CMU SCS

15-826: Multimedia Databases and Data Mining

Lecture #12: Fractals - case studies Part III
(regions, quadtrees, knn queries)
C. Faloutsos

CMU SCS

Must-read Material

- Alberto Belussi and Christos Faloutsos, Estimating the Selectivity of Spatial Queries Using the `Correlation' Fractal Dimension Proc. of VLDB, p. 299-310, 1995

15-826 2

CMU SCS

Optional Material

Optional, but very useful: Manfred Schroeder *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise* W.H. Freeman and Company, 1991

15-826 3

CMU SCS

Outline

Goal: 'Find similar / interesting things'

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

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
- ...

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+]
    - nn queries [Belussi+]

CMU SCS

'Fat' fractals & R-tree performance on region data

- Problem [Proietti+,'99]
- Given
  - N (# of data regions)
- estimate how many of them will qualify for the average range query (q1 x q2 x ... qE)

Of course, we need more info

Q: what?

CMU SCS

R-tree performance on region data

A: the distributions of their sizes

Q: do we also need some info about the locations?

CMU SCS

R-tree performance on region data

A: the distributions of their sizes

Q: do we also need some info about the locations?

A: no (not for range queries)

CMU SCS

R-tree performance on region data

A: the distributions of their sizes

Q: what exactly would we need?

CMU SCS

R-tree performance on region data

A: the distributions of their sizes

Q: what exactly would we need?

A: for self-similar regions (~ ‘fat’ fractals), we just need the slope of the Korcak law! (and the total area) [Proietti+]

CMU SCS

More power laws: areas – Korcak’s law

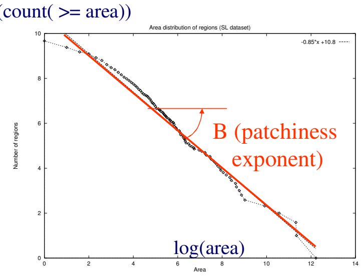
Scandinavian lakes

Any pattern?

CMU SCS

More power laws: areas – Korcak's law

log(count(>= area))

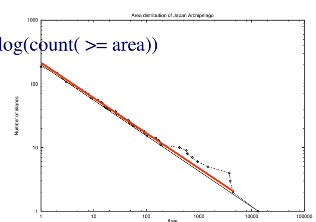


Scandinavian lakes area vs complementary cumulative count (log-log axes)

15-826 Copyright: C. Faloutsos (2009) 13

CMU SCS

More power laws: Korcak



Japan islands;

area vs cumulative count (log-log axes)

15-826 Copyright: C. Faloutsos (2009) 14

CMU SCS

Korcak's law & "fat fractals"

15-826 Copyright: C. Faloutsos (2009) 15

CMU SCS

R-tree performance on regions

- Once we know ‘B’ (and the total area)
- we can second-guess the individual sizes
- and then apply the [Pagel+93] formula
- Bottom line:

CMU SCS

R-tree performance on regions

Dataset	N	A	B
LAKES	816	75,910	0.85
ISLANDS	470	136,893	0.60
REGIONS	757	190,526	0.70

CMU SCS

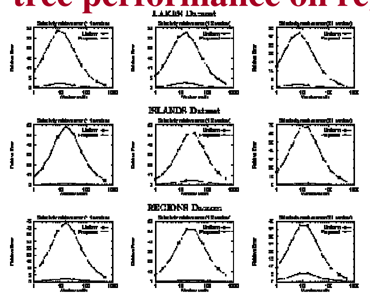
R-tree performance on regions

Dataset	N	A	B
LAKES	816	75,910	0.85
ISLANDS	470	136,893	0.60
REGIONS	757	190,526	0.70

CMU SCS

R-tree performance on regions

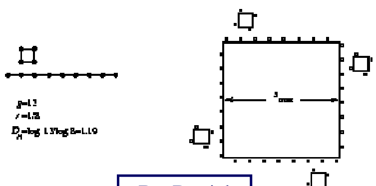
sel. error



query side

CMU SCS

‘Fat’ fractals - observation



$B = D_H / d$

B: patchiness exp; d: embedding dim

D_H: Hausdorff of periphery

CMU SCS

‘Fat’ fractals - observation

Dataset	D_H	B	$D_H - 2B$
LAKES	1.78	0.85	0.08
ISLANDS	1.23	0.60	0.03
REGIONS	1.48	0.70	0.08
Aegean Island	1.08	0.52	0.04
Japan archipelago	1.19	0.59	0.01
Italy plains	1.32	0.63	0.06
Whole Earth	1.2	0.6	0
Cypress vegetation	0.62	1.23	0.01

CMU SCS

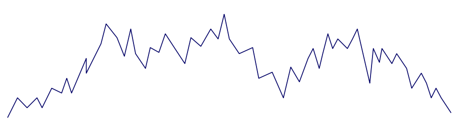
‘Fat’ fractals

- intuition behind $B = D_H / d$?

CMU SCS

‘Fat’ fractals

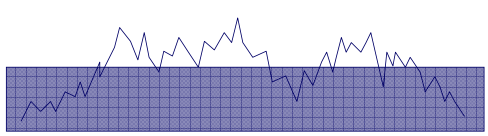
- intuition behind $B = D_H / d$?
- A: consider ‘flooding’:



CMU SCS

‘Fat’ fractals

- intuition behind $B = D_H / d$?
- A: consider ‘flooding’:



Conclusions

- 'Fat' fractals model regions well
- patchiness exp.: $B = D_H / d$
- can help us estimate selectivities

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+]
    - nn queries [Belussi+]

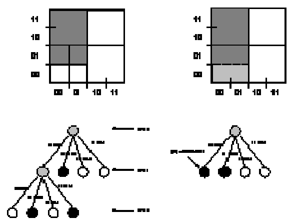
Fractals and Quadtrees

- Problem: how many quadtree nodes will we need, to store a region in some level of approximation? [Gaede+96]

CMU SCS

Fractals and Quadtrees

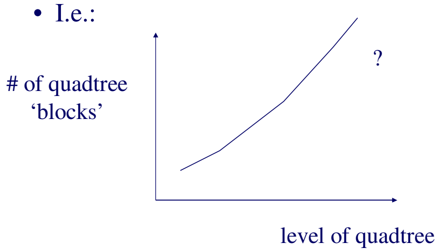
- I.e.:



CMU SCS

Fractals and Quadtrees

- I.e.:

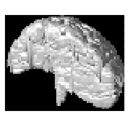


CMU SCS

Fractals and Quadtrees

- Datasets:

Franconia

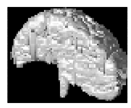


Brain Atlas

CMU SCS

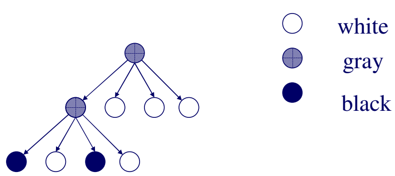
Fractals and Quadtrees

- Hint:
  - assume that the boundary is self-similar, with a given fd
  - how will the quad-tree (oct-tree) look like?



CMU SCS

Fractals and Quadtrees

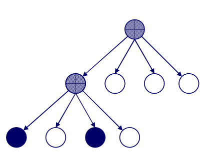


CMU SCS

Fractals and Quadtrees

Let $p_g(i)$ the prob. to find a gray node at level *i*.

If self-similar, what can we say for $p_g(i)$?

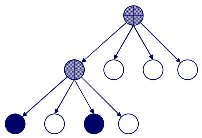


CMU SCS

Fractals and Quadtrees

Let $p_g(i)$ the prob. to find a gray node at level *i*.

If self-similar, what can we say for $p_g(i)$?



A: $p_g(i) = p_g$= constant

CMU SCS

Fractals and Quadtrees

Assume only ‘gray’ and ‘white’ nodes (ie., no volume’)

Assume that p_g is given - how many gray nodes at level *i*?

CMU SCS

Fractals and Quadtrees

Assume only ‘gray’ and ‘white’ nodes (ie., no volume’)

Assume that p_g is given - how many gray nodes at level *i*?

A: 1 at level 0;

$4*p_g$

$(4*p_g)*(4*p_g)$

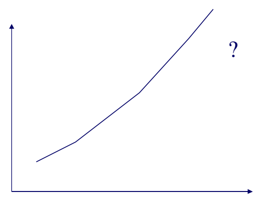
...

$(4*p_g)^i$

CMU SCS

Fractals and Quadtrees

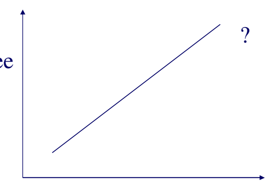
- I.e.:



CMU SCS

Fractals and Quadtrees

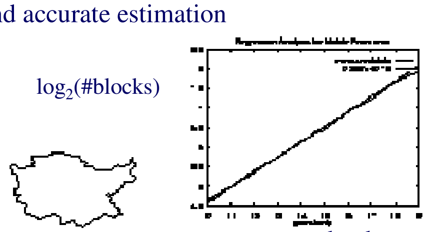
- I.e.:



CMU SCS

Fractals and Quadtrees

- Conclusion: Self-similarity leads to easy and accurate estimation

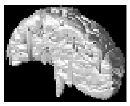


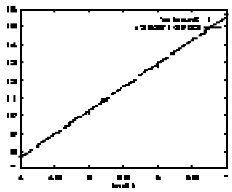
CMU SCS

Fractals and Quadtrees

- Conclusion: Self-similarity leads to easy and accurate estimation

log$_2$(#blocks)

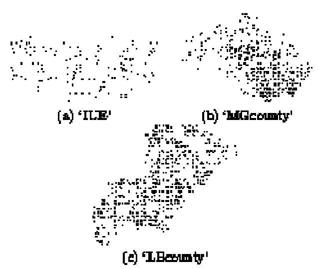




level

CMU SCS

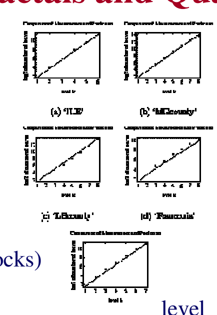
Fractals and Quadtrees



CMU SCS

Fractals and Quadtrees

log(#blocks)



level

CMU SCS

Fractals and Quadtrees

- Final observation: relationship between p_g and fractal dimension?

CMU SCS

Fractals and Quadtrees

- Final observation: relationship between p_g and fractal dimension?
- A: very close:

  $(4*p_g)^i$ = # of gray nodes at level i =

  # of Hausdorff grid-cells of side $(1/2)^i = r$

  Eventually: $D_H = 2 + \log_2(p_g)$

  and, for E-d spaces: $D_H = E + \log_2(p_g)$

CMU SCS

Fractals and Quadtrees

for E-d spaces: $D_H = E + \log_2(p_g)$

Sanity check:

- point: $D_H = 0$ p_g = ??

- line in 2-d: $D_H = 1$ p_g = ??

- plane in 2-d: D_H=2 p_g= ??

CMU SCS

Fractals and Quadtrees

Final conclusions:

- self-similarity leads to estimates for # of z-values = # of quadtree/oct-tree blocks
- close dependence on the Hausdorff fractal dimension of the boundary

15-826 Copyright: C. Faloutsos (2009) 46

CMU SCS

Indexing - Detailed outline

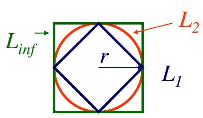
- fractals
  - intro
  - applications
    - disk accesses for R-trees (range queries)
    - dimensionality reduction
    - selectivity in M-trees
    - dim. curse revisited
    - "fat fractals"
    - quad-tree analysis [Gaede+]
    - nn queries [Belussi+]

15-826 Copyright: C. Faloutsos (2009) 47

CMU SCS

NN queries

- Q: in NN queries, what is the effect of the shape of the query region? [Belussi+95]

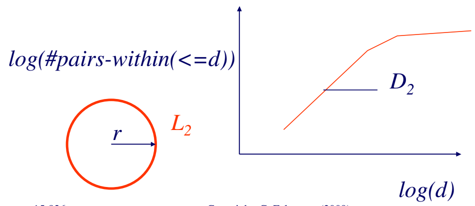


15-826 Copyright: C. Faloutsos (2009) 48

CMU SCS

NN queries

- Q: in NN queries, what is the effect of the shape of the query region?
- that is, for L2, and self-similar data:

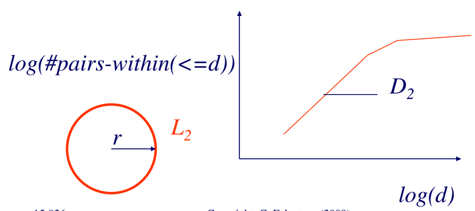


15-826 Copyright: C. Faloutsos (2009) 49

CMU SCS

NN queries

- Q: What about L_1, L_{inf}?

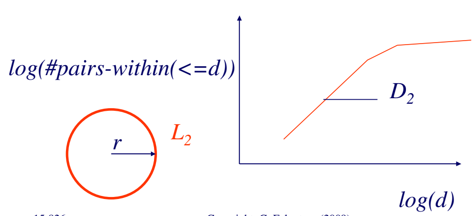


15-826 Copyright: C. Faloutsos (2009) 50

CMU SCS

NN queries

- Q: What about L_1, L_{inf}?
- A: Same slope, different intercept

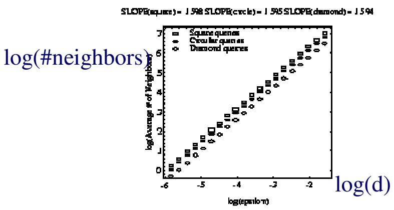


15-826 Copyright: C. Faloutsos (2009) 51

CMU SCS

NN queries

- Q: What about L_1, L_{inf}?
- A: **Same slope**, different intercept

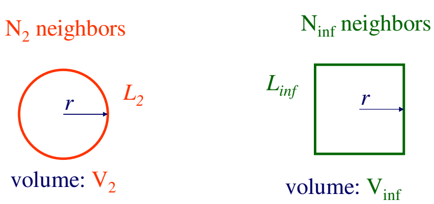


CMU SCS

SKIP

NN queries

- Q: what about the intercept? Ie., what can we say about N_2 and N_{inf}

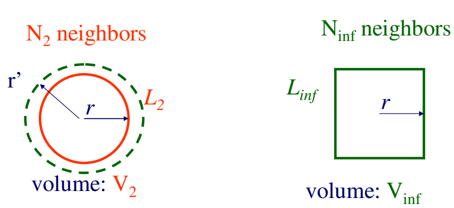


CMU SCS

SKIP

NN queries

- Consider sphere with volume V_{inf} and r' radius



CMU SCS

SKIP

NN queries

- Consider sphere with volume V_{inf} and r' radius
- $(r/r')^E = V_2 / V_{inf}$
- $(r/r')^{D_2} = N_2 / N_2'$
- $N_2' = N_{inf}$ (since shape does not matter)
- and finally:

15-826 Copyright: C. Faloutsos (2009) 55

CMU SCS

SKIP

NN queries

$(N_2 / N_{inf}) ^{1/D_2} = (V_2 / V_{inf}) ^{1/E}$

15-826 Copyright: C. Faloutsos (2009) 56

CMU SCS

NN queries

Conclusions: for self-similar datasets

- Avg # neighbors: grows like *(distance)^D_2*, regardless of query shape (circle, diamond, square, e.t.c.)

15-826 Copyright: C. Faloutsos (2009) 57

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+]
    - nn queries [Belussi+]
  - Conclusions

CMU SCS

Fractals - overall conclusions

- self-similar datasets: appear often
- powerful tools: correlation integral, NCDF, rank-frequency plot
- intrinsic/fractal dimension helps in
  - estimations (selectivities, quadtrees, etc)
  - dim. reduction / dim. curse
- (later: can help in image compression...)

CMU SCS

References

1. Belussi, A. and C. Faloutsos (Sept. 1995). Estimating the Selectivity of Spatial Queries Using the `Correlation' Fractal Dimension. Proc. of VLDB, Zurich, Switzerland.
2. Faloutsos, C. and V. Gaede (Sept. 1996). Analysis of the z-ordering Method Using the Hausdorff Fractal Dimension. VLDB, Bombay, India.
3. Proietti, G. and C. Faloutsos (March 23-26, 1999). I/O complexity for range queries on region data stored using an R-tree. International Conference on Data Engineering (ICDE), Sydney, Australia.