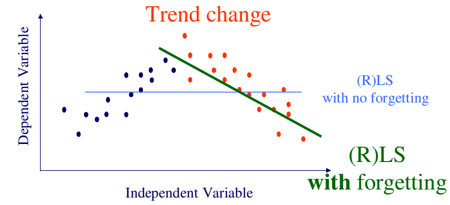


KDD 2006 J. VanBriesen, C. Faloutsos 140

SCS CMU

Skip

Adaptability - 'forgetting'



- RLS: can *trivially* handle 'forgetting'

KDD 2006 J. VanBriesen, C. Faloutsos 141

SCS CMU

Outline

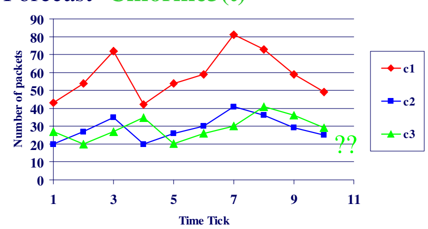
- Motivation
- ...
- Linear Forecasting
  - Auto-regression: Least Squares; RLS
  - Co-evolving time sequences
  - Conclusions

KDD 2006 J. VanBriesen, C. Faloutsos 142

SCS CMU

Co-Evolving Time Sequences

- Given: A set of **correlated** time sequences
- Forecast '**Chlorine3(t)**'



KDD 2006 J. VanBriesen, C. Faloutsos 143

SCS CMU

Solution:

Q: what should we do?

KDD 2006 J. VanBriesen, C. Faloutsos 144

SCS CMU

Solution:

Least Squares, with

- Dep. Variable: **c3(t)**
- Indep. Variables:
  - **c3(t-1)** … **c3(t-w)**;
  - **c2(t-1)** …**c2(t-w)**;
  - **c1(t-1)** ,...
- (named: 'MUSCLES' [Yi+00])

KDD 2006 J. VanBriesen, C. Faloutsos 145

SCS CMU

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]

KDD 2006 J. VanBriesen, C. Faloutsos 146

SCS CMU

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/`

KDD 2006 J. VanBriesen, C. Faloutsos 147

SCS CMU

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.

KDD 2006 J. VanBriesen, C. Faloutsos 148

SCS CMU

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)

KDD 2006 J. VanBriesen, C. Faloutsos 149