Carnegie Mellon

15-826: Multimedia Databases and Data Mining

Lecture #7: Spatial Access Methods - Metric trees

C. Faloutsos

Carnegie Mellon

Optional material

- Ciaccia, P., M. Patella, et al. (1997). M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. VLDB.

Carnegie Mellon

Outline

Goal: 'Find similar / interesting things'

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

Carnegie Mellon

Indexing - Detailed outline

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

Carnegie Mellon

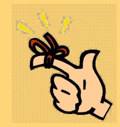
SAMs - Detailed outline

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

Carnegie Mellon

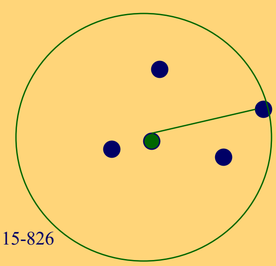
Metric trees - problem

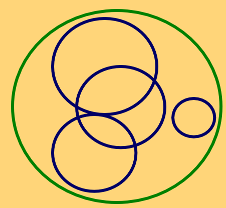
- What if we only have a distance function d(o1, o2)?
- (Applications?)



A: Metric trees

- M-trees = 'ball-trees' : Minimum Bounding spheres





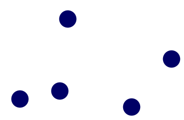
Metric trees

- (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+] ...)

Carnegie Mellon

Metric trees

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

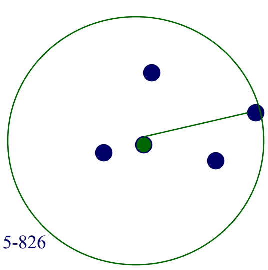


15-826 Copyright: C. Faloutsos (2019) #9

Carnegie Mellon

Metric trees

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

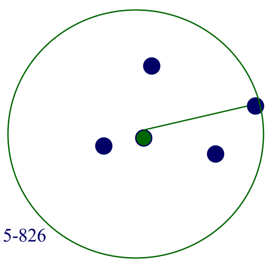


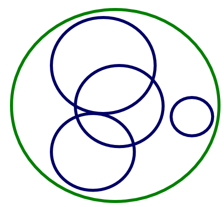
15-826 Copyright: C. Faloutsos (2019) #10

Carnegie Mellon

Metric trees

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



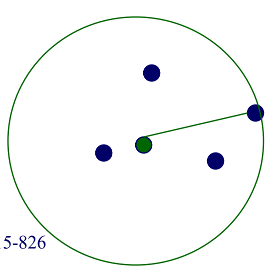


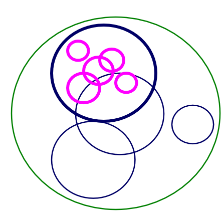
15-826 Copyright: C. Faloutsos (2019) #11

Carnegie Mellon

Metric trees

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





15-826 Copyright: C. Faloutsos (2019) #12

Carnegie Mellon

Metric trees

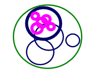
- Search (range and k-nn): like R-trees
- Split?

Carnegie Mellon

Metric trees

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

Carnegie Mellon



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

Carnegie Mellon

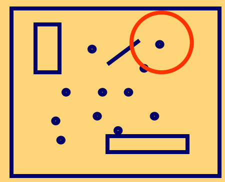
SAMs - Detailed outline

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

Carnegie Mellon

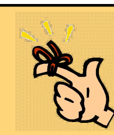
Spatial Access Methods - problem

- Given a collection of geometric objects (points, lines, polygons, ...)
- Find cities within 100mi from Pittsburgh



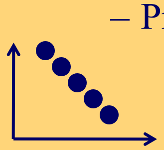
15-826 Copyright: C. Faloutsos (2019) 17

Carnegie Mellon



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’)



15-826 Copyright: C. Faloutsos (2019) #18

Carnegie Mellon

References

- Christian Böhm, Stefan Berchtold, Daniel A. Keim: *Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases*. ACM Comput. Surv. 33(3): 322-373 (2001)
- Edgar Chávez, Gonzalo Navarro, Ricardo A. Baeza-Yates, José L. Marroquín: *Searching in metric spaces*. ACM Comput. Surv. 33(3): 273-321 (2001)

Carnegie Mellon

References

- Ciaccia, P., M. Patella, et al. (1997). M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. VLDB.
- Filho, R. F. S., A. Traina, et al. (2001). Similarity search without tears: the OMNI family of all-purpose access methods. ICDE, Heidelberg, Germany.
- Friedman, J. H., F. Baskett, et al. (Oct. 1975). "An Algorithm for Finding Nearest Neighbors." IEEE Trans. on Computers (TOC) C-24: 1000-1006.