must-read material

- MM Textbook, Chapter 5
- Roberto F. Santos Filho, Agma Traina, Caetano Traina Jr., and Christos Faloutsos: *Similarity search without tears: the OMNI family of all-purpose access methods* ICDE, Heidelberg, Germany, April 2-6 2001. (code at [www.cs.cmu.edu/~christos/SRC/OmniUsrKit.tar.gz](www.cs.cmu.edu/~christos/SRC/OmniUsrKit.tar.gz))
Indexing - Detailed outline

• primary key indexing
• secondary key / multi-key indexing
• spatial access methods
  – problem dfn
  – z-ordering
  – R-trees
  – misc
• fractals
• text

SAMs - Detailed outline

• spatial access methods
  – problem dfn
  – z-ordering
  – R-trees
  – misc topics
    • metric trees
• fractals
• text, ...

Metric trees

• What if we only have a distance function $d(o1, o2)$?
• (Applications?)
Metric trees

• (assumption: d() is a metric: positive; symmetric; triangle inequality)
• then, we can use some variation of ‘Vantage Point’ trees [Yannilos]
• many variations (GNAT trees [Brin95], MVP-trees [Ozsoyoglu+] ...)

Metric trees

• Finally: M-trees [Ciaccia, Patella, Zezula, vldb 97]
• M-trees = ‘ball-trees’: groups in spheres

Metric trees

• Finally: M-trees [Ciaccia, Patella, Zezula, vldb 97]
• M-trees = ‘ball-trees’: Minimum Bounding spheres
Metric trees

- Finally: M-trees [Ciaccia, Patella, Zezula, vldb 97]
- M-trees = ‘ball-trees’: Minimum Bounding spheres

Metric trees

- Finally: M-trees [Ciaccia, Patella, Zezula, vldb 97]
- M-trees = ‘ball-trees’: Minimum Bounding spheres

Metric trees

- Search (range and k-nn): like R-trees
  - Split?
Metric trees

- Search (range and k-nn): like R-trees
- Split? Several criteria:
  - minimize max radius (or sum radii)
  - (even: random!)
- Algorithm?

Metric trees

- Search (range and k-nn): like R-trees
- Split? Several criteria:
  - minimize max radius (or sum radii)
  - (even: random!)
- Algorithm?
  - eg., similar to the quadratic split of Guttman

Metric trees - variations

- OMNI tree [Filho+, ICDE2001]
Metric trees - OMNI trees

- How to turn objects into vectors?
- (assume that distance computations are expensive; we need to answer range/nn queries quickly)

A: pick \( n \) ‘anchor’ objects; record the distance of each object from them \( \rightarrow n\)-d vector
How to turn objects into vectors?
- A: pick $n$ `anchor` objects; record the distance of each object from them $-> n$-d vector

OMNI trees – range queries
- and still answer range and nn queries! (see [Filho’01] for details)
OMNI trees – range queries

- Result: faster than M-trees and seq. scanning (especially if distance computations are expensive)

Metric trees - OMNI trees

- Q1: how to choose anchors?
- Q2: ... and how many?

Conclusions for SAMs

- z-ordering and R-trees for low-d points and regions – very successful
- M-trees & variants for metric datasets
- beware of the ‘dimensionality curse’
  - Estimate ‘intrinsic’ dimensionality (‘fractals’)
  - Project to lower dimensions (‘SVD/PCA’)
## References

References