15-826: Multimedia Databases and Data Mining

Lecture #11: Fractals - case studies Part III (regions, quadtrees, knn queries)

C. Faloutsos

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

Optional Material

Outline

Goal: ‘Find similar / interesting things’
• Intro to DB
• Indexing - similarity search
• Data Mining

Indexing - Detailed outline

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

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+]
‘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 ($q_1 \times q_2 \times \ldots \times q_E$)

Of course, we need more info
Q: what?

R-tree performance on region data

A: the distributions of their sizes

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

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)
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+]

More power laws: areas – Korcak’s law
Scandinavian lakes
Any pattern?
More power laws: areas – Korcak’s law

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

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:

<table>
<thead>
<tr>
<th>Dataset</th>
<th>N</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAKES</td>
<td>816</td>
<td>75,910</td>
<td>0.85</td>
</tr>
<tr>
<td>ISLANDS</td>
<td>470</td>
<td>136,893</td>
<td>0.60</td>
</tr>
<tr>
<td>REGIONS</td>
<td>757</td>
<td>190,526</td>
<td>0.70</td>
</tr>
</tbody>
</table>
R-tree performance on regions

'Fat' fractals - observation

<table>
<thead>
<tr>
<th>Dataset</th>
<th>α</th>
<th>β</th>
<th>0.50 + 0.10</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAKES</td>
<td>1.78</td>
<td>0.85</td>
<td>0.06</td>
</tr>
<tr>
<td>ISLANDS</td>
<td>1.53</td>
<td>0.65</td>
<td>0.03</td>
</tr>
<tr>
<td>REGIONS</td>
<td>1.08</td>
<td>0.72</td>
<td>0.06</td>
</tr>
<tr>
<td>Anrana Stricta</td>
<td>1.09</td>
<td>0.57</td>
<td>0.06</td>
</tr>
<tr>
<td>Insects &amp; Leaves</td>
<td>1.13</td>
<td>0.60</td>
<td>0.00</td>
</tr>
<tr>
<td>Holy Places</td>
<td>1.22</td>
<td>0.62</td>
<td>0.00</td>
</tr>
<tr>
<td>Whole Earth</td>
<td>1.92</td>
<td>0.56</td>
<td>0.00</td>
</tr>
<tr>
<td>Cypress vegetation</td>
<td>0.90</td>
<td>0.95</td>
<td>0.00</td>
</tr>
</tbody>
</table>
‘Fat’ fractals

• intuition behind $B = D_H / d$?

• A: consider ‘flooding’:

Optional
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]
Fractals and Quadtrees

• I.e.: 

![Diagram](image.png)

# of quadtree ‘blocks’
(= # gray nodes)

level of quadtree

Optional

Fractals and Quadtrees

• Datasets:

Franconia

Brain Atlas

Optional
Fractals and Quadtrees

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

Optional

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 $p_g(i)$?
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: $p_g(i) = p_g = \text{constant}$

Fractals and Quadtrees

Assume only 'gray' and 'white' nodes (i.e., no volume)

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

A: 1 at level 0;

\[
4p_g (4p_g)^i (4p_g) \\
\vdots \\
(4p_g)^i
\]
Fractals and Quadtrees

• I.e.:

\[ \text{level of quadtree ('i')} \]

# of quadtree ‘blocks’

\[ (4^*p^i) \]

Fractals and Quadtrees

• I.e.:

\[ \text{level of quadtree} \]

\[ \log(\text{# of quadtree ‘blocks’}) \]

\[ \log[(4^*p^i)] \]

Fractals and Quadtrees

• Conclusion: Self-similarity leads to easy and accurate estimation
Fractals and Quadtrees

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

\[ \log_2(\#\text{blocks}) \]

level
Fractals and Quadtrees

• Final observation: relationship between $p_g$ and fractal dimension?

• A: very close:
  
  $(4^i p_g)^i = \#$ 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: $D_H = E + \log_2(p_g)$

Sanity check:

- point in 2-d: $D_H = 0$  \[ p_g = ?? \]
- line in 2-d: $D_H = 1$  \[ p_g = ?? \]
- plane in 2-d: $D_H = 2$  \[ p_g = ?? \]
- point in 3-d: $D_H = 0$  \[ p_g = ?? \]
Fractals and Quadtrees

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

Sanity check:
- point in 2-d: \( D_H = 0 \) \( p_g = 1/4 \)
- line in 2-d: \( D_H = 1 \) \( p_g = 1/2 \)
- plane in 2-d: \( D_H = 2 \) \( p_g = 1 \)
- point in 3-d: \( D_H = 0 \) \( p_g = 1/8 \)

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

Indexing - Detailed outline

• fractals
  – intro
  – applications
    ✓ disk accesses for R-trees (range queries)
    ✓ dimensionality reduction
    ✓ selectivity in M-trees
    ✓ “fat fractals”
    ✓ quad-tree analysis [Gaede+]
  • nn queries [Belussi+]
**NN queries**

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

![Diagram of L2, L1, and L_inf norms with a graph showing log(#pairs-within(<=d)) vs. log(d)]

**Optional**

- that is, for L2, and self-similar data:

![Graph showing log(#pairs-within(<=d)) vs. log(d)]

**NN queries**

- Q: What about L1, L_inf?

![Graph showing log(#pairs-within(<=d)) vs. log(d)]
NN queries

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

\[ \log(\#\text{pairs-within}(<d)) \]

$D_2$

$\log(d)$

$\log(\#\text{neighbors})$

Q: what about the intercept? I.e., what can we say about $N_2$ and $N_{\infty}$

$N_2$ neighbors

$r \quad L_2$

volume: $V_2$

$N_{\infty}$ neighbors

$r \quad L_{\infty}$

volume: $V_{\infty}$
NN queries

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

\[ \frac{(r/r')^E}{V_{\text{inf}}} = \frac{N_2^2}{N_{\text{inf}}^2} \]

- $N_2' = N_{\text{inf}}$ (since shape does not matter)

- And finally:

\[ \left( \frac{N_2}{N_{\text{inf}}} \right)^{1/D_2} = \left( \frac{V_2}{V_{\text{inf}}} \right)^{1/E} \]
NN queries
Conclusions: for self-similar datasets
• Avg # neighbors: grows like \((\text{distance})^{D_s}\), regardless of query shape (circle, diamond, square, etc.)

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

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...)
References