


**15-826: Multimedia Databases
and Data Mining**


Lecture #10: Fractals - case studies - I
C. Faloutsos



Must-read Material


- Christos Faloutsos and Ibrahim Kamel,
*Beyond Uniformity and Independence:
Analysis of R-trees Using the Concept of
Fractal Dimension*, Proc. ACM SIGACT-
SIGMOD-SIGART PODS, May 1994, pp.
4-13, Minneapolis, MN.

15-826 Copyright: C. Faloutsos (2011) 2




Optional Material

Optional, but **very** useful: Manfred Schroeder
*Fractals, Chaos, Power Laws: Minutes
from an Infinite Paradise* W.H. Freeman
and Company, 1991 (on reserve in the WeH
library)



15-826 Copyright: C. Faloutsos (2011) 3



CMU SCS


Reminder

- Code at www.cs.cmu.edu/~christos/SRC/fdnq_h.zip

Also, in 'R'

```
> library(fdim);
```

15-826 Copyright: C. Faloutsos (2011) 4




CMU SCS

Outline

Goal: 'Find **similar** / **interesting** things'

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

15-826 Copyright: C. Faloutsos (2011) 5




CMU SCS

Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
 - z-ordering
 - R-trees
 - misc
- ➔ • fractals
 - intro
 - applications
- text

15-826 Copyright: C. Faloutsos (2011) 6




CMU SCS

Indexing - Detailed outline

- fractals
 - intro
 - applications
 - disk accesses for R-trees (range queries)
 - dimensionality reduction
 - selectivity in M-trees
 - dim. curse revisited
 - “fat fractals”
 - quad-tree analysis [Gaede+]

15-826 Copyright: C. Faloutsos (2011) 7




CMU SCS

(Fractals mentioned before:)

- for performance analysis of R-trees
- fractals for dim. reduction

15-826 Copyright: C. Faloutsos (2011) 8



CMU SCS

Case study#1: R-tree performance

Problem

- Given
 - N points in E -dim space
- Estimate # disk accesses for a range query
($q_1 \times \dots \times q_E$)

(assume: ‘good’ R-tree, with tight, cube-like MBRs)

15-826 Copyright: C. Faloutsos (2011) 9

CMU SCS

Case study#1: R-tree performance

Problem

- Given
 - N points in E-dim space
 - with fractal dimension D
- Estimate # disk accesses for a range query ($q_1 \times \dots \times q_E$)

(assume: 'good' R-tree, with tight, cube-like MBRs)
Typically, in DB Q-opt: uniformity + independence

15-826 Copyright: C. Faloutsos (2011) 10

CMU SCS

Examples: World's countries

- BUT: area vs population for ~200 countries (1991 CIA fact-book).

pop

log(pop)

area log(area)

15-826 Copyright: C. Faloutsos (2011) 11

CMU SCS

Examples: World's countries

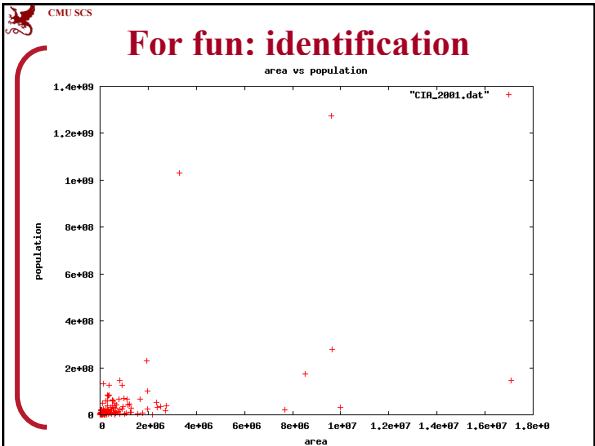
- neither uniform, nor independent!

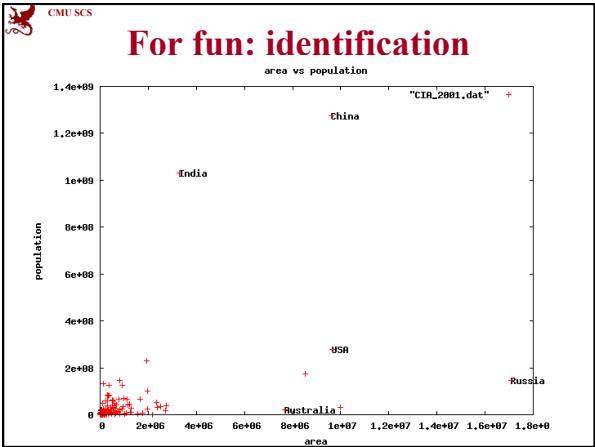
pop

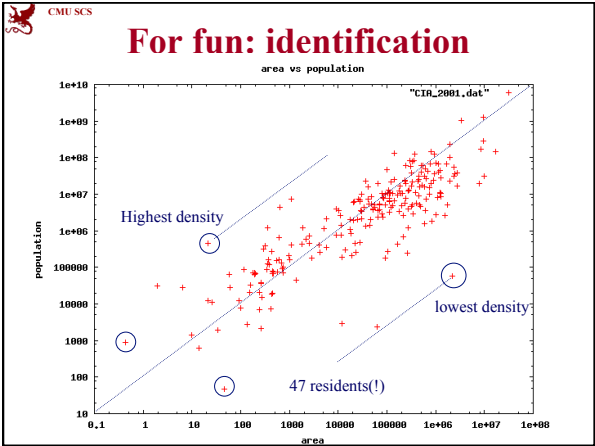
log(pop)

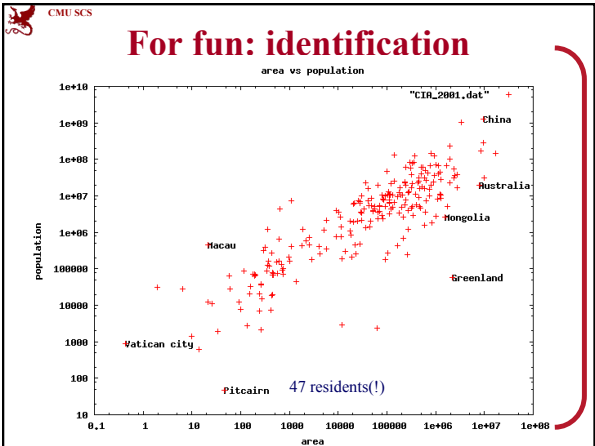
area log(area)

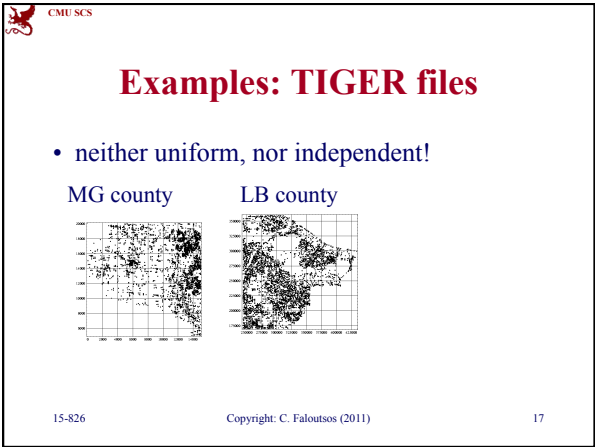
15-826 Copyright: C. Faloutsos (2011) 12

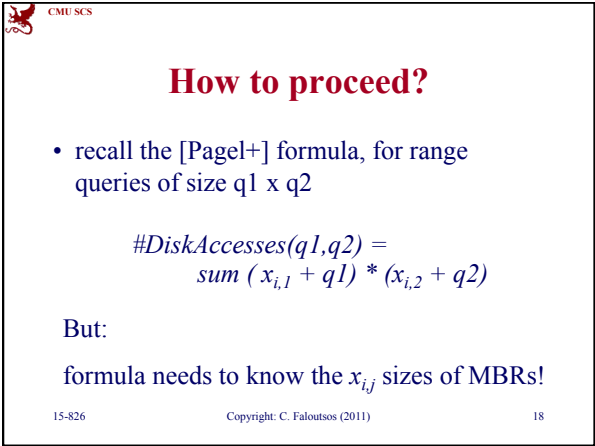












CMU SCS

How to proceed?

But:
formula needs to know the x_{ij} sizes of MBRs!

Answer (jumping ahead):

$$s = (C/N)^{1/D_0}$$

15-826 Copyright: C. Faloutsos (2011) 19

CMU SCS

How to proceed?

But:
formula needs to know the x_{ij} sizes of MBRs!

Answer (jumping ahead):

$$s = (C/N)^{1/D_0}$$

side of (parent) MBR → s ← Hausdorff fd
 page capacity → C ← # of data points


15-826 Copyright: C. Faloutsos (2011) 20


CMU SCS

Let's see the rationale

$$s = (C/N)^{1/D_0}$$

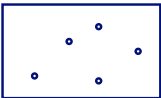

15-826 Copyright: C. Faloutsos (2011) 21

CMU SCS



R-trees - performance analysis


I.e: for range queries - how many disk accesses,
if we just now that we have
- N points in E -d space?
A: can not tell! need to know distribution




15-826

Copyright: C. Faloutsos (2011)

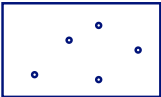

22

CMU SCS



R-trees - performance analysis


Q: OK - so we are told that the **Hausdorff** fractal
dim. = D_0 - Next step?
(also know that there are at most C points per
page)
 $D_0=1$ $D_0=2$




15-826

Copyright: C. Faloutsos (2011)

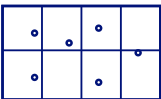
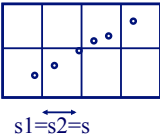
23

CMU SCS



R-trees - performance analysis

Assumption1: square-like parents ($s*s$)
Assumption2: fully packed (C points each)
Assumption3: non-overlapping
 $D_0=1$ $D_0=2$



15-826

Copyright: C. Faloutsos (2011)

24

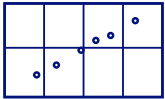
8

CMU SCS

R-trees - performance analysis

Assumption1: square-like parents ($s \times s$)
 Assumption2: fully packed (N/C non-empty)
 Assumption3: non-overlapping

$D_0=1$




$s_1=s_2=s$

15-826 Copyright: C. Faloutsos (2011) 25

CMU SCS

R-trees - performance analysis

Hint: defn of Hausdorff f.d.:



Felix Hausdorff (1868-1942)

15-826 Copyright: C. Faloutsos (2011) 26

CMU SCS


Reminder:


Hausdorff or box-counting fd:

- Box counting plot: $\log(N(r))$ vs $\log(r)$
- r : grid side
- $N(r)$: count of non-empty cells
- (Hausdorff) fractal dimension D_0 :

$$D_0 = -\frac{\partial \log(N(r))}{\partial \log(r)}$$

15-826 Copyright: C. Faloutsos (2011) 27

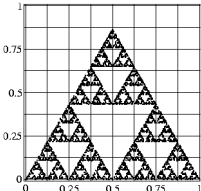
CMU SCS



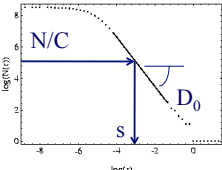
Reminder

- Hausdorff fd:

$r \quad \log(\text{\#non-empty cells})$




SLOPE = -1.5743




15-826

Copyright: C. Faloutsos (2011)

28

CMU SCS



Reminder

- dfn of Hausdorff fd implies that


$N(r) \sim r^{(-D_0)}$


non-empty cells of side r

15-826

Copyright: C. Faloutsos (2011)

29

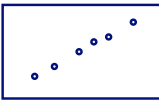
CMU SCS



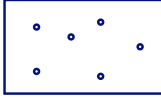
R-trees - performance analysis

Q (rephrased): what is the side s_1, s_2, \dots of parent nodes, given N data points, packed by C , with f.d. = D_0

$D_0=1$



$D_0=2$



15-826

Copyright: C. Faloutsos (2011)

30

CMU SCS

R-trees - performance analysis

Q (rephrased): what is the side s_1, s_2, \dots of parent nodes, given N data points, packed by C , with f.d. = D_0

D0=1 **D0=2**

15-826 Copyright: C. Faloutsos (2011) 31

CMU SCS

R-trees - performance analysis

Q (rephrased): what is the side s_1, s_2, \dots of parent nodes, given N data points, packed by C , with f.d. = D_0

D0=1 **D0=2**

15-826 Copyright: C. Faloutsos (2011) 32

CMU SCS

R-trees - performance analysis



A: (educated guess)

- $s=s_1=s_2 (= \dots)$ - square-like MBRs
- N/C non-empty cells = $K * s^{(-D_0)}$

$\log(\#cells)$

D0=1 **D0=2**

15-826 Copyright: C. Faloutsos (2011) 33

CMU SCS  


R-trees - performance analysis

Details of derivations: in [PODS 94].
 Finally, expected side s of parent MBRs:


$$s = (C/N)^{1/D0}$$

Q: sanity check: how does s change with $D0$?
 A:

15-826 Copyright: C. Faloutsos (2011) 34

CMU SCS 

R-trees - performance analysis


Details of derivations: in [Kamel+, PODS 94]. 

Finally, expected side s of parent MBRs:

$$s = (C/N)^{1/D0}$$

Q: sanity check: how does s change with $D0$?
 A: s grows with $D0$
 Q: does it make sense?
 Q: does it suffer from (intrinsic) dim. curse?


15-826 Copyright: C. Faloutsos (2011) 35

CMU SCS 

R-trees - performance analysis

Q: Final-final formula (# disk accesses for range queries $q1 \times q2 \times \dots$):
 A:

15-826 Copyright: C. Faloutsos (2011) 36



R-trees - performance analysis


Q: Final-final formula (# disk accesses for range queries $q1 \times q2 \times \dots$):

A: # of parent-node accesses:

$$N/C * (s + q1) * (s + q2) * \dots (s + q_E)$$

A: # of grand-parent node accesses

15-826 Copyright: C. Faloutsos (2011) 37



R-trees - performance analysis

Q: Final-final formula (# disk accesses for range queries $q1 \times q2 \times \dots$):

A: # of parent-node accesses:


$$N/C * (s + q1) * (s + q2) * \dots (s + q_E)$$

A: # of grand-parent node accesses

$$N/(C^2) * (s' + q1) * (s' + q2) * \dots (s' + q_E)$$

$$s' = ??$$

15-826 Copyright: C. Faloutsos (2011) 38



R-trees - performance analysis

Q: Final-final formula (# disk accesses for range queries $q1 \times q2 \times \dots$):

A: # of parent-node accesses:


$$N/C * (s + q1) * (s + q2) * \dots (s + q_E)$$

A: # of grand-parent node accesses

$$N/(C^2) * (s' + q1) * (s' + q2) * \dots (s' + q_E)$$

$$s' = (C^2/N)^{1/D0}$$

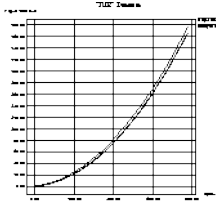
15-826 Copyright: C. Faloutsos (2011) 39

CMU SCS

R-trees - performance analysis

Results: IUE (x-y star coordinates)

leaf accesses




(a) IUE - Leaf accesses vs. query side

query side

15-826

Copyright: C. Faloutsos (2011)


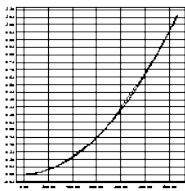
40

CMU SCS

R-trees - performance analysis

Results: LB County

leaf accesses




query side

15-826

Copyright: C. Faloutsos (2011)


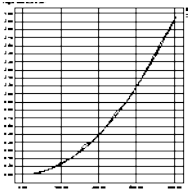
41

CMU SCS

R-trees - performance analysis

Results: MG-county

leaf accesses



query side

15-826

Copyright: C. Faloutsos (2011)

42

14

CMU SCS

R-trees - performance analysis

Results: 2D- uniform

leaf accesses

query side

15-826 Copyright: C. Faloutsos (2011) 43

CMU SCS

R-trees - performance analysis

Conclusions: usually, <5% relative error, for range queries

15-826 Copyright: C. Faloutsos (2011) 44

CMU SCS

Indexing - Detailed outline

- fractals
 - intro
 - applications
 - disk accesses for R-trees (range queries)
 - dimensionality reduction
 - selectivity in M-trees
 - dim. curse revisited
 - “fat fractals”
 - quad-tree analysis [Gaede+]
 -

15-826 Copyright: C. Faloutsos (2011) 45




CMU SCS

Case study #2: Dim. reduction

Problem definition: 'Feature selection'

- given N points, with E dimensions
- keep the k most 'informative' dimensions

[Traina+, SBBD'00]

Caetano
Traina

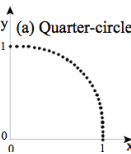
Agma
Traina

Leejay
Wu

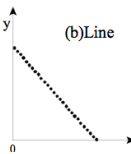
15-826 Copyright: C. Faloutsos (2011) 46

CMU SCS

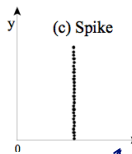
Dim. reduction - w/ fractals



(a) Quarter-circle



(b) Line



(c) Spike

not informative

15-826 Copyright: C. Faloutsos (2011) 47

CMU SCS

Dim. reduction


Problem definition: 'Feature selection'

- given N points, with E dimensions
- keep the k most 'informative' dimensions

Re-phrased: spot and drop attributes with strong (non-)linear correlations

Q: how do we do that?

15-826 Copyright: C. Faloutsos (2011) 48



CMU SCS

Dim. reduction

A: Hint: correlated attributes do not affect the intrinsic/fractal dimension, e.g., if


$$y = f(x,z,w)$$

we can drop y
(hence: ‘*partial fd*’ (PFD) of a set of attributes = the fd of the dataset, when projected on those attributes)

15-826

Copyright: C. Faloutsos (2011)

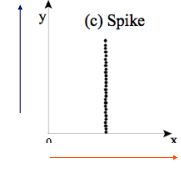
49



CMU SCS

Dim. reduction - w/ fractals

global FD=1
PFD=1




(c) Spike

PFD=0

15-826

Copyright: C. Faloutsos (2011)

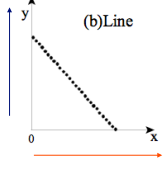
50



CMU SCS

Dim. reduction - w/ fractals

global FD=1
PFD=1



(b) Line

PFD=1

15-826

Copyright: C. Faloutsos (2011)

51

CMU SCS

Dim. reduction - w/ fractals

global FD=1 PFD~1

(a) Quarter-circle

15-826 Copyright: C. Faloutsos (2011) 52

CMU SCS

Dim. reduction - w/ fractals

- (problem: given N points in E -d, choose k best dimensions)
- Q: Algorithm?


15-826 Copyright: C. Faloutsos (2011) 53

CMU SCS

Dim. reduction - w/ fractals

- Q: Algorithm?
- A: e.g., greedy - forward selection:
 - keep the attribute with highest partial fd
 - add the one that causes the highest increase in pfd
 - etc., until we are within *epsilon* from the full f.d.

15-826 Copyright: C. Faloutsos (2011) 54




CMU SCS

Dim. reduction - w/ fractals

- (backward elimination: \sim reverse)
 - drop the attribute with least impact on the p.f.d.
 - repeat
 - until we are *epsilon* below the full f.d.

15-826 Copyright: C. Faloutsos (2011) 55




CMU SCS

Dim. reduction - w/ fractals

- Q: what is the smallest # of attributes we should keep?

15-826 Copyright: C. Faloutsos (2011) 56




CMU SCS

Dim. reduction - w/ fractals

- Q: what is the smallest # of attributes we should keep?
- A: we should keep at least as many as the f.d. (and probably, a few more)


15-826 Copyright: C. Faloutsos (2011) 57

CMU SCS

Dim. reduction - w/ fractals

- Results: E.g., on the ‘currency’ dataset
- (daily exchange rates for USD, HKD, BP, FRF, DEM, JPY - i.e., 6-d vectors, one per day - base currency: CAD)

e.g.: FRF




USD

15-826

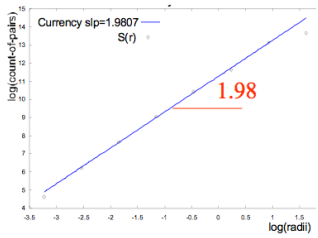
Copyright: C. Faloutsos (2011)

58

CMU SCS

E.g., on the ‘currency’ dataset

$\log(\#\text{pairs}(\leq r))$ correlation integral




Currency size=1,9807
 $S(r)$
1.98

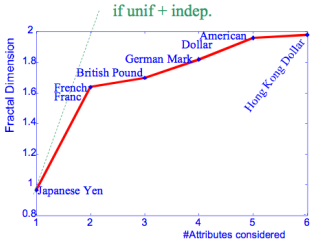
15-826

Copyright: C. Faloutsos (2011)

59

CMU SCS

E.g., on the ‘currency’ dataset



if unif + indep.

Fractal Dimension

#Attributes considered

Japanese Yen, French Franc, British Pound, German Mark, American Dollar, Hong Kong Dollar

15-826

Copyright: C. Faloutsos (2011)

60

20

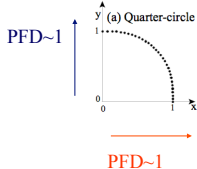
CMU SCS

Dim. reduction - w/ fractals

Conclusion:

- can do non-linear dim. reduction

global FD=1



15-826 Copyright: C. Faloutsos (2011) 61

CMU SCS

References

- [PODS94] Faloutsos, C. and I. Kamel (May 24-26, 1994). *Beyond Uniformity and Independence: Analysis of R-trees Using the Concept of Fractal Dimension*. Proc. ACM SIGACT-SIGMOD-SIGART PODS, Minneapolis, MN.
- [Traina+, SBBD'00] Traina, C., A. Traina, et al. (2000). *Fast feature selection using the fractal dimension*. XV Brazilian Symposium on Databases (SBBD), Paraiba, Brazil.

15-826 Copyright: C. Faloutsos (2011) 62
