Principles of Database Systems

V. Megalooikonomou

Fractals and Databases

(based on notes by C. Faloutsos at CMU)
Indexing - Detailed outline

- fractals
  - intro
  - applications
Intro to fractals - outline

- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More examples and tools
- Discussion - putting fractals to work!
- Conclusions – practitioner’s guide
- Appendix: gory details - boxcounting plots
Road end-points of Montgomery county:

• Q1: how many d.a. for an R-tree?

• Q2: distribution?
  • not uniform
  • not Gaussian
  • no rules??
Problem #2 - spatial d.m.

Galaxies (Sloan Digital Sky Survey - B. Nichol)

- ‘spiral’ and ‘elliptical’ galaxies
  (stores and households ...)
- patterns?
- attraction/ repulsion?
- how many ‘spi’ within r from an ‘ell’?
Problem #3: traffic

- disk trace (from HP - J. Wilkes); Web traffic - fit a model

- how many explosions to expect?
- queue length distr.?
Common answer:

- Fractals / self-similarities / power laws
- Seminal works from Hilbert, Minkowski, Cantor, Mandelbrot, (Hausdorff, Lyapunov, Ken Wilson, …)
Road map

- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More examples and tools
- Discussion - putting fractals to work!
- Conclusions – practitioner’s guide
- Appendix: gory details - boxcounting plots
What is a fractal?

= self-similar point set, e.g., Sierpinski triangle:

zero area; infinite perimeter!
Definitions (cont’d)

- Paradox: Infinite perimeter; Zero area!
- ‘dimensionality’: between 1 and 2
- actually: $\log(3)/\log(2) = 1.58...$
Dfn of fd:

**ONLY** for a perfectly self-similar point set:

\[ \frac{\log(n)}{\log(f)} = \log(3)/\log(2) = 1.58 \]

a perfectly self-similar object with \( n \) similar pieces each scaled down by a factor \( f \)

Zero area; infinite length!
Intrinsic (‘fractal’) dimension

- Q: fractal dimension of a line?
- A: 1 (= \(\log(2)/\log(2)\))
Intrinsic (‘fractal’) dimension

Q: dfn for a given set of points?

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
Intrinsic (‘fractal’) dimension

- Q: fractal dimension of a line?
- A: \( \text{nn (} \leq r \text{)} \sim r^1 \) ('power law': \( y = x^a \))

- Q: fd of a plane?
- A: \( \text{nn (} \leq r \text{)} \sim r^2 \) \( \text{fd} = \) slope of (log(nn) vs log(r))
Intrinsic (‘fractal’) dimension

- Algorithm, to estimate it?

Notice

- \( \text{avg } nn(<=r) \) is exactly
  \[
  \frac{\text{tot#pairs}(<=r)}{(2*N)}
  \]
  including ‘mirror’ pairs
Sierpinsky triangle

\[ \log(\text{#pairs within } \leq r) \]

\[ \text{\textasciitilde correlation integral} \]

\[ \log(r) \]

1.58
Observations:

- Euclidean objects have integer fractal dimensions
  - point: 0
  - lines and smooth curves: 1
  - smooth surfaces: 2
- fractal dimension -> roughness of the periphery
Important properties

- $fd = embedding\ dimension\ -> uniform\ pointset$
- a point set may have several $fd$, depending on scale
Road map

- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More examples and tools
- Discussion - putting fractals to work!
- Conclusions – practitioner’s guide
- Appendix: gory details - boxcounting plots
Problem #1: GIS points

Cross-roads of Montgomery county:

• any rules?
Solution #1

\[ \log(#\text{pairs(within } \leq r)) \]

A: self-similarity ->
- \( \leftrightarrow \) fractals
- \( \leftrightarrow \) scale-free
- \( \leftrightarrow \) power-laws
  \( y = x^a, \ F = C \cdot r^{-(\text{-}2)} \)
- \( \text{avg#neighbors}(\leq r) = r^D \)
Solution #1

\[ \log(\#\text{pairs(within } \leq r)) \]

A: self-similarity
- \( \text{avg\#neighbors}(\leq r) \sim r^{1.51} \)

SLOPE = 1.51847

\[ \log(s2) \]

\[ \log(r) \]
Examples: MG county

- Montgomery County of MD (road endpoints)
Examples: LB county

- Long Beach county of CA (road endpoints)
Solution#2: spatial d.m.

Galaxies (‘BOPS’ plot - [sigmod2000])
Solution #2: spatial d.m.

\[ \log(\text{#pairs within } \leq r) \]

- 1.8 slope
- plateau!
- repulsion!

![Graph showing log(r) vs. log(#pairs within r)]
spatial d.m.

$log(\#\text{pairs within } \leq r)$

- 1.8 slope
- plateau!
- repulsion!

log(r)
spatial d.m.

Heuristic on choosing # of clusters
spatial d.m.

\[ \log(\# \text{pairs within } \leq r) \]

- 1.8 slope
- plateau!
- repulsion!

\[ \log(r) \]
spatial d.m.

$$\log(\# \text{pairs within } \leq r)$$

- 1.8 slope
- plateau!
- repulsion
- duplicates

[Diagram showing log-log plot with curves for different pairs: spi-spi, ell-ell, spi-ell, demonstrating repulsion and plateau effects.]
Solution #3: traffic

- disk traces: self-similar:

```
#bytes
```

```
<table>
<thead>
<tr>
<th>time</th>
<th>#bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
</tr>
<tr>
<td>3</td>
<td>300</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
</tr>
<tr>
<td>5</td>
<td>500</td>
</tr>
</tbody>
</table>
```

```
time
```

```
#bytes
```

```
<table>
<thead>
<tr>
<th>time</th>
<th>#bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
</tr>
<tr>
<td>3</td>
<td>300</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
</tr>
<tr>
<td>5</td>
<td>500</td>
</tr>
</tbody>
</table>
```
Solution #3: traffic

- disk traces (80-20 ‘law’ = ‘multifractal’)
Solution #3: traffic

Clarification:

- **fractal**: a set of points that is self-similar
- **multifractal**: a probability density function that is self-similar

Many other time-sequences are bursty/clustered: (such as?)
Tape accesses

# tapes needed, to retrieve $n$ records?
(# days down, due to failures / hurricanes / communication noise...)

Tape#1 ↔ Tape# N

---

<table>
<thead>
<tr>
<th>Tape#1</th>
<th>Tape# N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

time
Tape accesses

# tapes retrieved

Tape#1  Tape# N

time

50-50 = Poisson

# qual. records
Road map

- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More **tools** and examples
- Discussion - putting fractals to work!
- Conclusions – practitioner’s guide
- Appendix: gory details - boxcounting plots
More tools

- Zipf’s law
- Korcak’s law / “fat fractals”
A famous power law: Zipf’s law

• Q: vocabulary word frequency in a document - any pattern?
A famous power law: Zipf’s law

- Bible - rank vs frequency (log-log)

**log(freq)**

```
log(rank)
```

```
log(freq)
```

```
log(rank)
```
A famous power law: Zipf’s law

- Bible - rank vs frequency (log-log)
- similarly, in many other languages; for customers and sales volume; city populations etc etc
A famous power law: Zipf’s law

- Zipf distr: \( \text{freq} = \frac{1}{\text{rank}} \)
- Generalized Zipf: \( \text{freq} = \frac{1}{(\text{rank})^a} \)

\[ \log(\text{freq}) \quad \text{BIBLE rank-freq. plot} \]

\[ \log(\text{freq}) \quad \log(\text{rank}) \]
Olympic medals (Sidney):

\[ y = -0.9676x + 2.3054 \]

\[ R^2 = 0.9458 \]
More power laws: areas – Korcak’s law

Scandinavian lakes

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

\[ \log(\text{count}( \geq \text{area})) \]

Scandinavian lakes area vs complementary cumulative count (log-log axes)
More power laws: Korcak

Japan islands
More power laws: Korcak

Japan islands; area vs cumulative count (log-log axes)

$\log(\text{count( } \geq \text{ area}))$

$log(\text{area})$

$log(\text{area})$
(Korcak’s law: Aegean islands)
Korcak’s law & “fat fractals”

How to generate such regions?
Korca̧k’s law & “fat fractals”

Q: How to generate such regions?
A: recursively, from a single region
so far we’ve seen:

- **concepts:**
  - fractals, multifractals and fat fractals

- **tools:**
  - correlation integral (= pair-count plot)
  - rank/frequency plot (Zipf’s law)
  - CCDF (Korcak’s law)
Road map

- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More tools and examples
- Discussion - putting fractals to work!
- Conclusions – practitioner’s guide
- Appendix: gory details - boxcounting plots
Other applications: Internet

- How does the internet look like?

CMU
Other applications: Internet

- How does the internet look like?
- Internet routers: how many neighbors within $h$ hops?
(reminder: our tool-box:)

- **concepts:**
  - fractals, multifractals and fat fractals

- **tools:**
  - correlation integral (= pair-count plot)
  - rank/frequency plot (Zipf’s law)
  - CCDF (Korcak’s law)
Internet topology

- Internet routers: how many neighbors within $h$ hops?

Reachability function: number of neighbors within $r$ hops, vs $r$ (log-log).

Mbone routers, 1995
More power laws on the Internet

degree vs rank, for Internet domains (log-log) [sigcomm99]
More power laws - internet

- pdf of degrees: (slope: 2.2)

Log(count) vs. Log(degree) graph with a slope of -2.2.
Even more power laws on the Internet

\[ \log(\text{i-th eigenvalue}) \]

Scree plot for Internet domains (log-log) [sigcomm99]

0.47
More apps: Brain scans

- Oct-trees; brain-scans

\[
\log(#\text{octants}) = 2.63 = \text{fd}
\]
More apps: Medical images

[Burdett et al, SPIE ‘93]:
- benign tumors: $fd \sim 2.37$
- malignant: $fd \sim 2.56$
More fractals:

- cardiovascular system: 3 (!)
- stock prices (LYCOS) - random walks: 1.5

Coastlines: 1.2-1.58 (Norway!)
More power laws

- duration of UNIX jobs [Harchol-Balter]
- Energy of earthquakes (Gutenberg-Richter law) [simscience.org]
Even more power laws:

- publication counts (Lotka’s law)
- Distribution of UNIX file sizes
- Income distribution (Pareto’s law)
- web hit counts [Huberman]
Power laws, cont’ed

- In- and out-degree distribution of web sites [Barabasi], [IBM-CLEVER]
- Length of file transfers [Bestavros+]
- Click-stream data (w/ A. Montgomery (CMU-GSIA) + MediaMetrix)
Road map

- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More examples and tools
- Discussion - putting fractals to work!
- Conclusions – practitioner’s guide
- Appendix: gory details - boxcounting plots
Settings for fractals:

Points; areas (-> fat fractals), eg:
Settings for fractals:

Points; areas, eg:

- cities/stores/hospitals, over earth’s surface
- time-stamps of events (customer arrivals, packet losses, criminal actions) over time
- regions (sales areas, islands, patches of habitats) over space
Settings for fractals:

- customer feature vectors (age, income, frequency of visits, amount of sales per visit)
Some uses of fractals:

- Detect non-existence of rules (if points are uniform)
- Detect non-homogeneous regions (eg., legal login time-stamps may have different fd than intruders’)
- Estimate number of neighbors / customers / competitors within a radius
Multi-Fractals

Setting: points or objects, w/ some value, eg:

- cities w/ populations
- positions on earth and amount of gold/water/oil underneath
- product ids and sales per product
- people and their salaries
- months and count of accidents
Use of multifractals:

- Estimate tape/disk accesses
  - how many of the 100 tapes contain my 50 phonecall records?
  - how many days without an accident?
Use of multifractals

- how often do we exceed the threshold?

#bytes

Poisson

time
Use of multifractals cont’d

- Extrapolations for/from samples
Use of multifractals cont’d

- How many distinct products account for 90% of the sales?
  - 20% 
  - 80%

![Graph showing sales distribution](image)
Road map

- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More examples and tools
- Discussion - putting fractals to work!
- Conclusions – practitioner’s guide
- Appendix: gory details - boxcounting plots
Conclusions

- **Real data often disobey textbook assumptions** (Gaussian, Poisson, uniformity, independence)
  - avoid ‘mean’ - use median, or even better, use:
  - fractals, self-similarity, and power laws, to find patterns - specifically:
Conclusions

- **tool#1**: (for points) ‘correlation integral’: (# pairs within <= r) vs (distance r)

- **tool#2**: (for categorical values) rank-frequency plot (a’la Zipf)

- **tool#3**: (for numerical values) CCDF: Complementary cumulative distr. function (# of elements with value >= a)
Practitioner’s guide:

**tool#1**: #pairs vs distance, for a set of objects, with a distance function (slope = intrinsic dimensionality)

\[
\log(#\text{pairs}) = \log(hops) + 2.8 \\
\log(#\text{pairs(within} \leqslant r)) = \log(r) + 1.51
\]
Practitioner’s guide:

- **tool#2**: rank-frequency plot (for **categorical attributes**)

![Graphs showing log(rank) vs. log(degree) for internet domains and Bible data.]
Practitioner’s guide:

**tool #3**: CCDF, for (skewed) **numerical attributes**, eg. areas of islands/lakes, UNIX jobs…

\[ \log(\text{count}( \geq \text{area}) \) \]

---

Scandinavian lakes
Books

- Strongly recommended intro book:
  - Manfred Schroeder *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise*

- Classic book on fractals:
References


References


- [vldb96] Christos Faloutsos, Yossi Matias and Avi Silberschatz, *Modeling Skewed Distributions Using Multifractals and the \`80-20 Law\’* Conf. on Very Large Data Bases (VLDB), Bombay, India, Sept. 1996.
References


References

- [icde99] Guido Proietti and Christos Faloutsos, *I/O complexity for range queries on region data stored using an R-tree*, International Conference on Data Engineering (ICDE), Sydney, Australia, March 23-26, 1999

Appendix - Gory details

- Bad news: There are more than one fractal dimensions
  - Minkowski fd; Hausdorff fd; Correlation fd; Information fd

- Great news:
  - they can all be computed fast!
  - they usually have nearby values
Fast estimation of fd(s):

- How, for the (correlation) fractal dimension?

Box-counting plot:

\[ \log(\text{sum}(\pi^2)) \]

\[ \log(\text{r}) \]
Definitions

- \( p_i \): the percentage (or count) of points in the \( i \)-th cell
- \( r \): the side of the grid
Fast estimation of $fd(s)$:

- compute $\sum (\pi^2)$ for another grid side, $r'$

$\log(\sum(\pi^2))$

$\log(r)$
Fast estimation of $fd(s)$:

etc; if the resulting plot has a linear part, its slope is the correlation fractal dimension $D_2$.
Definitions (cont’d)

Many more fractal dimensions $D_q$ (related to Renyi entropies):

$$D_q = \frac{1}{q - 1} \left( \frac{\partial \log(\sum p_i^q)}{\partial \log(r)} \right)$$

for $q \neq 1$

$$D_1 = \frac{\partial \sum p_i \log(p_i)}{\partial \log(r)}$$
Hausdorff or box-counting fd:

- **Box counting plot:** $\log(N(r))$ vs $\log(r)$

- $r$: grid side

- $N(r)$: count of non-empty cells

- (Hausdorff) fractal dimension $D_0$:

$$D_0 = -\frac{\partial \log(N(r))}{\partial \log(r)}$$
Definitions (cont’d)

- Hausdorff fd:

\[ r \sim \log(\text{#non-empty cells}) \]
Observations

- $q=0$: Hausdorff fractal dimension
- $q=2$: Correlation fractal dimension (identical to the exponent of the number of neighbors vs radius)
- $q=1$: Information fractal dimension
in general, the $D_q$’s take similar, but not identical, values.

except for perfectly self-similar point-sets, where $D_q = D_{q'}$ for any $q$, $q'$
Examples: MG county

- Montgomery County of MD (road endpoints)
Examples: LB county

- Long Beach county of CA (road endpoints)
Conclusions

- many fractal dimensions, with nearby values
- can be computed quickly
  \(O(N)\) or \(O(N \log(N))\)
- (code: on the web)