 CMU SCS

## 15-826: Multimedia Databases and Data Mining

Lecture #7: Spatial Access Methods - IV  
Grid files, dim. curse  
*C. Faloutsos*

---

---

---


---

---

---

---

---

 CMU SCS

## Reading material

- Textbook, Chapter 5.3
- Ramakrinshan+Gehrke, Chapter 28.5

15-826 Copyright: C. Faloutsos (2008) #2

---

---

---


---

---

---


---

---

 CMU SCS

## Outline

Goal: 'Find similar / interesting things'

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

15-826 Copyright: C. Faloutsos (2008) #3

---

---

---

---

---

---

---

---

CMU SCS

## Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
  - problem defn
  - z-ordering
  - R-trees
  - misc
- text
- ...

15-826 Copyright: C. Faloutsos (2008) #4

---

---

---

---

---

---

---

---

CMU SCS

## SAMs - Detailed outline

- spatial access methods
  - problem defn
  - z-ordering
  - R-trees
  - misc topics
    - grid files
    - dimensionality curse; dim. reduction
    - metric trees
    - other nn methods
- text, ...

15-826 Copyright: C. Faloutsos (2008) #5

---

---

---

---

---

---

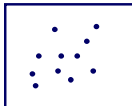
---

---

CMU SCS

## Grid files

- problem: spatial queries in  $k$ -d point-sets
- Main idea: try to generalize hashing to  $k$ -d
- (how?)



15-826 Copyright: C. Faloutsos (2008) #6

---

---

---

---

---

---

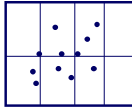
---

---

CMU SCS

### Grid files

- A: put a grid
- specs: [Nievergelt +, 84]
  - symmetric to all attributes
  - 2 disk accesses for exact match queries
  - adaptive to non-uniform distr.
- Q: details?



15-826 Copyright: C. Faloutsos (2008) #7

---

---

---

---

---

---

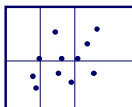
---

---

CMU SCS

### Grid files

- cuts: all the way through
- cuts: at  $\frac{1}{2}$ ,  $\frac{3}{4}$ ,  $\frac{1}{4}$  etc; but on demand
- each cell  $\rightarrow$  disk page



15-826 Copyright: C. Faloutsos (2008) #8

---

---

---

---

---

---

---

---

CMU SCS

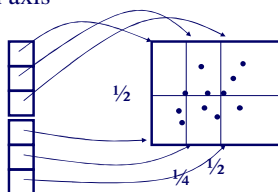
### Grid files

Thus, we only need:

- cut-points for each axis
- k-d directory

x-cuts  $\frac{1}{4}$   $\frac{1}{2}$

y-cuts  $\frac{1}{2}$



15-826 Copyright: C. Faloutsos (2008) #9

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

Search ( for exact match) – eg., (0.3; 0.3)

15-826 Copyright: C. Faloutsos (2008) #10

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

Search ( for exact match) – eg., (0.3; 0.3)

15-826 Copyright: C. Faloutsos (2008) #11

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

- specs: [Nievergelt +, 84]
  - symmetric to all attributes
  - ✗ 2 disk accesses for exact match queries
  - adaptive to non-uniform distr.

15-826 Copyright: C. Faloutsos (2008) #12

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

partial match – eg.,  $0 < x < 0.3$

x-cuts  $\frac{1}{4}$   $\frac{1}{2}$

y-cuts  $\frac{1}{2}$

15-826 Copyright: C. Faloutsos (2008) #13

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

partial match – eg.,  $0 < x < 0.3$

x-cuts  $\frac{1}{4}$   $\frac{1}{2}$

y-cuts  $\frac{1}{2}$

15-826 Copyright: C. Faloutsos (2008) #14

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

exactly the symmetric algo for eg.,  $0 < y < 0.3$

15-826 Copyright: C. Faloutsos (2008) #15

---

---

---

---

---

---

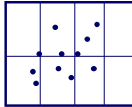
---

---

CMU SCS

### Grid files

- specs: [Nievergelt +, 84]
  - ✗ symmetric to all attributes
  - ✗ 2 disk accesses for exact match queries
    - adaptive to non-uniform distr.



15-826 Copyright: C. Faloutsos (2008) #16

---

---

---

---

---

---

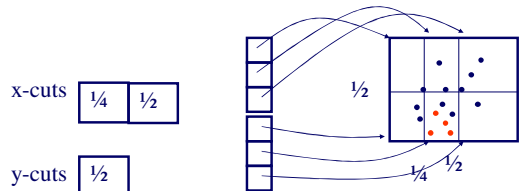
---

---

CMU SCS

### Grid files

Q: How to split an overflowing page?



x-cuts  $\frac{1}{4}$   $\frac{1}{2}$

y-cuts  $\frac{1}{2}$

15-826 Copyright: C. Faloutsos (2008) #17

---

---

---

---

---

---

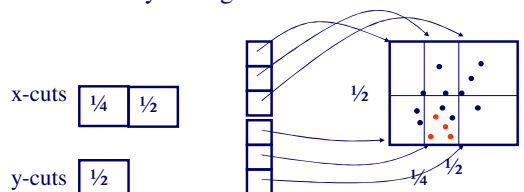
---

---

CMU SCS

### Grid files

A: pick the 'best' axis, and cut all the way through



x-cuts  $\frac{1}{4}$   $\frac{1}{2}$

y-cuts  $\frac{1}{2}$

15-826 Copyright: C. Faloutsos (2008) #18

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

A: pick the 'best' axis, and cut all the way through...

x-cuts  $\frac{1}{4}$   $\frac{3}{8}$   $\frac{1}{2}$

y-cuts  $\frac{1}{2}$

15-826 Copyright: C. Faloutsos (2008) #19

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

... updating the directory appropriately (ouch!)

x-cuts  $\frac{1}{4}$   $\frac{3}{8}$   $\frac{1}{2}$

y-cuts  $\frac{1}{2}$

15-826 Copyright: C. Faloutsos (2008) #20

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

- specs: [Nievergelt +, 84]
  - ✗ symmetric to all attributes
  - ✗ 2 disk accesses for exact match queries
  - ✗ adaptive to non-uniform distr.

15-826 Copyright: C. Faloutsos (2008) #21

---

---

---

---

---

---

---

---

CMU SCS

### Grid files

- it meets the three goals
- had follow-up work [twin grid files, multi-level; etc]
- BUT: has some disadvantages (which ones?)

15-826 Copyright: C. Faloutsos (2008) #22

---

---

---

---

---

---

---

---

CMU SCS

### Grid files - disadvantages

- #1: problems in high-d: directory splits can be expensive

15-826 Copyright: C. Faloutsos (2008) #23

---

---

---

---

---

---

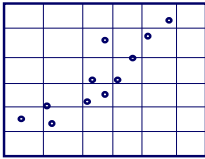
---

---

CMU SCS

### Grid files - disadvantages

- #2: even in low-d, suffers on correlated attributes:



15-826 Copyright: C. Faloutsos (2008) #24

---

---

---

---

---

---

---

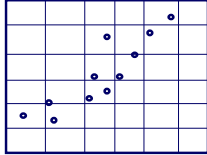
---



CMU SCS

### Grid files - disadvantages

- (Q: how to fix, for 2-d, linearly correlated points?)



15-826 Copyright: C. Faloutsos (2008) #25

---

---

---

---

---

---

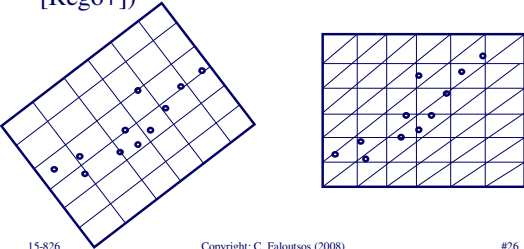
---

---

CMU SCS

### Grid files - disadvantages

- (A1: rotate [Hinrichs+]; A2: triangular cells [Rego+])



15-826 Copyright: C. Faloutsos (2008) #26

---

---

---

---

---

---

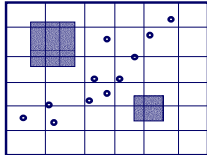
---

---

CMU SCS

### Grid files - disadvantages

- #3: how about region data?



15-826 Copyright: C. Faloutsos (2008) #27

---

---

---

---

---

---

---

---

CMU SCS

### Grid files - disadvantages

- #3: how about region data?
- if we 'cut' them, then we have  $O(\text{volume})$  pieces (while z-ordering:  $O(\text{surface})$ )
- what to do?

15-826 Copyright: C. Faloutsos (2008) #28

---

---

---

---

---

---

---

---

CMU SCS

### Grid files - disadvantages

- what to do?
- Translation to  $2k - d$  points! (clever, BUT, still has subtle problems) E.g., 1-d 'regions'

15-826 Copyright: C. Faloutsos (2008) #29

---

---

---

---

---

---

---

---

CMU SCS

### Grid files - disadvantages

- what to do?
- Translation to  $2k - d$  points! (clever, BUT, still has subtle problems) E.g., 1-d 'regions'

15-826 Copyright: C. Faloutsos (2008) #30

---

---

---

---

---

---

---

---

CMU SCS

### Grid files - disadvantages

- what to do?
- Translation to 2k – d points! (clever, BUT, still has subtle problems) E.g., 1-d ‘regions’

15-826 Copyright: C. Faloutsos (2008) #31

---

---

---

---

---

---

---

---

---

---

CMU SCS

### Grid files - disadvantages

- what is the problem, then?

15-826 Copyright: C. Faloutsos (2008) #32

---

---

---

---

---

---

---

---

---

---

CMU SCS

### Grid files - disadvantages

- what is the problem, then?
- A: dimensionality curse; large query regions

15-826 Copyright: C. Faloutsos (2008) #33

---

---

---

---

---

---

---

---

---

---

CMU SCS

### Grid files – conclusions

- works OK in low-d un-correlated points
- but z-ordering/R-trees seem to work better for higher-d
- smart idea to translate k-d rectangles into  $2^k$  - points (but: dim. curse)

15-826 Copyright: C. Faloutsos (2008) #34

---

---

---

---

---

---

---

---

CMU SCS

### SAMs - Detailed outline

- spatial access methods
  - problem defn
  - z-ordering
  - R-trees
  - misc topics
    - grid files
    - dimensionality curse; dim. reduction
    - metric trees
    - other nn methods
- text, ...

15-826 Copyright: C. Faloutsos (2008) #35

---

---

---

---

---

---

---

---

CMU SCS

### Dimensionality ‘curse’

- Q: What is the problem in high-d?

15-826 Copyright: C. Faloutsos (2008) #36

---

---

---

---

---

---

---

---

CMU SCS

## Dimensionality ‘curse’

- Q: What is the problem in high-d?
- A: indices do not seem to help, for many queries (eg., k-nn)
  - in high-d (& uniform distributions), most points are equidistant -> k-nn retrieves too many near-neighbors
  - [Yao & Yao, '85]: search effort  $\sim O(N^{(1-1/d)})$

15-826 Copyright: C. Faloutsos (2008) #37

---

---

---

---

---

---

---

---

CMU SCS

## Dimensionality ‘curse’

- (counter-intuitive, for db mentality)
- Q: What to do, then?

15-826 Copyright: C. Faloutsos (2008) #38

---

---

---

---

---

---

---

---

CMU SCS

## Dimensionality ‘curse’

- A1: switch to seq. scanning
- A2: dim. reduction
- A3: consider the ‘intrinsic’/fractal dimensionality
- A4: find approximate nn

15-826 Copyright: C. Faloutsos (2008) #39

---

---

---

---

---

---

---

---

CMU SCS

## Dimensionality 'curse'

- A1: switch to seq. scanning
  - X-trees [Kriegel+, VLDB 96]
  - VA-files [Schek+, VLDB 98]

15-826 Copyright: C. Faloutsos (2008) #40

---

---

---

---

---

---

---

---

CMU SCS

## Dimensionality 'curse'

- A1: switch to seq. scanning
- ➔ • A2: dim. reduction
- A3: consider the 'intrinsic'/fractal dimensionality
- A4: find approximate nn

15-826 Copyright: C. Faloutsos (2008) #41

---

---

---

---

---

---

---

---

CMU SCS

## Dim. reduction

a.k.a. feature selection/extraction:

- SVD (optimal, to preserve Euclidean distances)
- random projections
- using the fractal dimension [Traina+ SBBD2000]

15-826 Copyright: C. Faloutsos (2008) #42

---

---

---

---

---

---

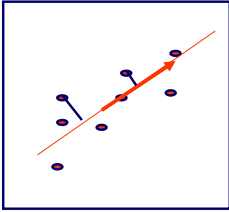
---

---

CMU SCS

## Singular Value Decomposition (SVD)

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



LSI: S. Dumais; M. Berry  
 KL: eg, Duda+Hart  
 PCA: eg., Jolliffe  
 MANY more details: soon

15-826 Copyright: C. Faloutsos (2008) #43

---

---

---

---

---

---

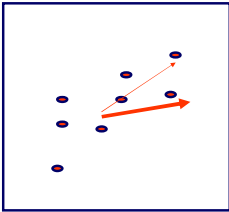
---

---

CMU SCS

## Random projections

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



15-826 Copyright: C. Faloutsos (2008) #44

---

---

---

---

---

---

---

---

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])

15-826 Copyright: C. Faloutsos (2008) #45

---

---

---

---

---

---

---

---

CMU SCS

## Dim. reduction - w/ fractals

- Main idea: drop those attributes that don't affect the intrinsic ('fractal') dimensionality [Traina+, SBBB 2000]

15-826 Copyright: C. Faloutsos (2008) #46

---

---

---

---

---

---

---

---

CMU SCS

## Dim. reduction - w/ fractals

global FD=1

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

PFD~1

PFD~1

15-826 Copyright: C. Faloutsos (2008) #47

---

---

---

---

---

---

---

---

CMU SCS

## Dimensionality 'curse'

- A1: switch to seq. scanning
- A2: dim. reduction
- ➔ A3: consider the 'intrinsic'/fractal dimensionality
- A4: find **approximate** nn

15-826 Copyright: C. Faloutsos (2008) #48

---

---

---

---

---

---

---

---





CMU SCS

## Intrinsic dimensionality

- before we give up, compute the intrinsic dim.:
- the lower, the better... [Pagel+, ICDE 2000]
- more details: under 'fractals'

intr. d = 2





intr. d = 1

15-826 Copyright: C. Faloutsos (2008) #49

---

---

---

---

---

---

---

---

CMU SCS

## Dimensionality 'curse'

- A1: switch to seq. scanning
- A2: dim. reduction
- A3: consider the 'intrinsic'/fractal dimensionality
- ➔ A4: find approximate nn

15-826 Copyright: C. Faloutsos (2008) #50

---

---

---

---

---

---

---

---

CMU SCS

## Approximate nn

- [Arya + Mount, SODA93], [Patella+ ICDE 2000]
- Idea: find k neighbors, such that the distance of the k-th one is guaranteed to be within epsilon of the actual.

15-826 Copyright: C. Faloutsos (2008) #51

---

---

---

---

---

---

---

---

CMU SCS

## SAMs - Detailed outline

- spatial access methods
  - problem defn
  - z-ordering
  - R-trees
  - misc topics
    - grid files
    - dimensionality curse; dim. reduction
    - metric trees
    - other nn methods
- text, ...

15-826 Copyright: C. Faloutsos (2008) #52

---

---

---

---

---

---

---

---

---

---

CMU SCS

## Conclusions

- Dimensionality ‘curse’:
  - for high-d, indices slow down to  $\sim O(N)$
- If the **intrinsic** dim. is low, there is hope
- otherwise, do seq. scan, or sacrifice accuracy (approximate nn)

15-826 Copyright: C. Faloutsos (2008) #53

---

---

---

---

---

---

---

---

---

---

CMU SCS

## References

- Berchtold, S., D. A. Keim, et al. (1996). The X-tree : An Index Structure for High-Dimensional Data. VLDB, Mumbai (Bombay), India.
- Faloutsos, C. and W. Rego (1989). “Tri-cell: A Data Structure for Spatial Objects.” Information Systems 14(2): 131-139.
- Hinrichs, K. and J. Nievergelt (1983). The Grid File: A Data Structure to Support Proximity Queries on Spatial Objects. Proc. of the WG’83 (Intern. Workshop on Graph Theoretic Concepts in Computer Science), Linz, Austria, Trauner Verlag.

15-826 Copyright: C. Faloutsos (2008) #54

---

---

---

---

---

---

---

---

---

---

CMU SCS

## References cnt'd

- Nievergelt, J., H. Hinterberger, et al. (March 1984). "The Grid File: An Adaptable, Symmetric Multikey File Structure." ACM TODS 9(1): 38-71.
- Papadimitriou, C. H., P. Raghavan, et al. (1998). Latent Semantic Indexing: A Probabilistic Analysis. PODS, Seattle, WA.

15-826 Copyright: C. Faloutsos (2008) #55

---

---

---

---

---

---

---

---

CMU SCS

## References cnt'd

- Weber, R., H.-J. Schek, et al. (1998). A Quantitative Analysis and Performance Study for Similarity-Search Methods in high-dimensional spaces. VLDB, New York, NY.
- Yao, A. C. and F. F. Yao (May 6-8, 1985). A General Approach to d-Dimensional Geometric Queries. Proc. of the 17th Annual ACM Symposium on Theory of Computing (STOC), Providence, RI.

15-826 Copyright: C. Faloutsos (2008) #56

---

---

---

---

---

---

---

---