Carnegie Mellon

15-826: Multimedia (Databases) and Data Mining

Lecture #8: Fractals - introduction

C. Faloutsos

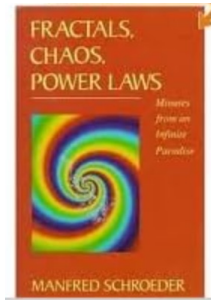
Must-read Material

- Christos Faloutsos and Ibrahim Kamel, *Beyond Uniformity and Independence: Analysis of R-trees Using the Concept of Fractal Dimension*, Proc. ACM SIGACT-SIGMOD-SIGART PODS, May 1994, pp. 4-13, Minneapolis, MN.

Recommended Material

optional, but **very** useful:

- Manfred Schroeder *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise* W.H. Freeman and Company, 1991
  - Chapter 10: boxcounting method
  - Chapter 1: Sierpinski triangle



Outline

Goal: 'Find similar / interesting things'

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

Indexing - Detailed outline

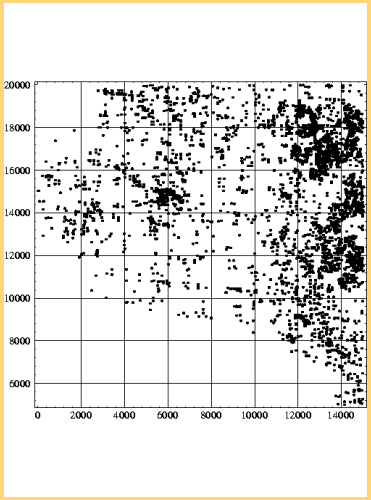
- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
  - z-ordering
  - R-trees
  - misc
- fractals
  - intro
  - applications
- text

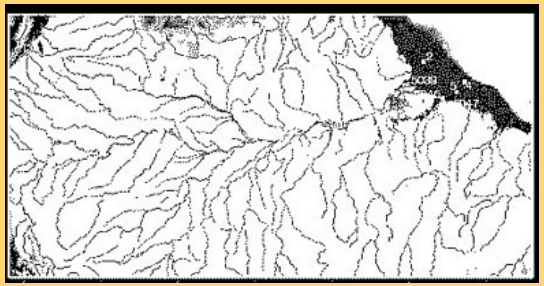
Intro to fractals - outline

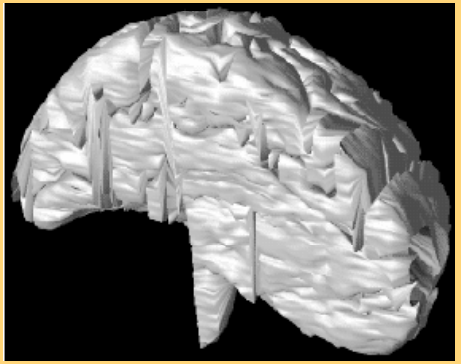
- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More examples and tools
- Discussion - putting fractals to work!
- Conclusions – practitioner's guide
- Appendix: gory details - boxcounting plots

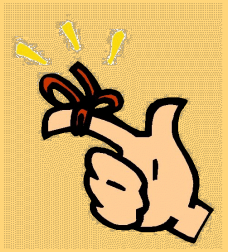
Problem

- What patterns are in real *k*-dim points?



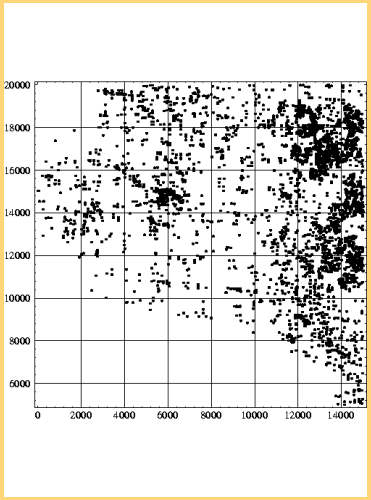


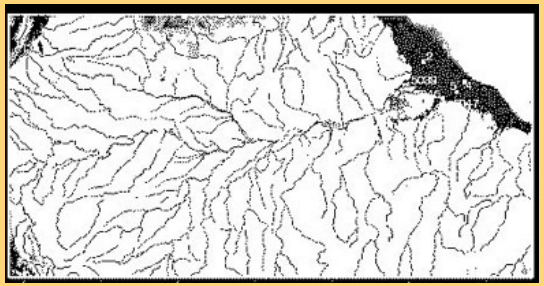


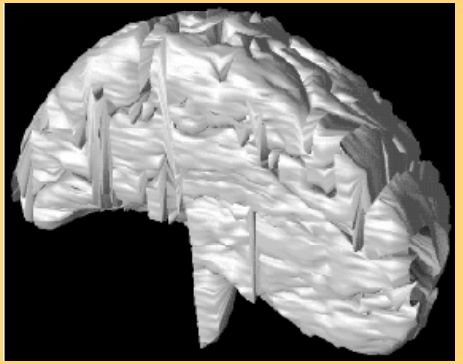


Conclusions

- What patterns are in real *k*-dim points?
- Self-similarity (= fractals -> power laws)

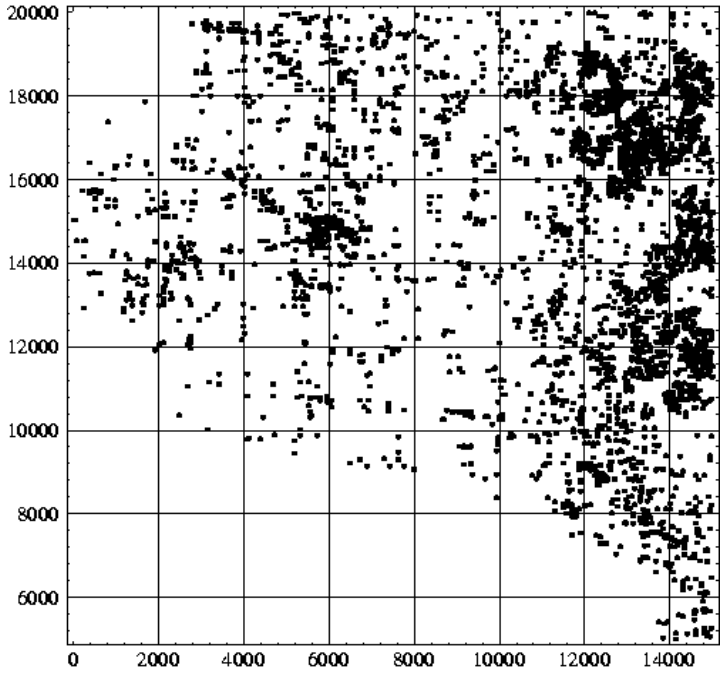






Carnegie Mellon

Problem #1: GIS - points

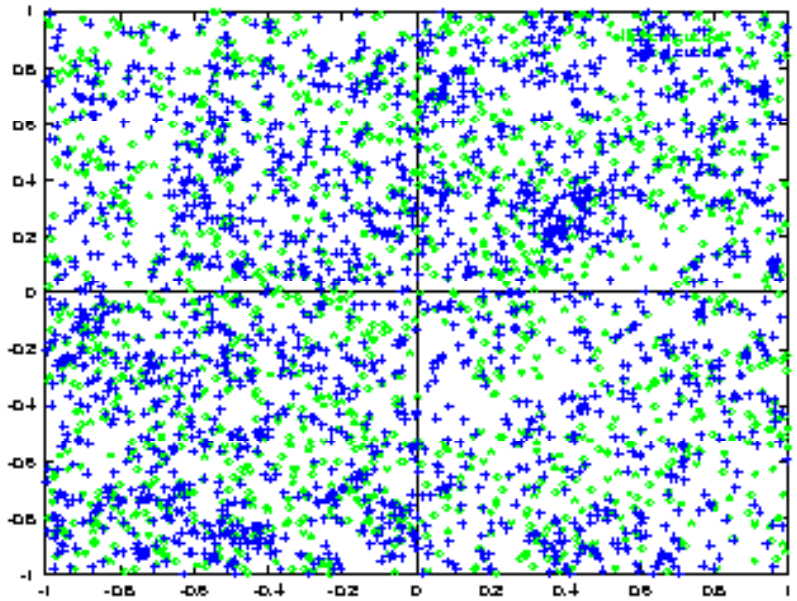


Road end-points of Montgomery county:

- 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 w/ B. Nichol)



- 'spiral' and 'elliptical' galaxies

  (stores and households ...)

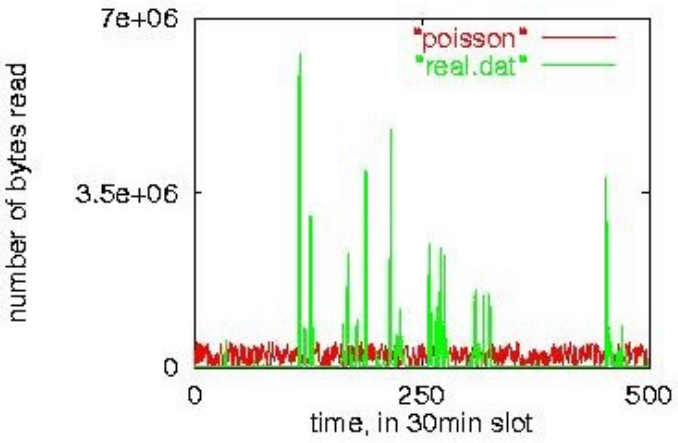
- patterns?

- attraction/repulsion?

- how many 'spi' within r from an 'ell' ?

Problem #3: traffic

 # bytes

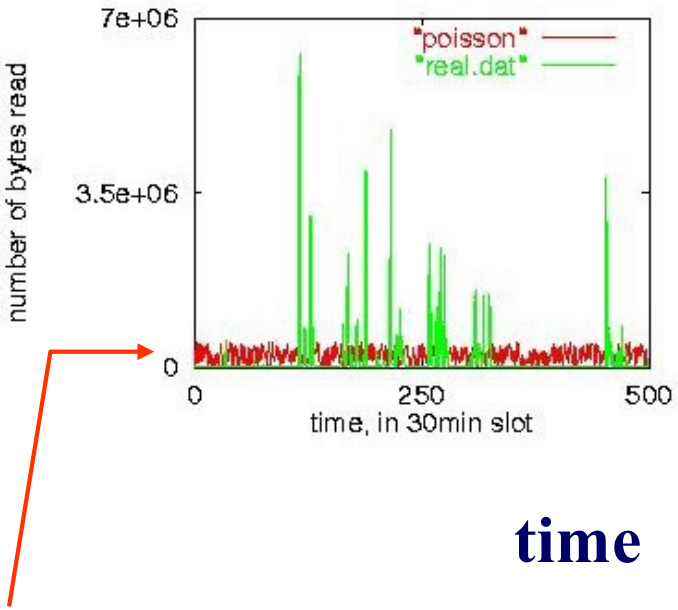


time

- disk trace (from HP - J. Wilkes); Web traffic - fit a model
- how many explosions to expect?
- queue length distr.?

Problem #3: traffic

 # bytes

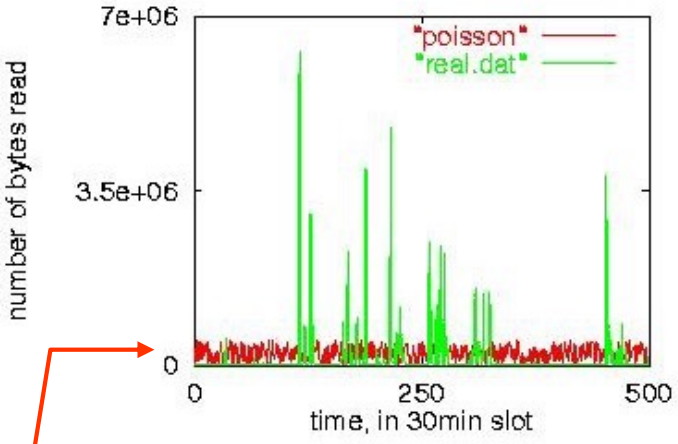


time

Poisson
indep.,
ident. distr

Problem #3: traffic

 # bytes



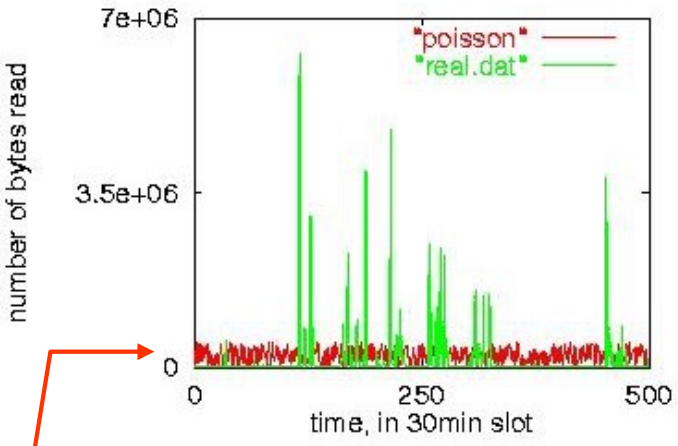
time

~~Poisson~~

indep., ident. distr

Problem #3: traffic

 # bytes



time

~~**Poisson**~~
~~**indep.,**~~
ident. distr

Q: Then, how to generate such bursty traffic?

Common answer:

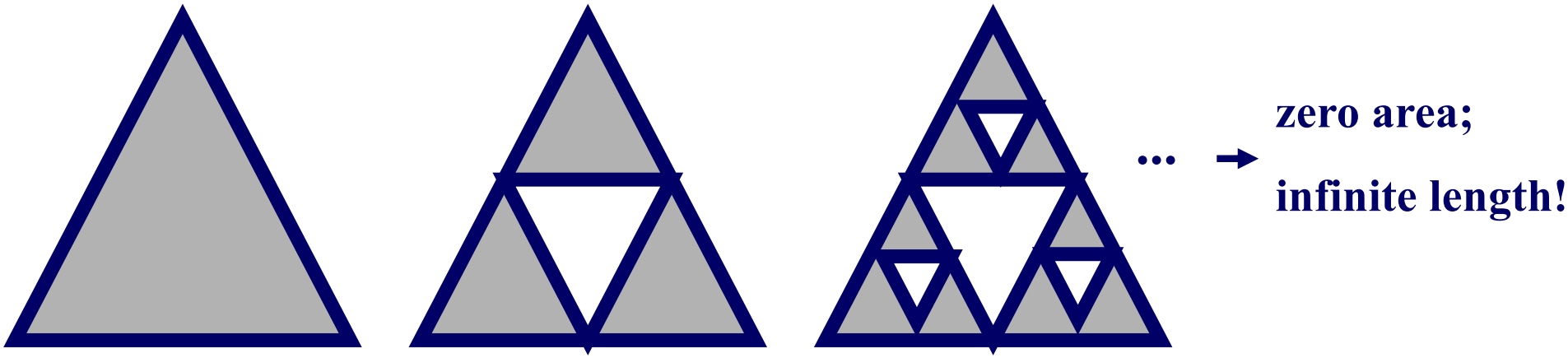
- Fractals / self-similarities / power laws
- Seminal works from Hilbert, Minkowski, Cantor, Mandelbrot, (Hausdorff, Lyapunov, Ken Wilson, …)

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:



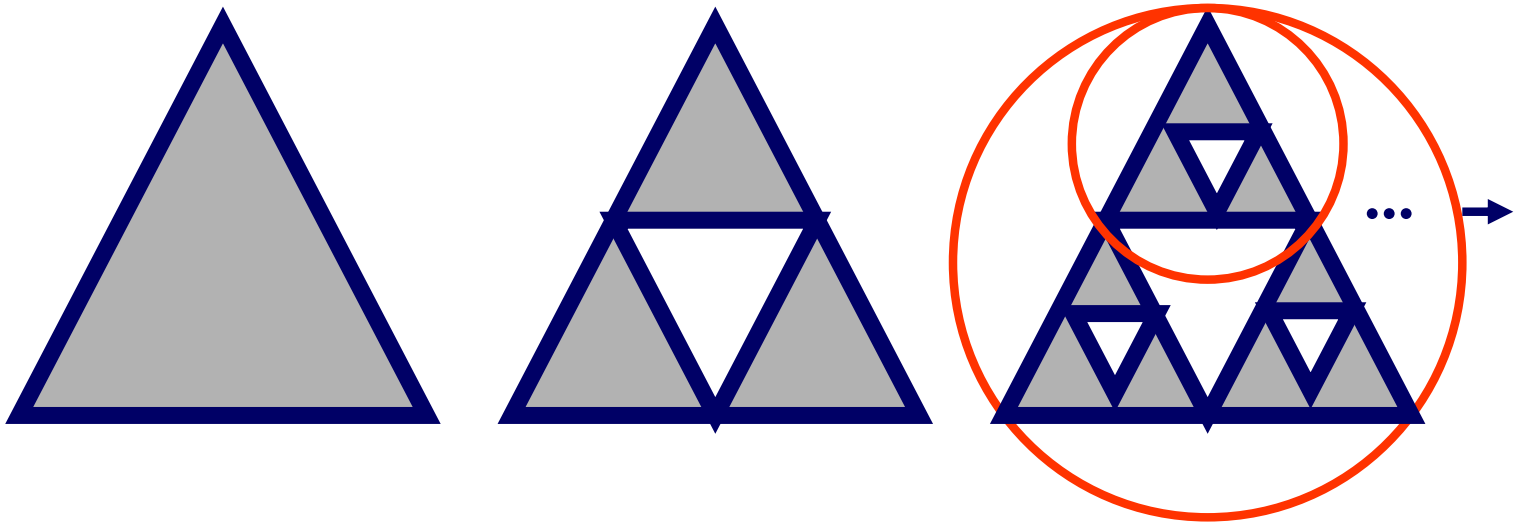
Dimensionality??

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:

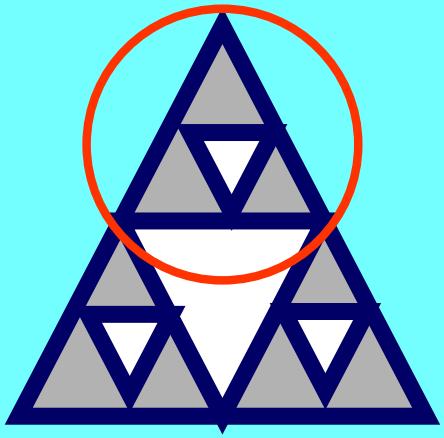


=log(n)/log(f) = log(3)/log(2) = 1.58

Definitions of f.d.

For mathematical fractal

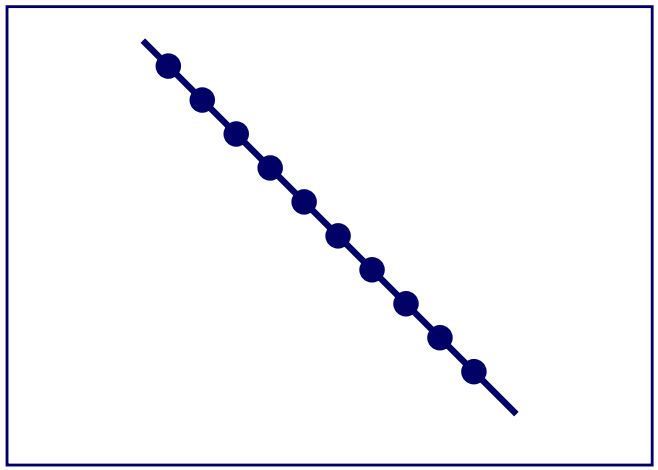
$$\text{fd} = \frac{\log(n)}{\log(f)}$$



For real set of points:

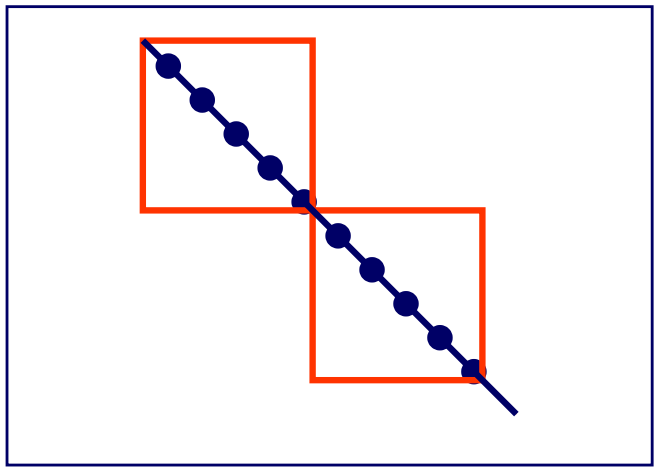
Intrinsic ('fractal') dimension

- Q: fractal dimension of a line?
- A: 1 (= log(2)/log(2)!)



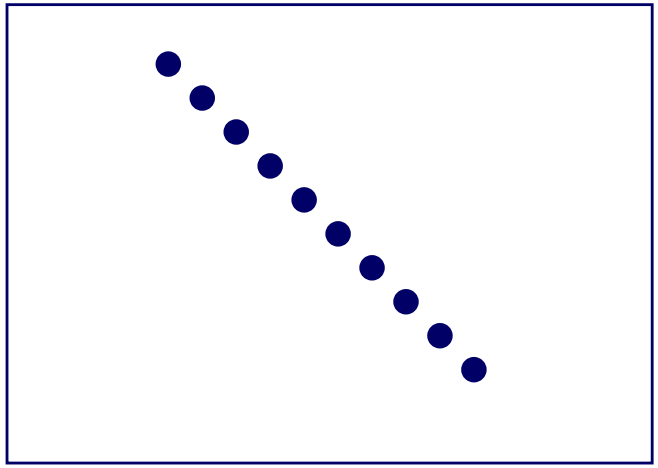
Intrinsic ('fractal') dimension

- Q: fractal dimension of a line?
- A: 1 (= log(2)/log(2)!)



Intrinsic ('fractal') dimension

- Q: dfn for a finite set of points?



x	y
5	1
4	2
3	3
2	4

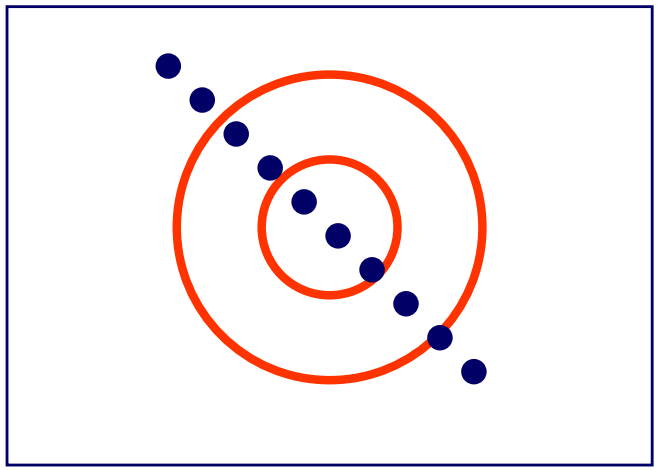
Intrinsic ('fractal') dimension

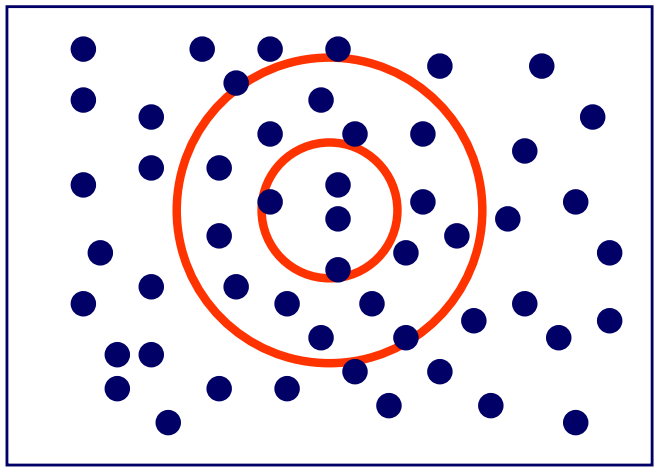
- Q: fractal dimension of a line?
- A: nn (<= r) ~ r^1

('power law': y=x^a)

- Q: fd of a plane?
- A: nn (<= r) ~ r^2

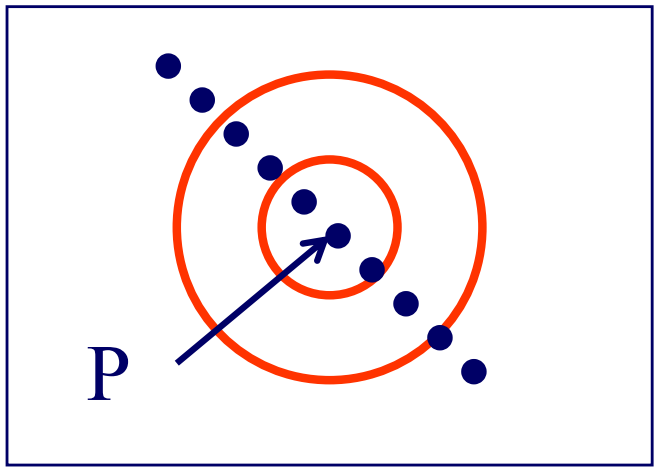
fd== slope of (log(nn) vs log(r))





Intrinsic ('fractal') dimension

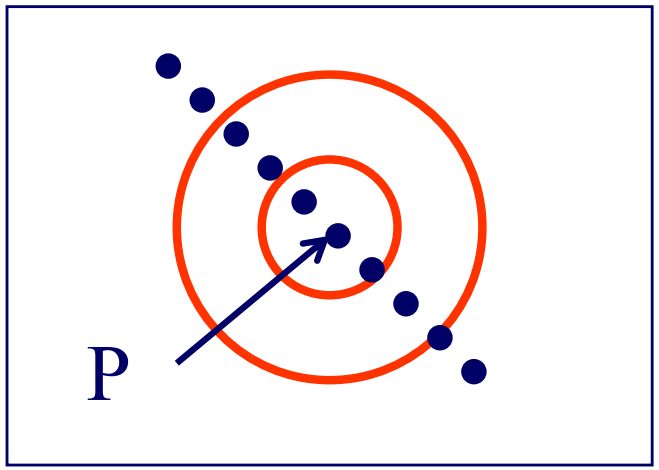
- **Local** fractal dimension of point 'P'?
- A: nn_P (<= r) ~ r^1



- If this equation holds for several values of r,
- Then, the **local fractal dimension** of point P:
- Local fd = exp = 1

Intrinsic ('fractal') dimension

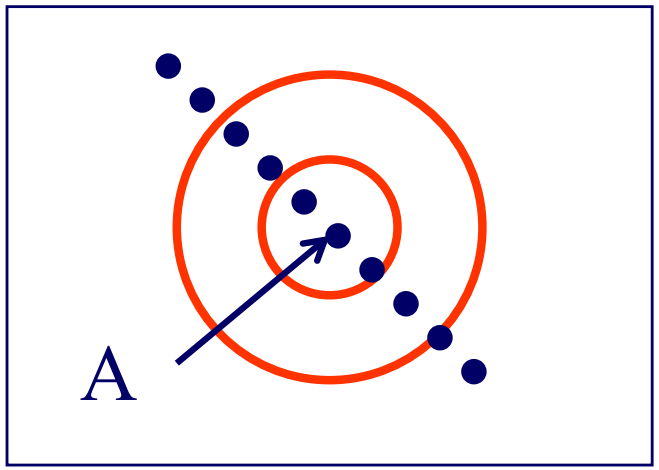
- **Local** fractal dimension of point 'A'?
- A: nn_P (<= r) ~ r^1



- If this is true for all points of the cloud
- Then the exponent is the **global** f.d.
- Or simply the f.d.

Intrinsic ('fractal') dimension

- **Global** fractal dimension?
- A: if
- sum_{all_P} [nn_P (<= r)] ~ r^1
- Then: exp = global f.d.

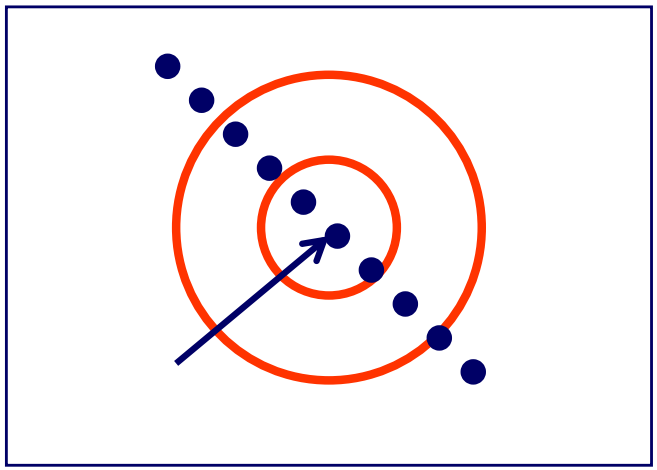


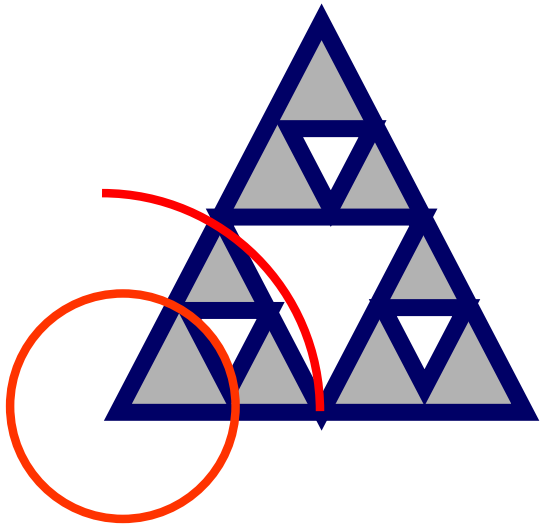
- If this is true for all points of the cloud
- Then the exponent is the **global** f.d.
- Or simply the f.d.

Intrinsic ('fractal') dimension

- **Local** fractal dimension

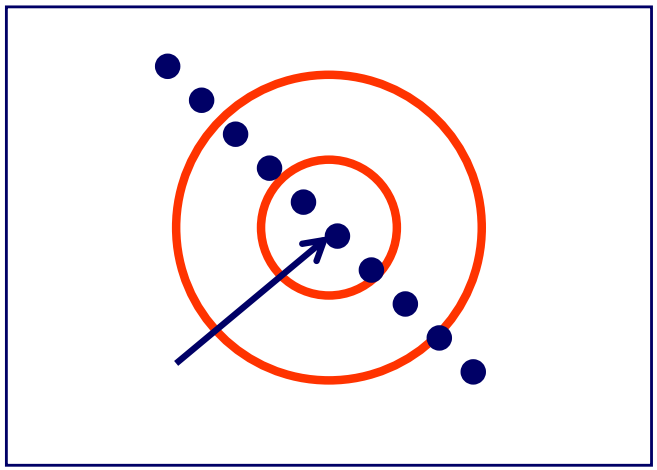
for sierpinski triangle ?



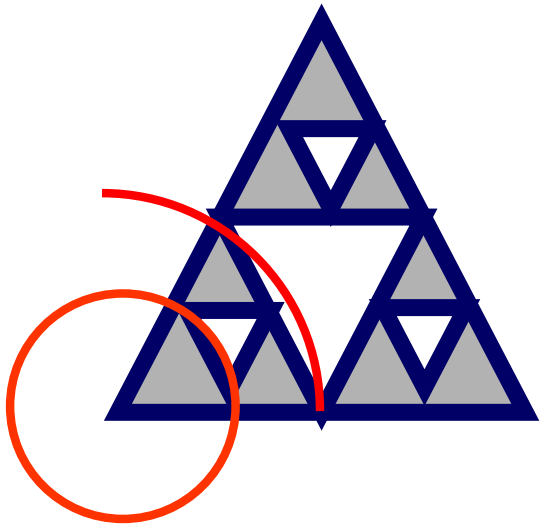


Intrinsic ('fractal') dimension

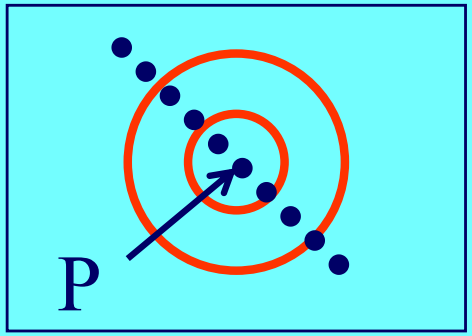
- **Local** fractal dimension for sierpinski triangle ?



- 2x radius, 3x points
- n = r ^ (log3/log2)

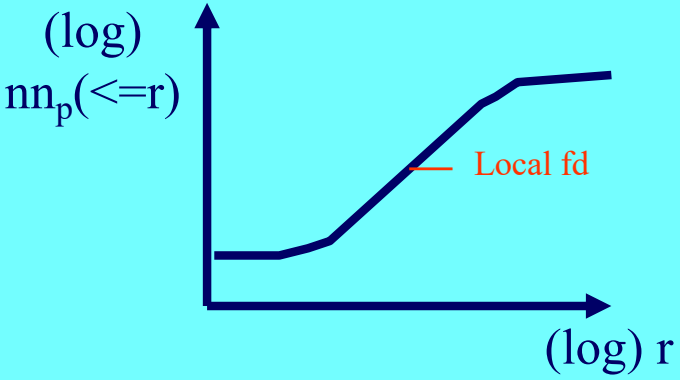


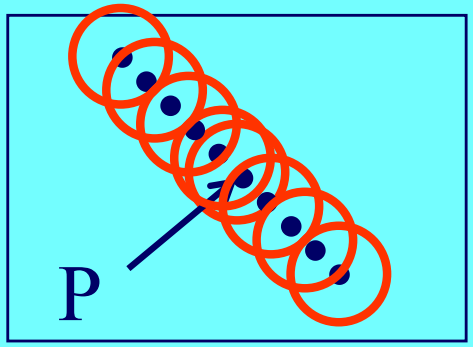
Local and global fd



neighbors of P within r

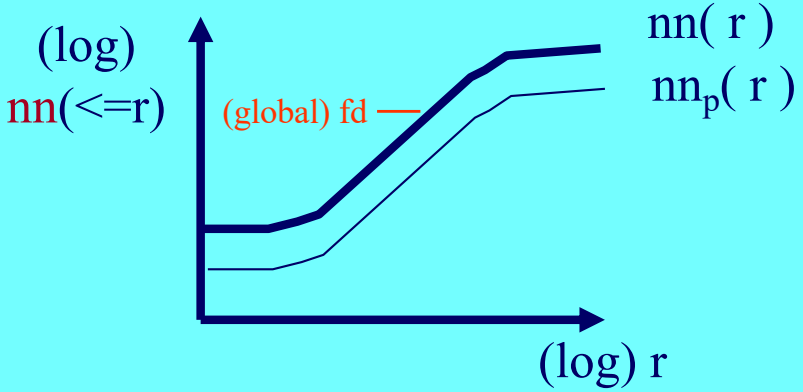
$$nn_P(\leq r) \propto r^{fd}$$





Sum_P(# neighbors within r) =
All pairs of neighbors within r =

$$nn(\leq r) \propto r^{fd}$$



Intrinsic ('fractal') dimension

- Algorithm, to estimate it?

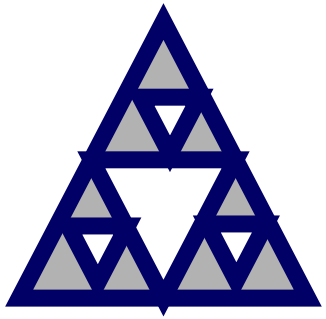
Notice

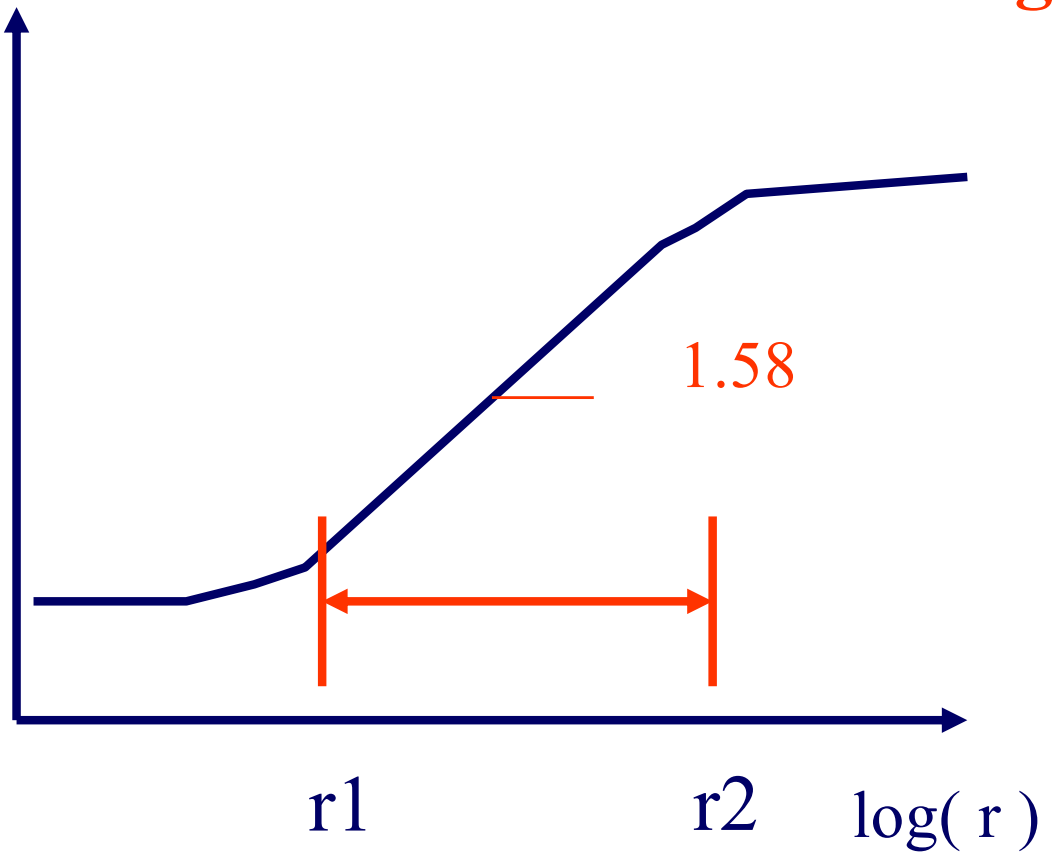
- Sum_{all_P} [nn_P (<=r)] is exactly *tot#pairs(<=r)*

including 'mirror' pairs

Sierpinsky triangle

== 'correlation integral'

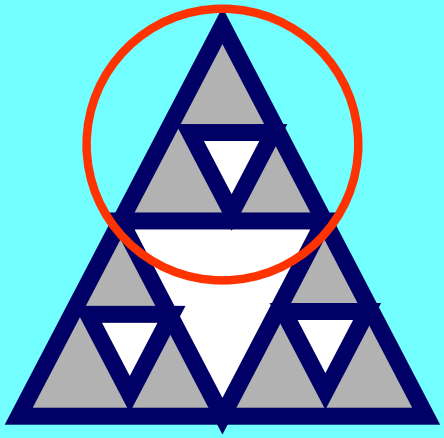




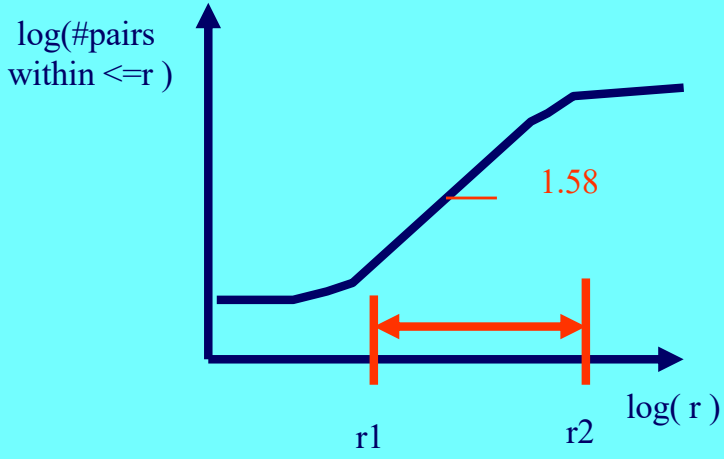
Definitions of f.d.

For mathematical fractal:

$$\text{fd} = \frac{\log(n)}{\log(f)}$$



For real set of points:
fd in the **range** (r1, r2):
Slope of corr. integral



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

Important properties

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

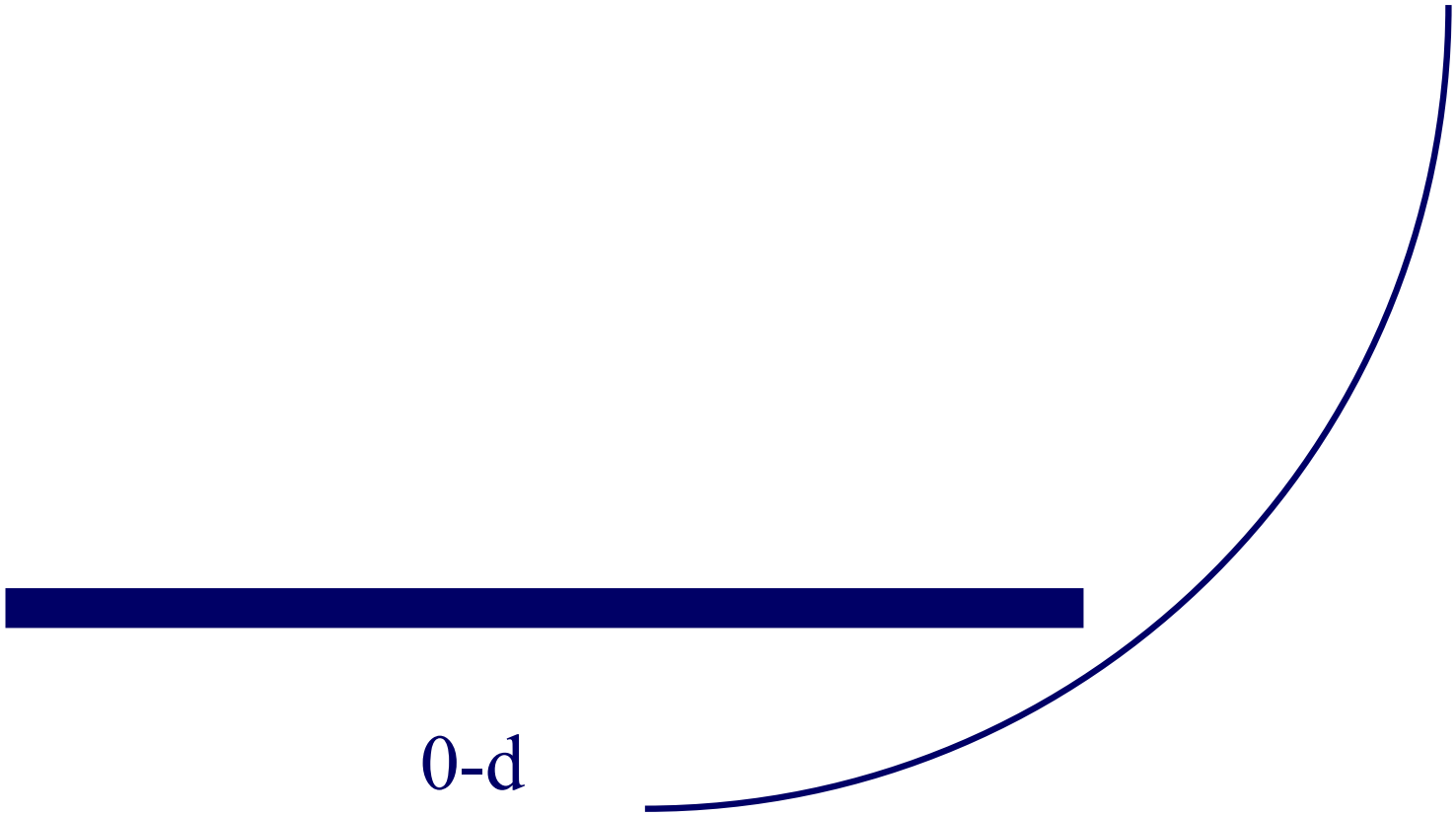
2-d

Important properties

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

1-d

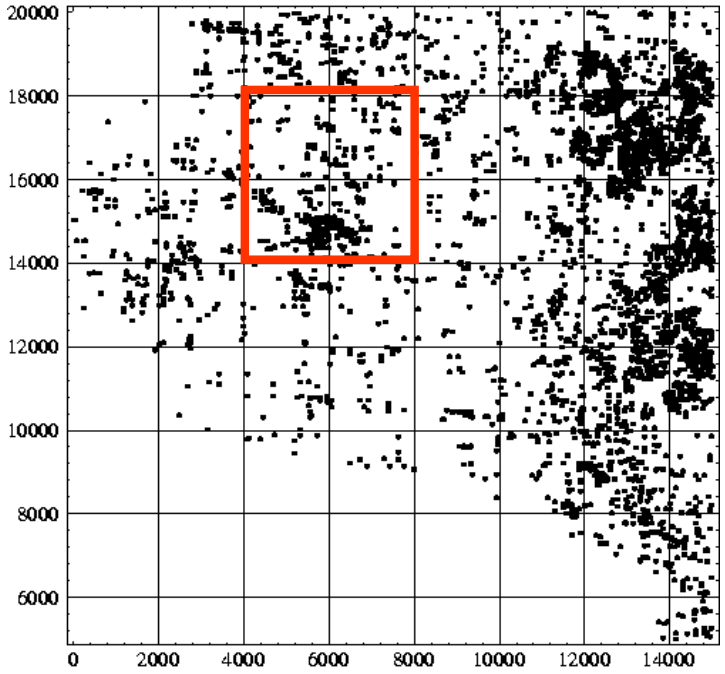
Important properties



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

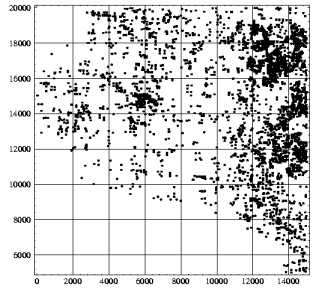
Problem #1: GIS points



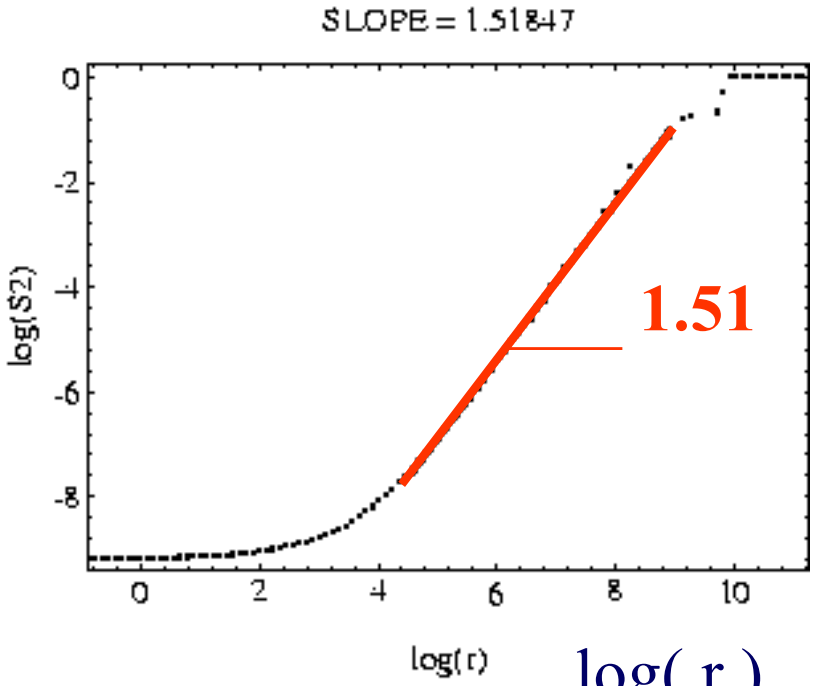
Cross-roads of Montgomery county:

- any rules?

Solution #1



log(#pairs(within <= r))

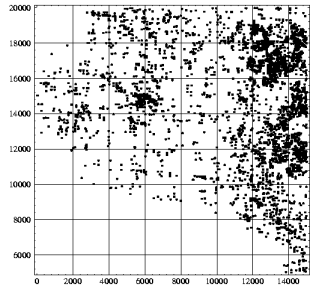


log(r)

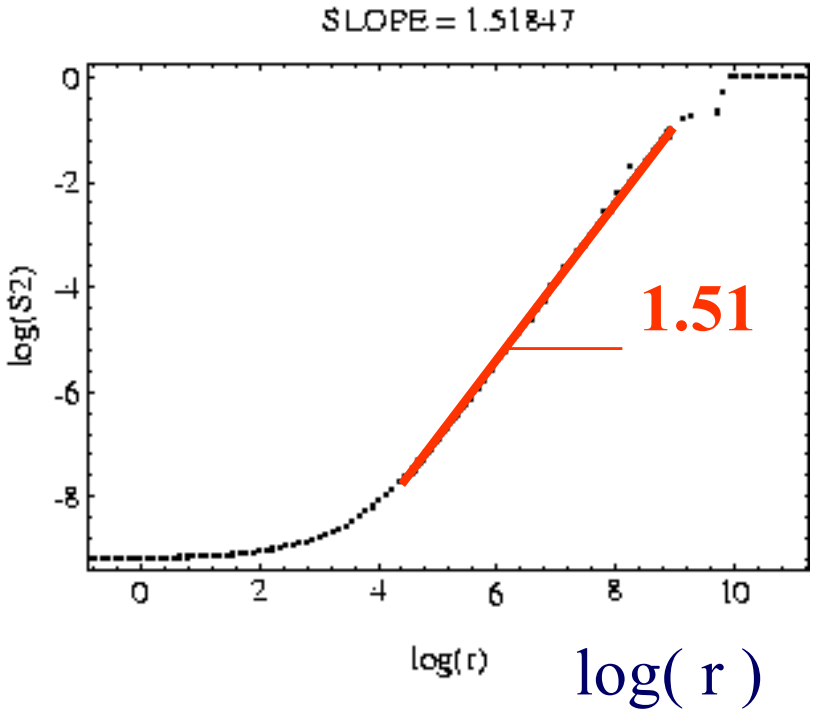
A: self-similarity ->

- <=> fractals
- <=> scale-free
- <=> power-laws (y=x^a, F=C*r^(-2))
- avg#neighbors(<= r) = r^D

Solution #1



log(#pairs(within <= r))



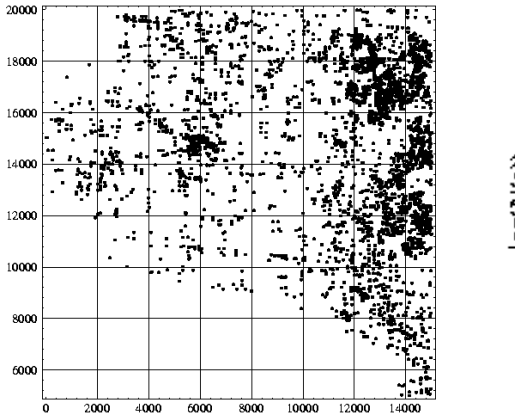
log(r)

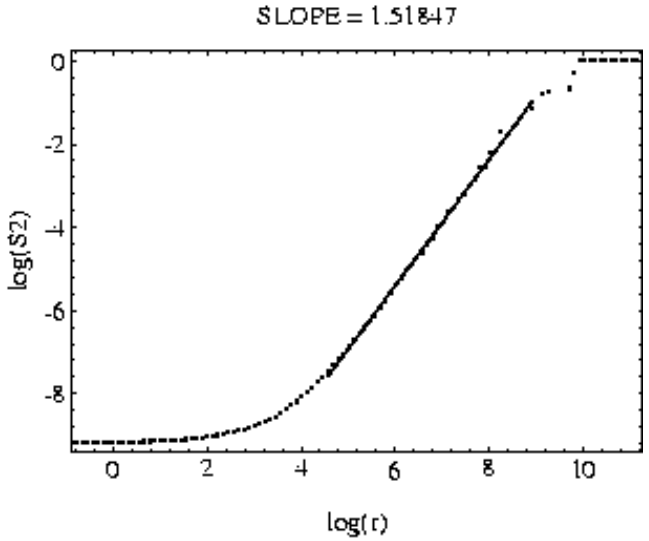
A: self-similarity

- avg#neighbors(<= r) ~ r^(1.51)

Examples:MG county

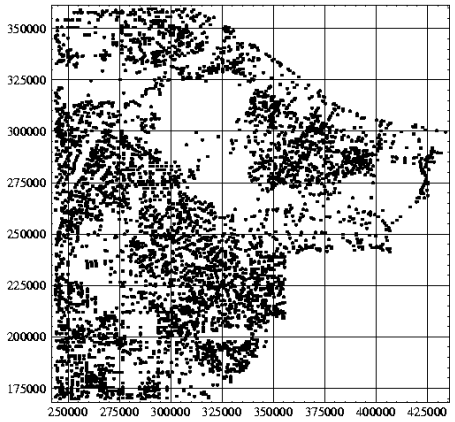
- Montgomery County of MD (road end-points)

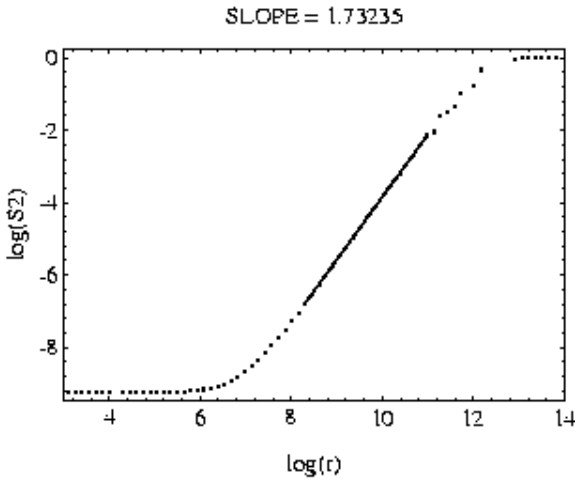




Examples:LB county

- Long Beach county of CA (road end-points)

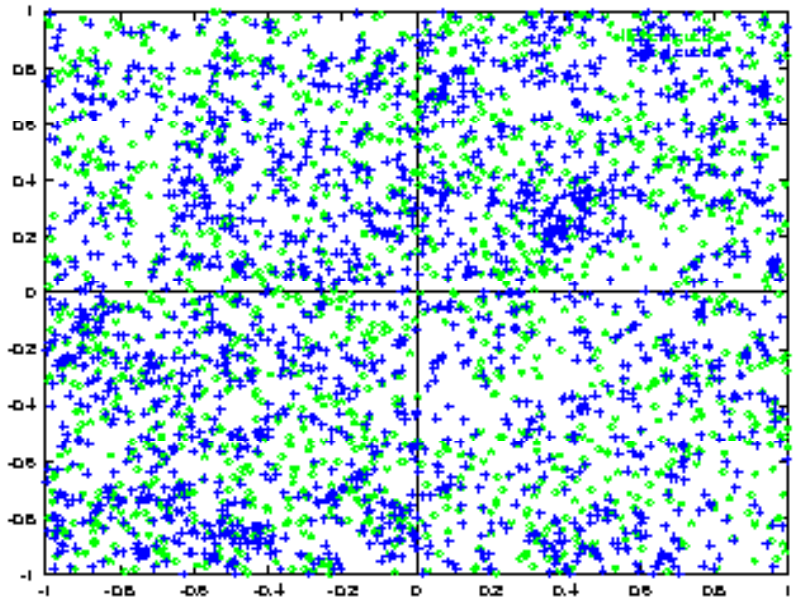




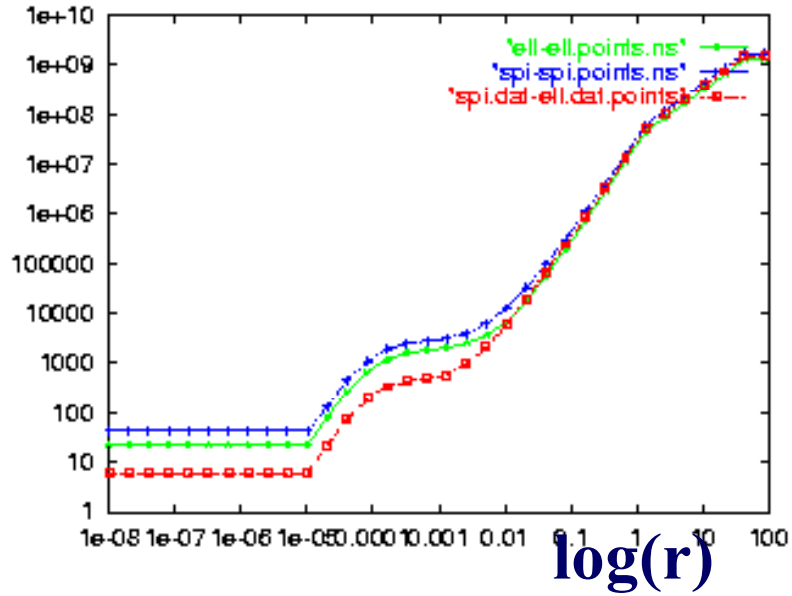
Carnegie Mellon

Solution#2: spatial d.m.

Galaxies ('BOPS' plot - [sigmod2000])



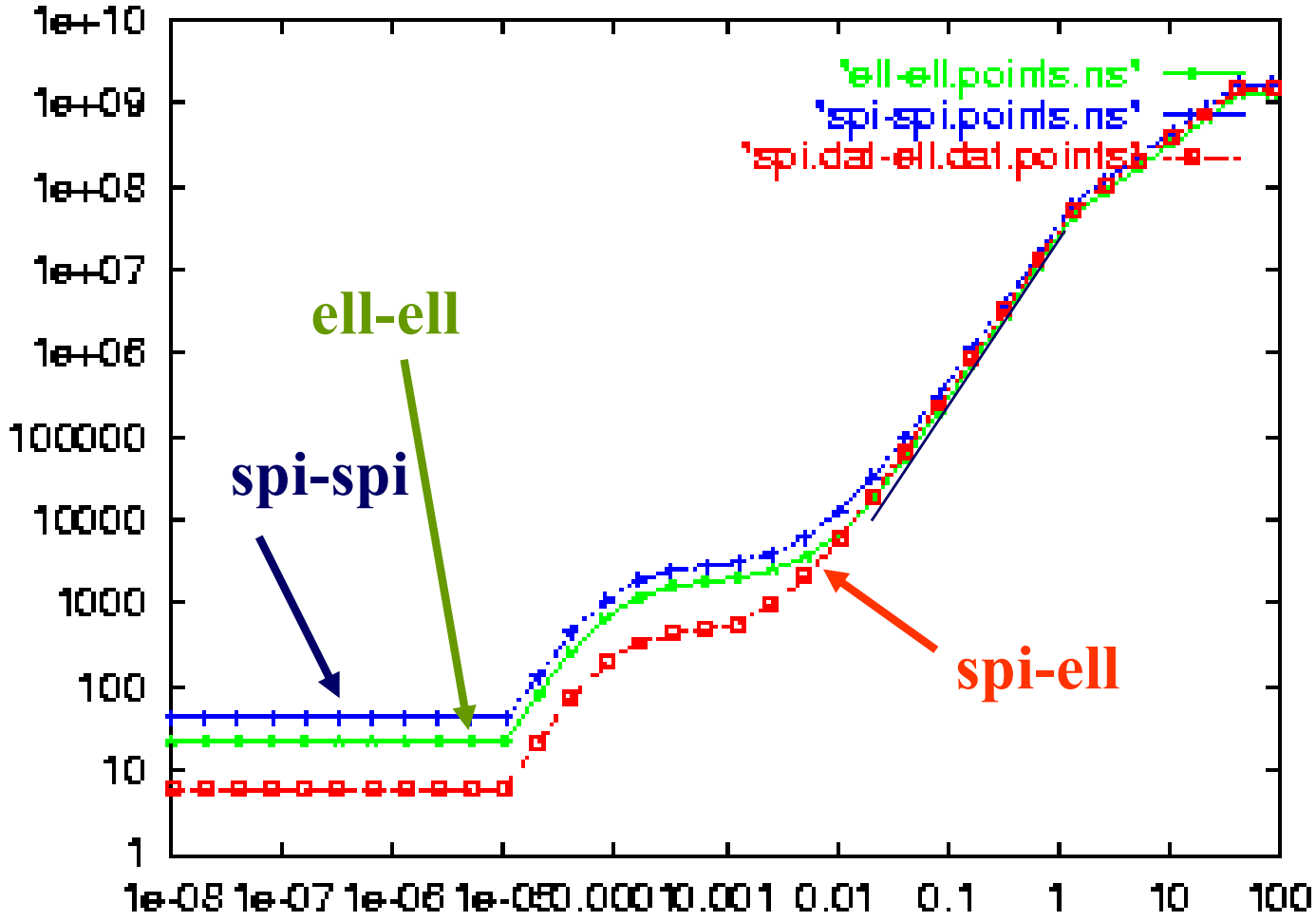
log(#pairs)



log(r)

Solution#2: spatial d.m.

log(#pairs within <=r)

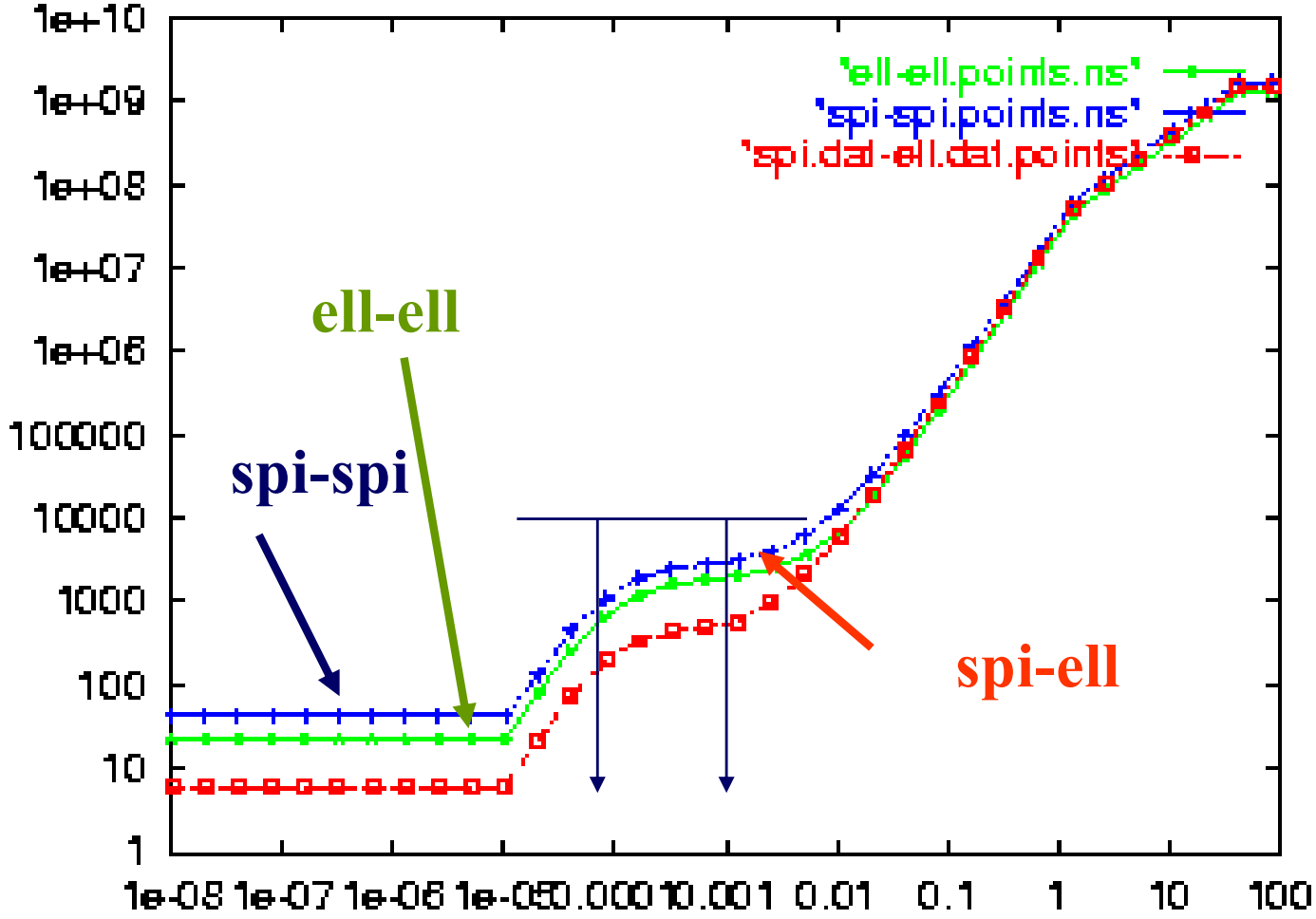


- 1.8 slope
- plateau!
- repulsion!

log(r)

Spatial d.m.

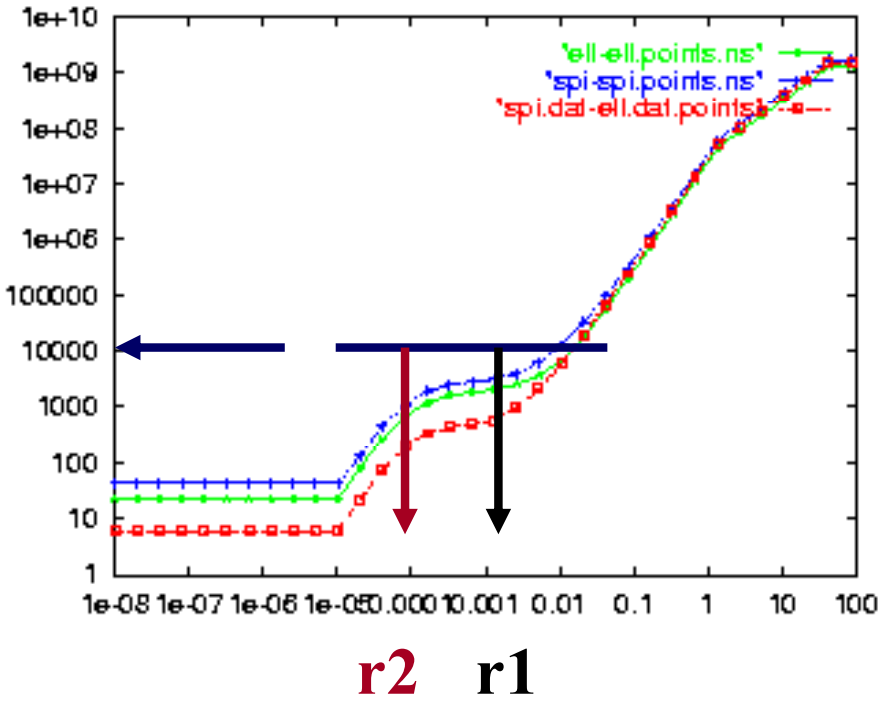
log(#pairs within <=r)

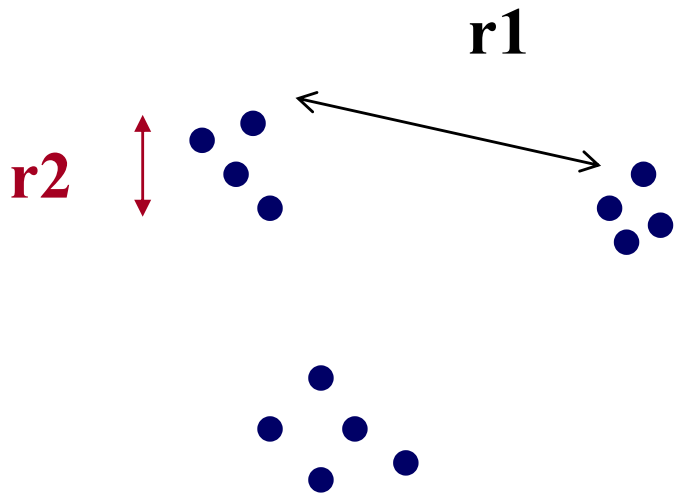


- 1.8 slope
- plateau!
- repulsion!

log(r)

Spatial d.m.

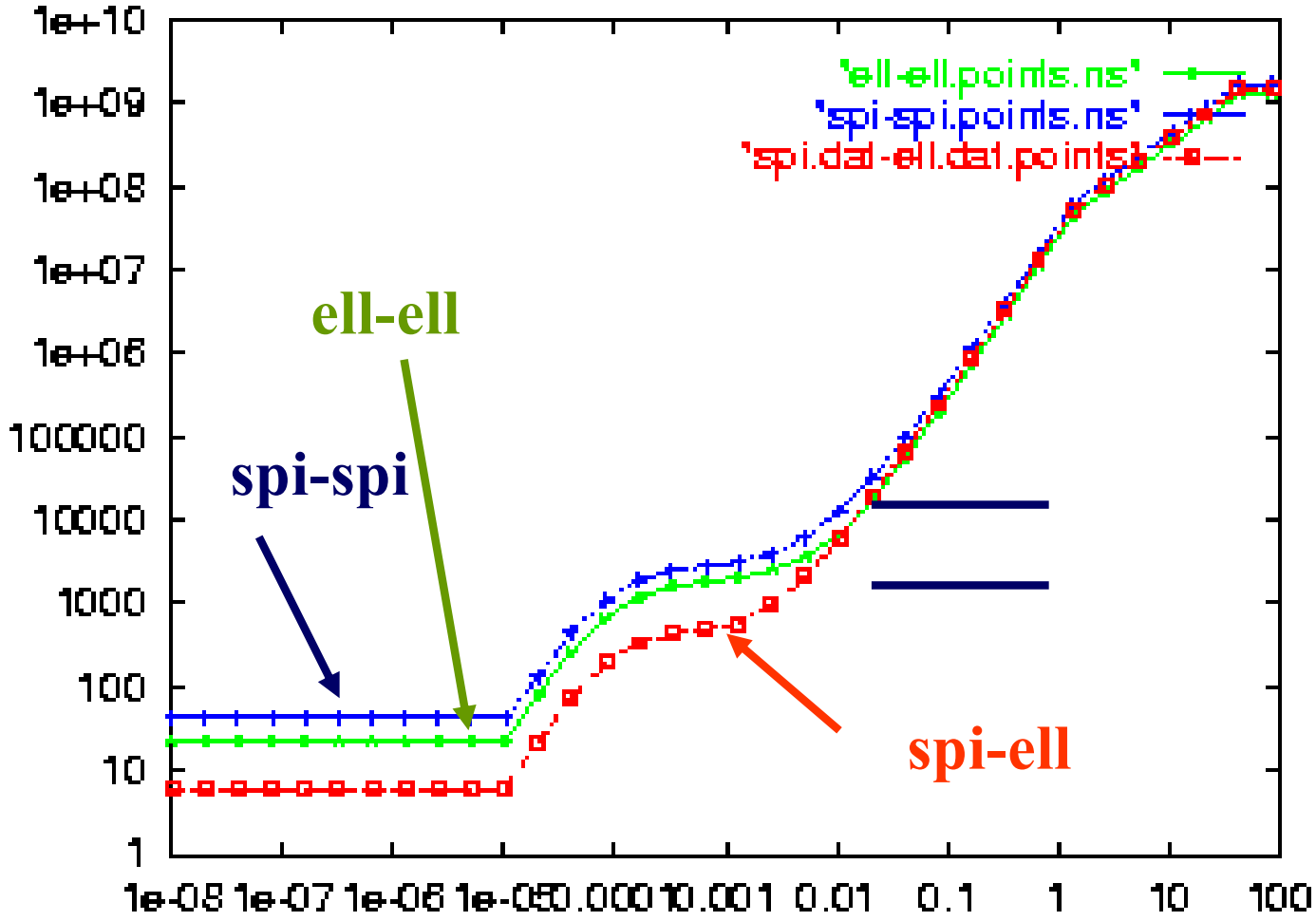




Heuristic on choosing # of clusters

Spatial d.m.

log(#pairs within <=r)

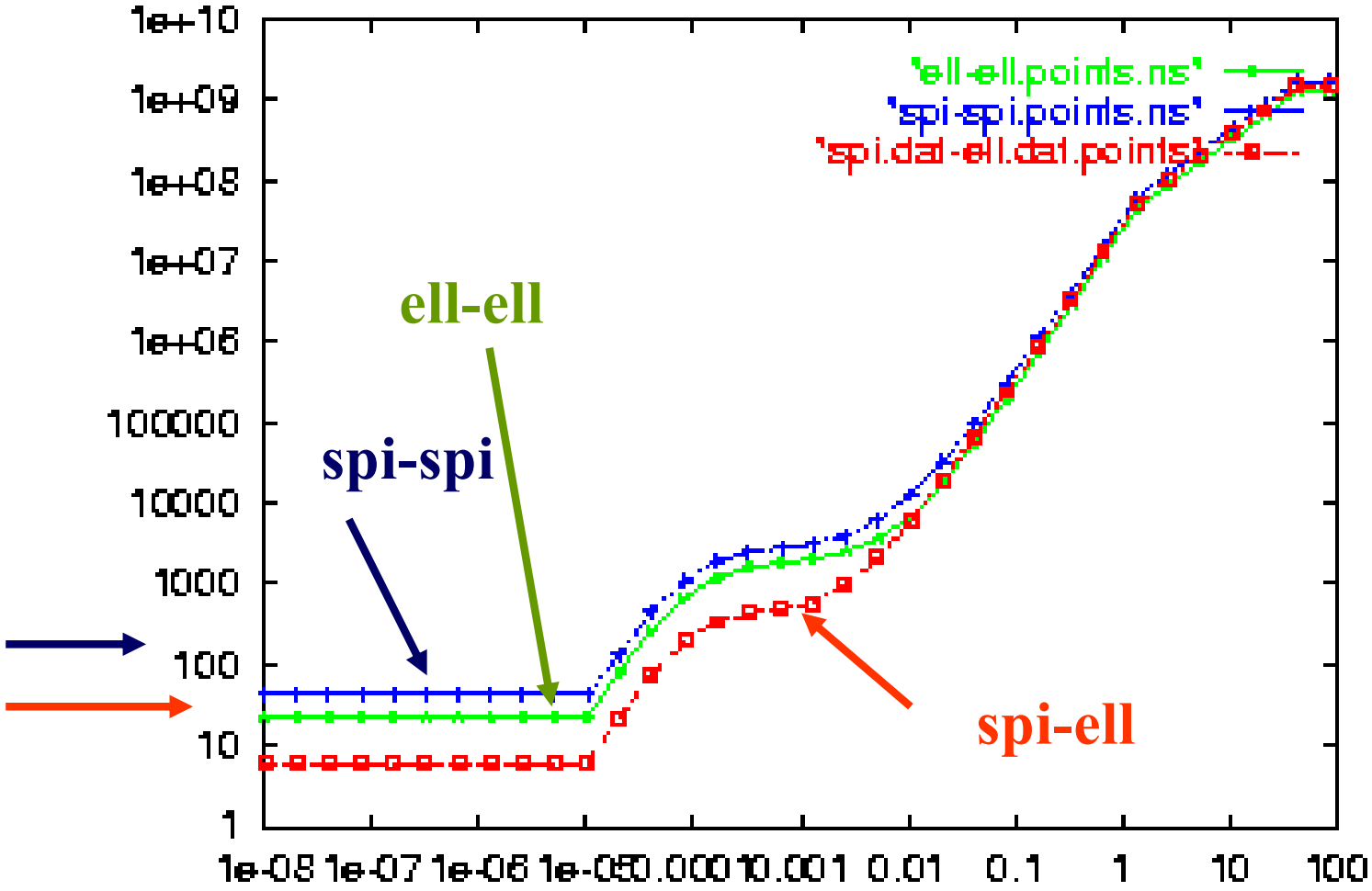


- 1.8 slope
- plateau!
- repulsion!

log(r)

Spatial d.m.

log(#pairs within <=r)



- 1.8 slope

- plateau!

-repulsion!!

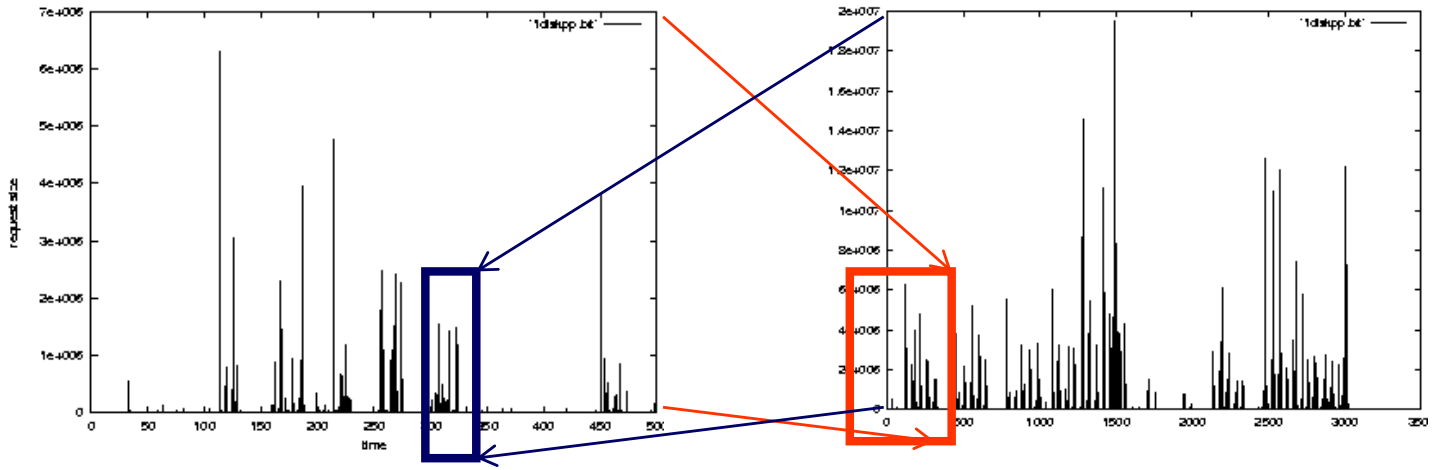
-duplicates

log(r)

Solution #3: traffic

- disk traces: self-similar:

#bytes

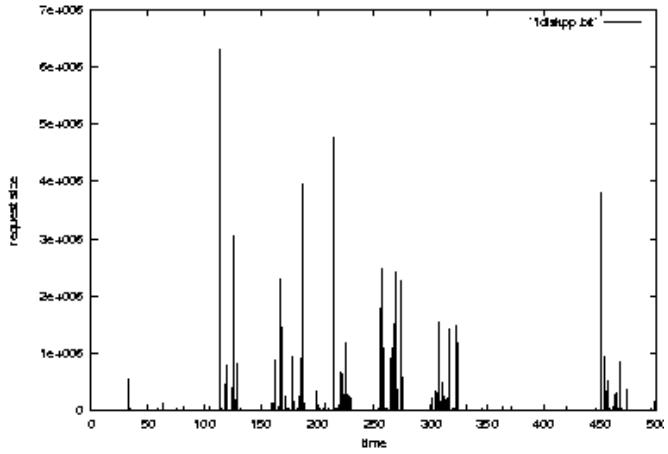


time

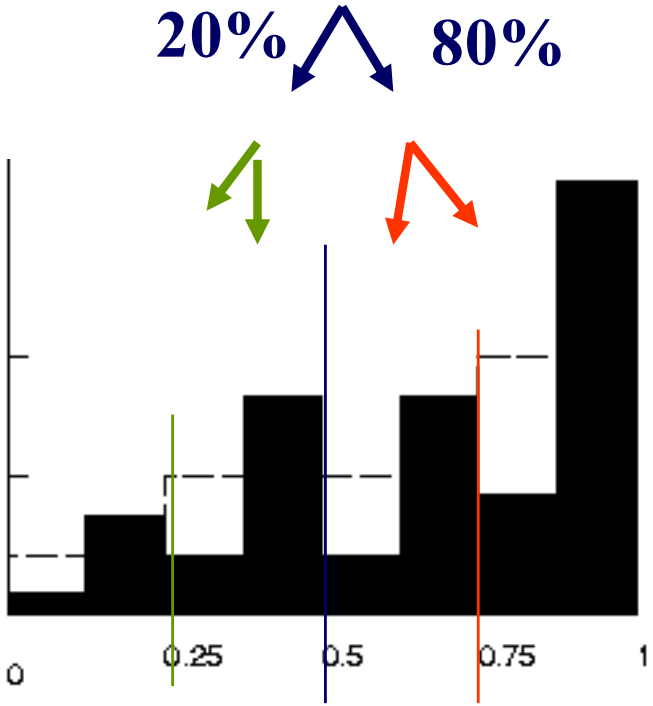
Solution #3: traffic

- disk traces (80-20 'law' = 'multifractal')

#bytes

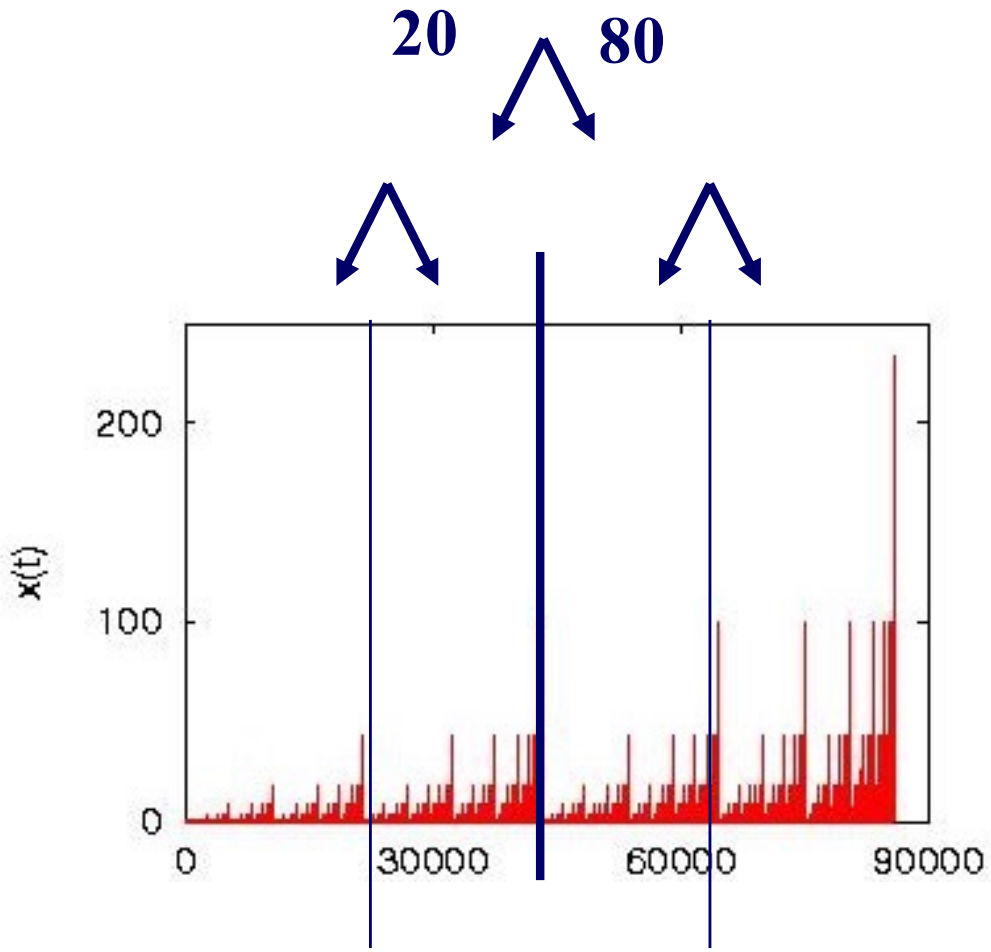


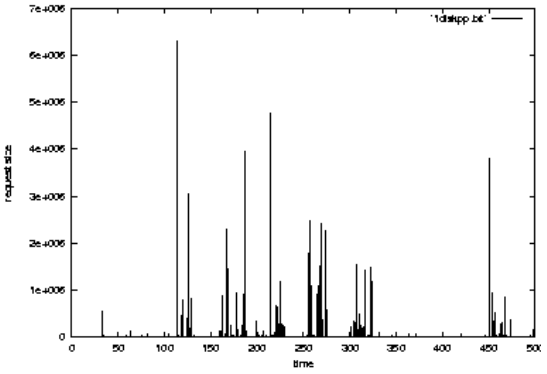
time

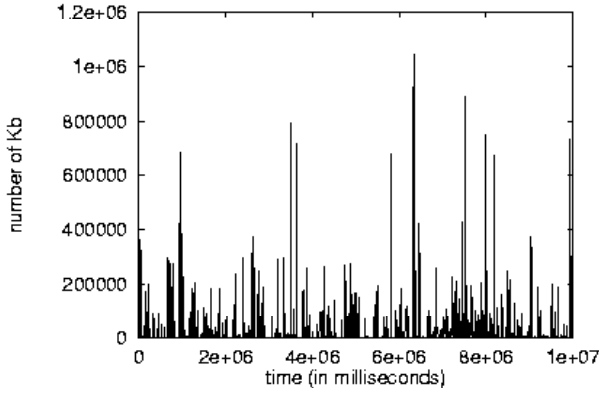


Carnegie Mellon

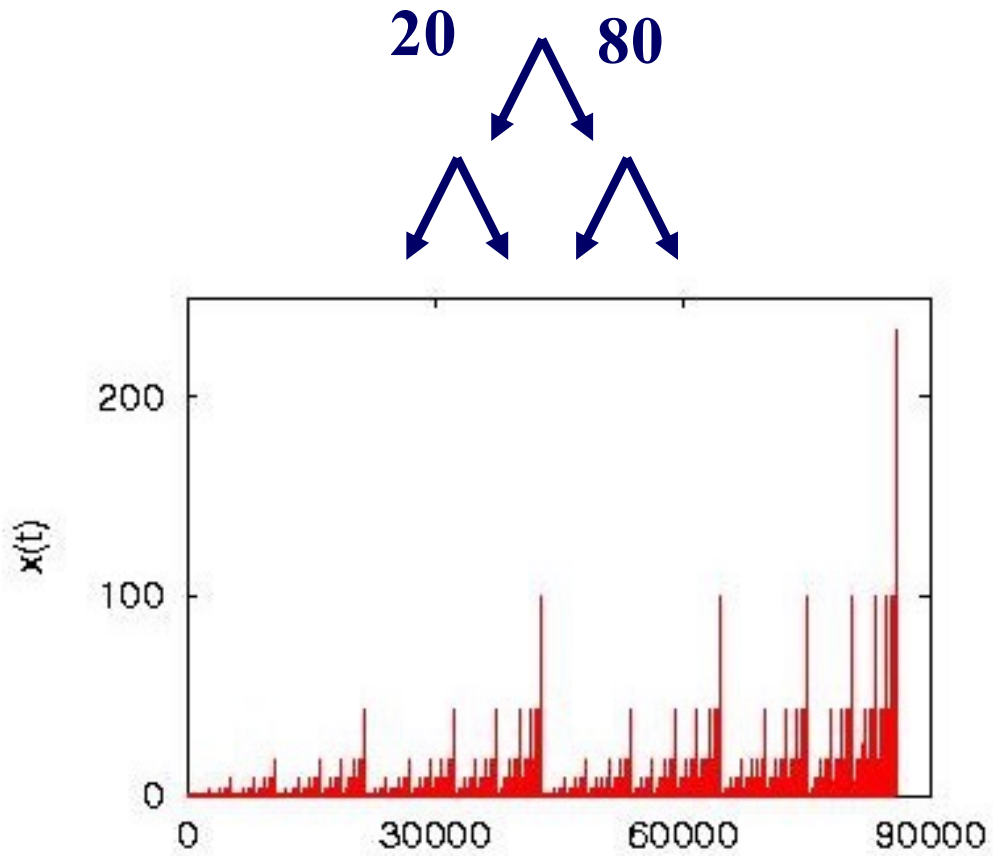
80-20 / multifractals







80-20 / multifractals



- p ; (1-p) in general
- **yes, there are dependencies**

Solution#3: traffic

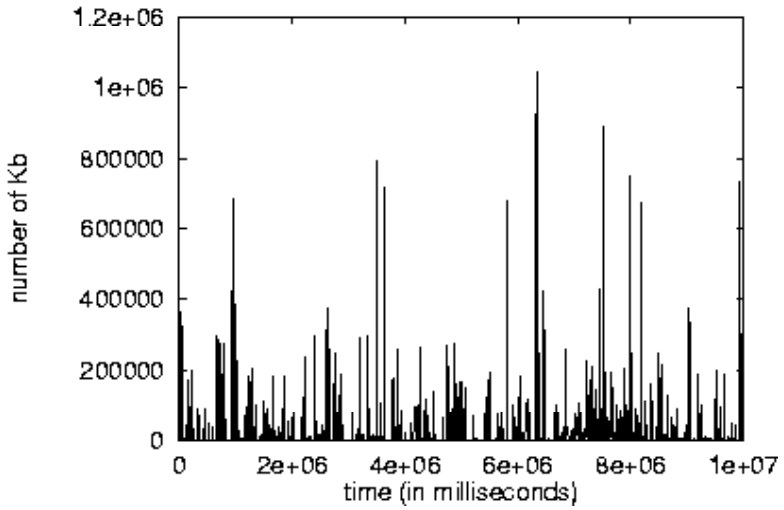
Clarification:

- 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?)

Example:

- network traffic

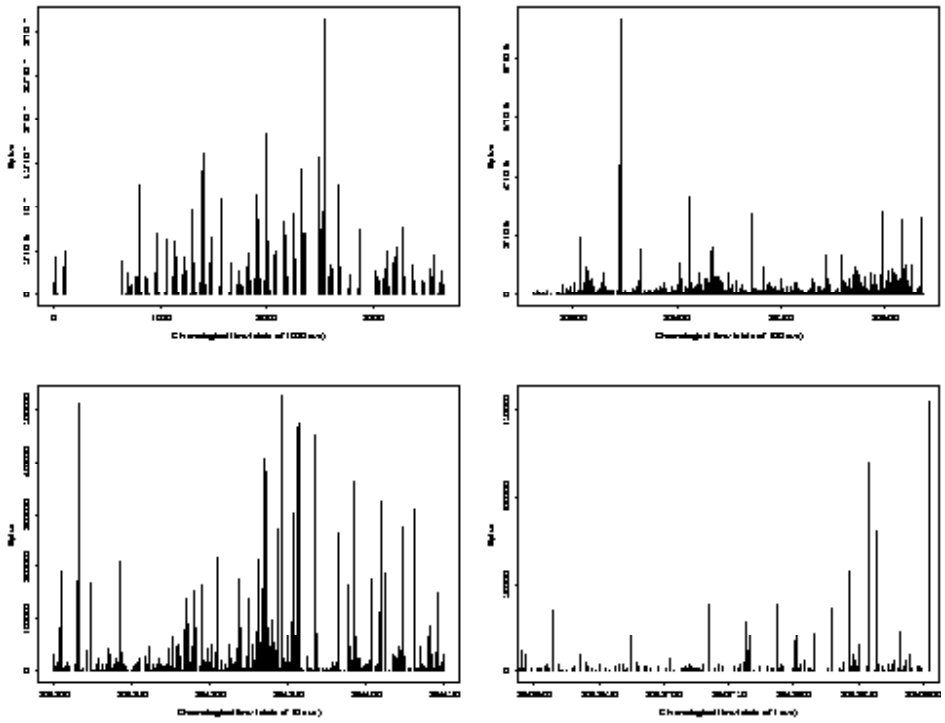


http://repository.cs.vt.edu/lbl-conn-7.tar.Z

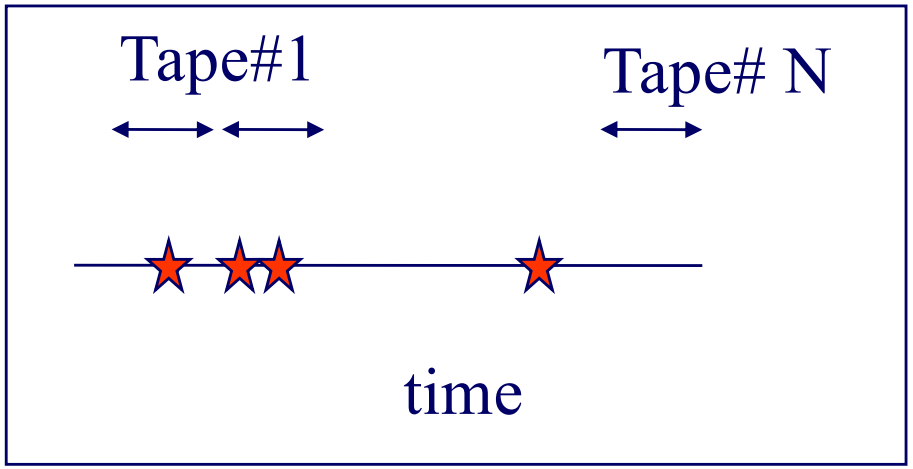
Web traffic

- [Crovella Bestavros, SIGMETRICS'96]

1000 sec; 100sec
10sec; 1sec



Tape accesses



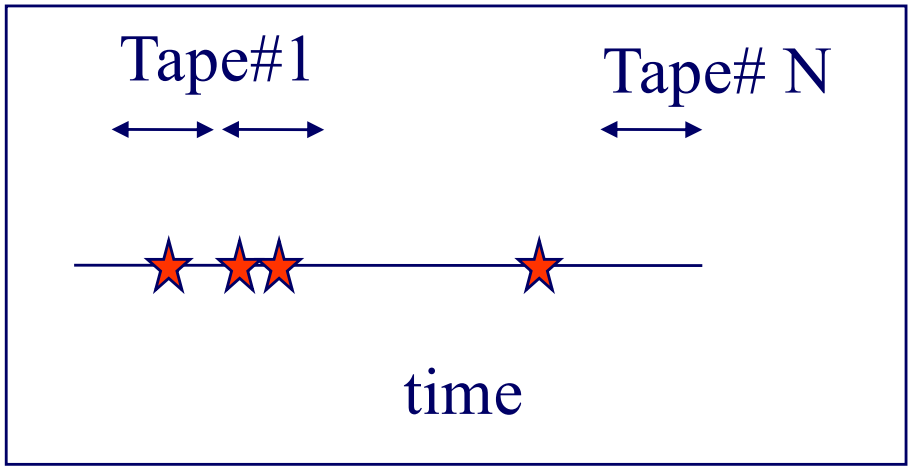
tapes needed, to retrieve *n* records?

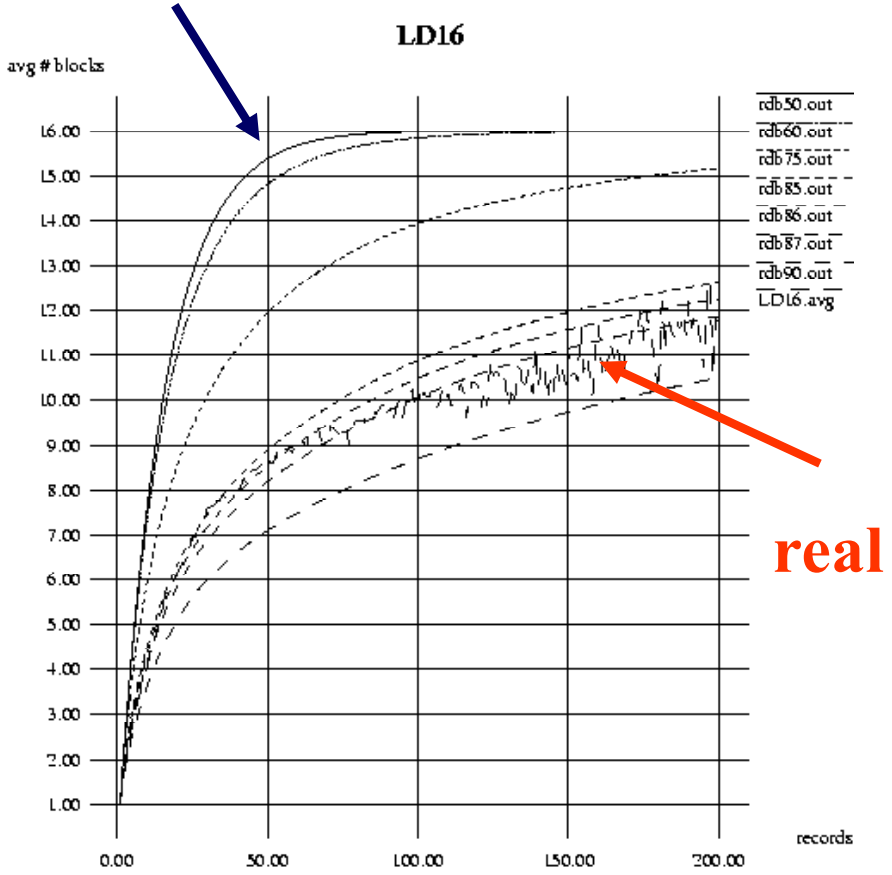
(# days down, due to failures / hurricanes / communication noise...)

Tape accesses

50-50 = Poisson

tapes retrieved





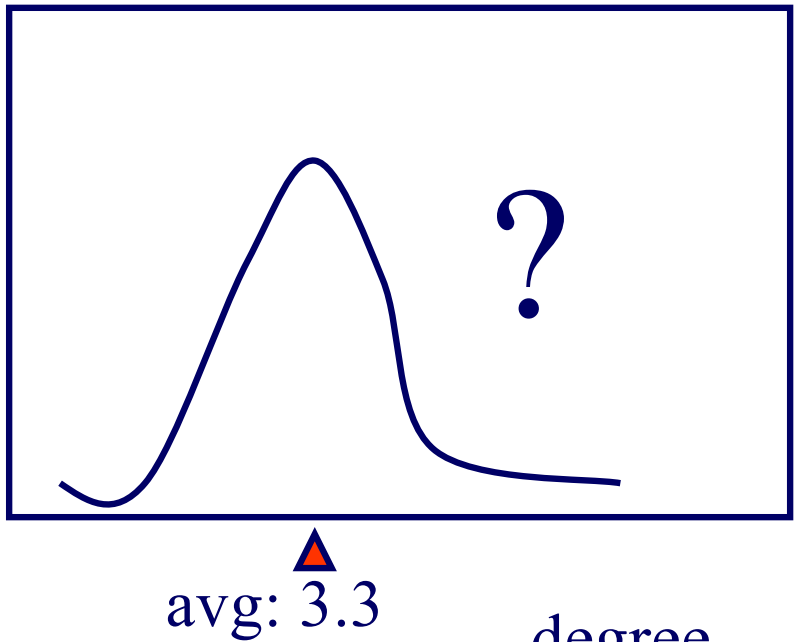
qual. records

Road map

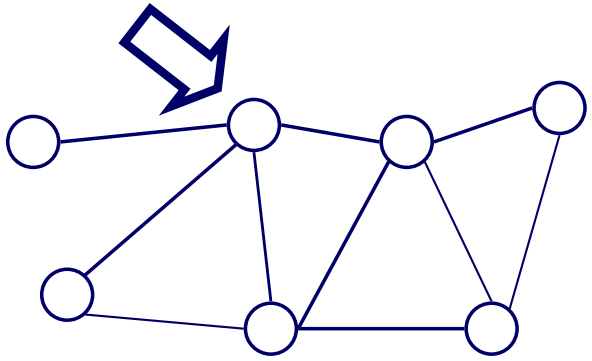
- Motivation – 3 problems / case studies
- Definition of fractals and power laws
- Solutions to posed problems
- More **tools** and examples
- Discussion - putting fractals to work!
- Conclusions – practitioner's guide
- Appendix: gory details - boxcounting plots

A counter-intuitive example

count

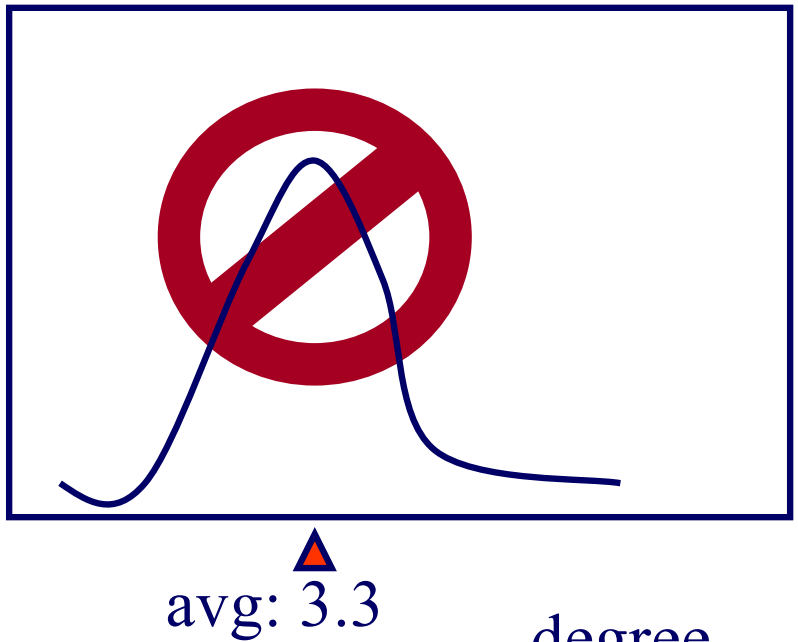


- avg degree is, say 3.3
- pick a node at random – guess its degree, exactly (-> “mode”)



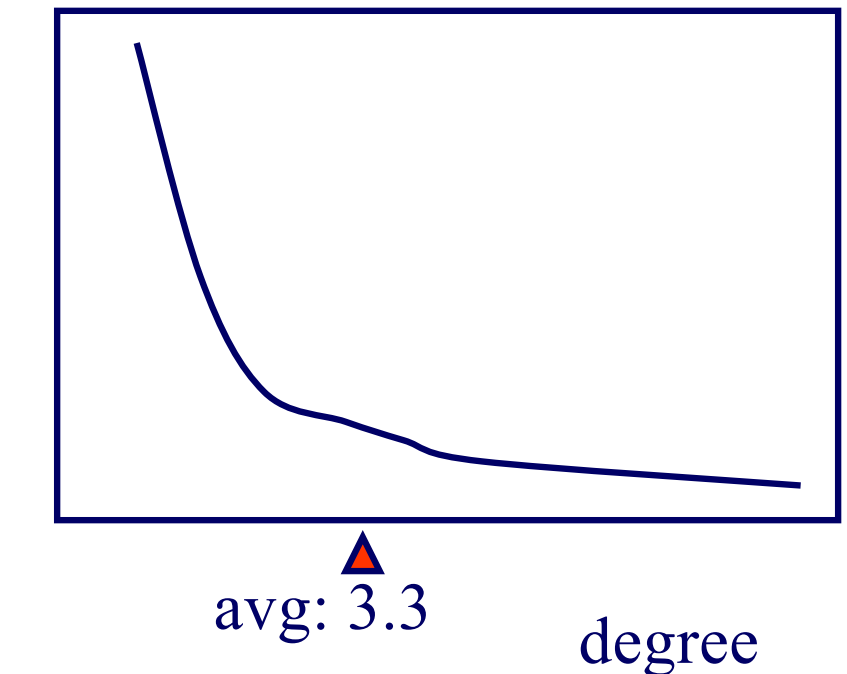
A counter-intuitive example

count



- avg degree is, say 3.3
- pick a node at random – guess its degree, exactly (-> “mode”)
- A: 1!!

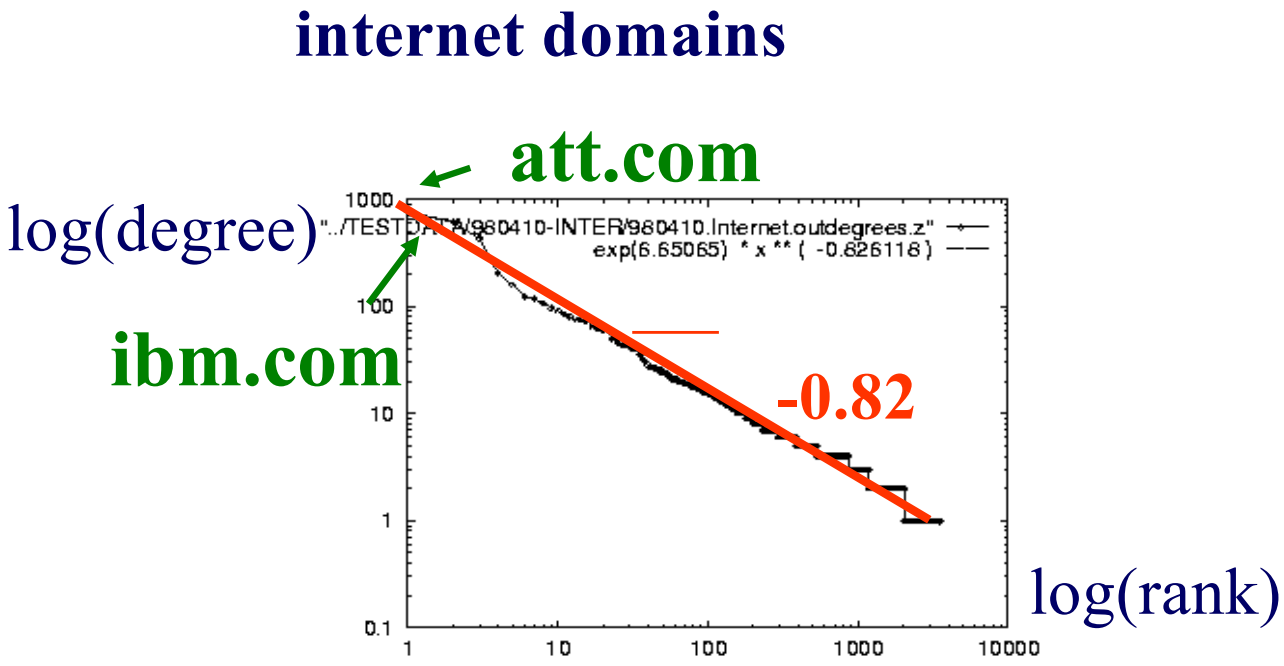
A counter-intuitive example



- avg degree is, say 3.3
- pick a node at random - what is the degree you expect it to have?
- A: 1!!
- A': very skewed distr.
- Corollary: **the mean is meaningless**!
- (and std -> infinity (!))

Rank exponent *R*

- Power law in the degree distribution [SIGCOMM99]

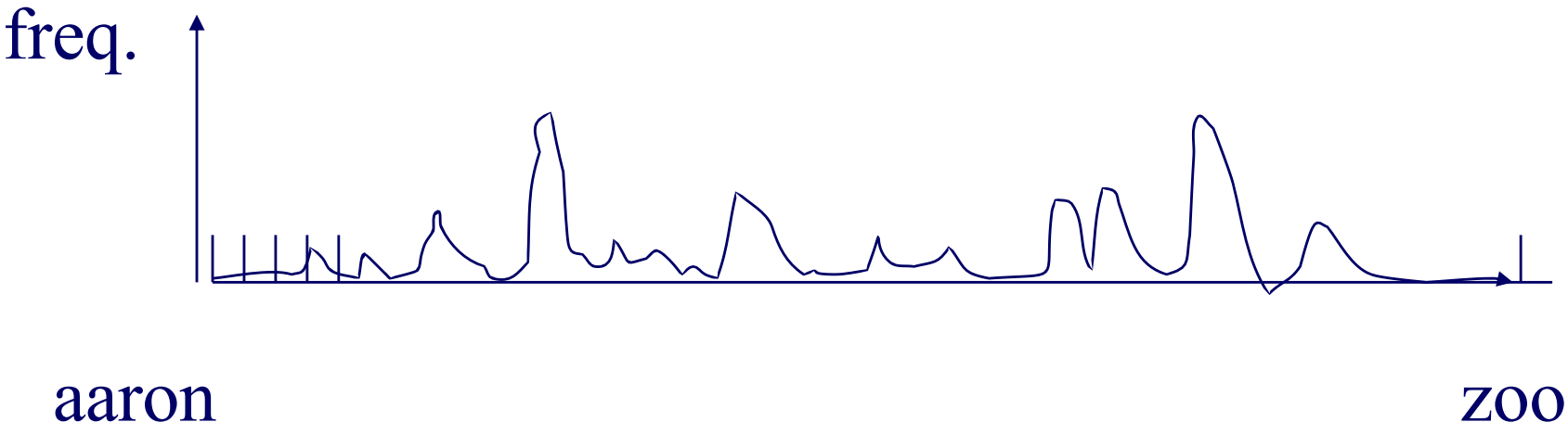


More tools

- Zipf's law
- Korcak's law / "fat fractals"

A famous power law: Zipf's law

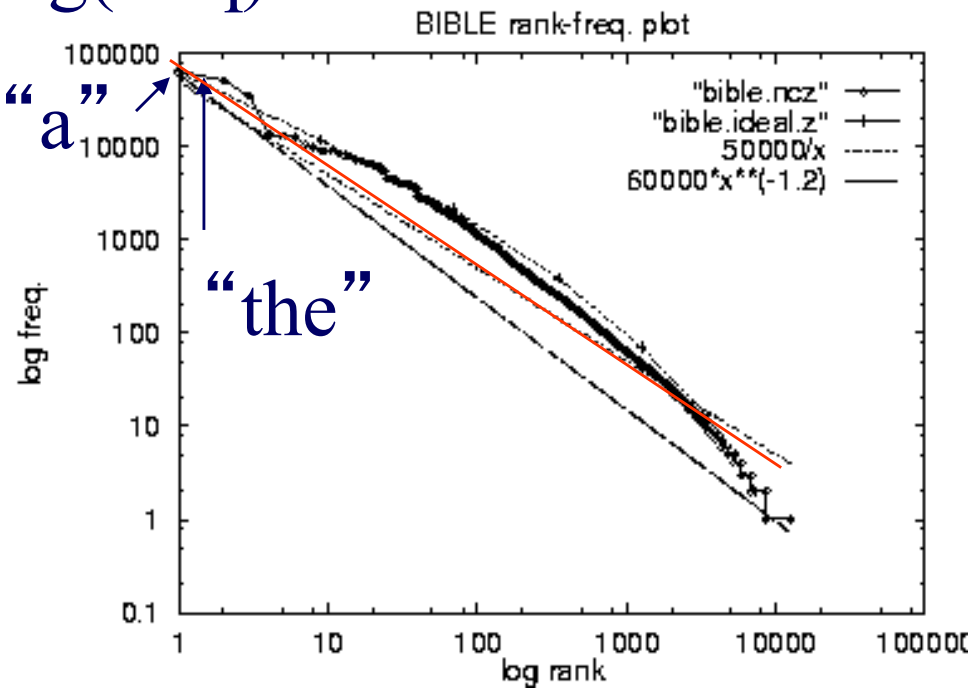
- Q: vocabulary word frequency in a document
- any pattern?



Carnegie Mellon

A famous power law: Zipf's law

log(freq)



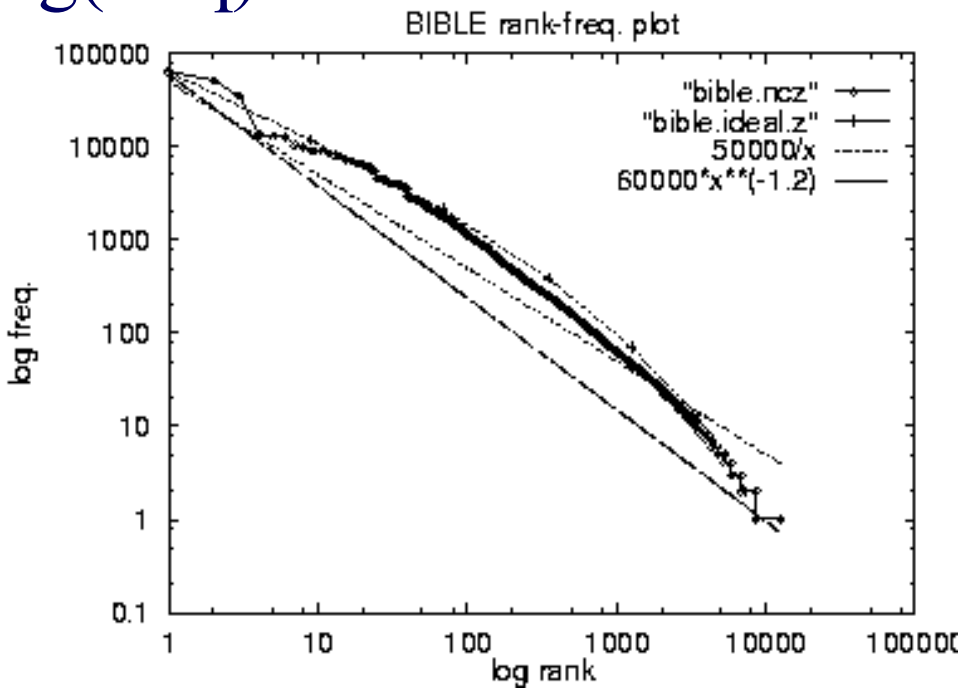
log(rank)

- Bible - **rank** vs **frequency** (log-log)

Carnegie Mellon

A famous power law: Zipf's law

log(freq)

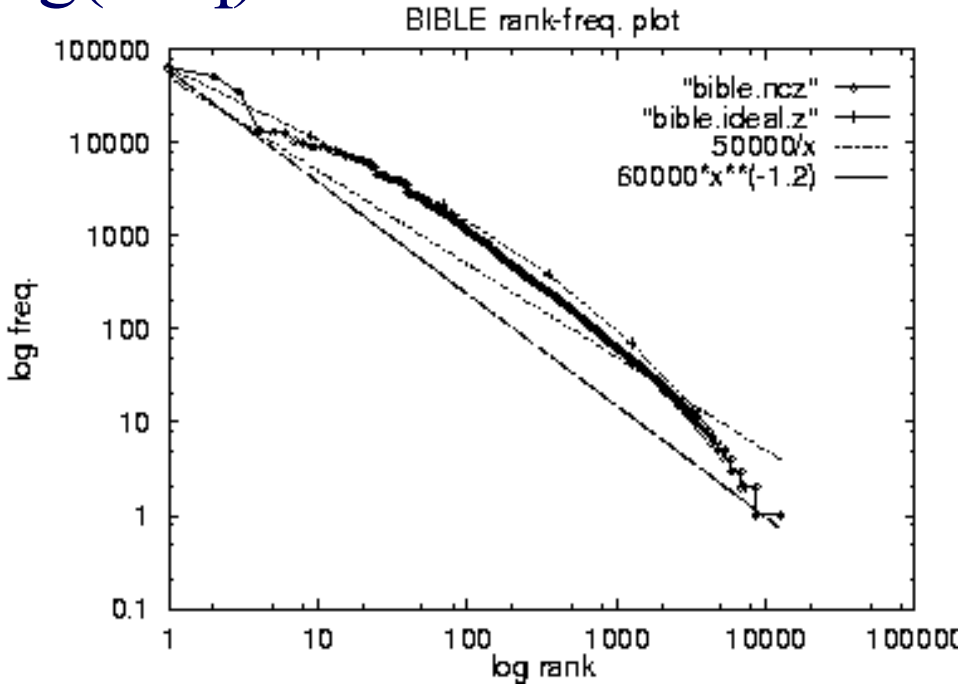


log(rank)

- 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

log(freq)



log(rank)

- Zipf distr:

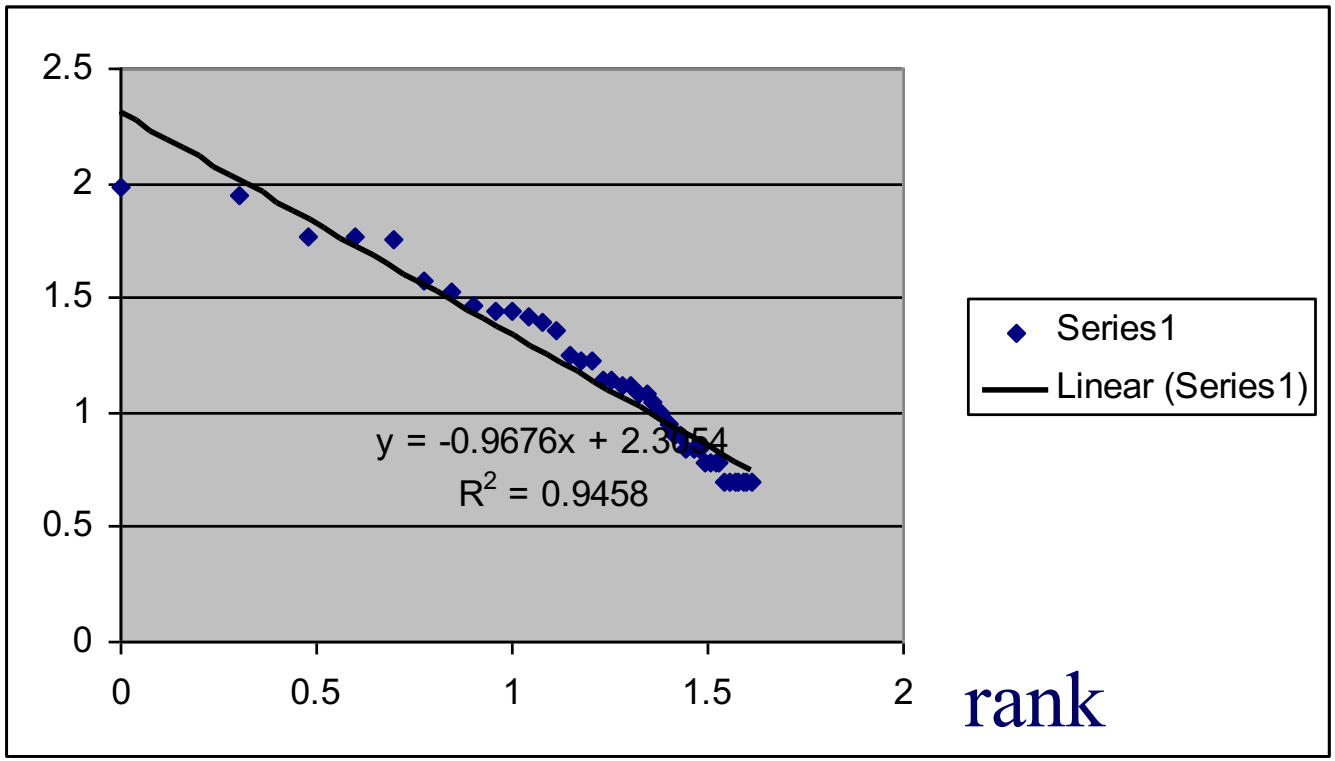
  freq = 1/ rank

- generalized Zipf:

  freq = 1 / (rank)^a

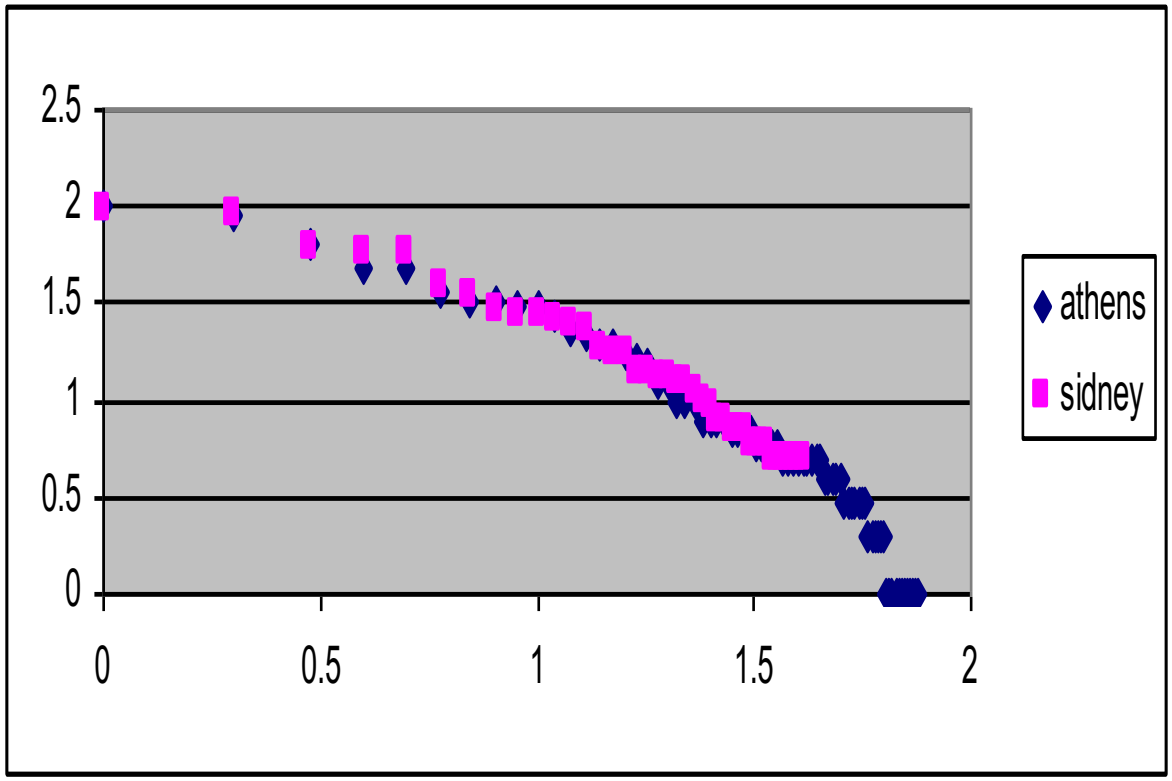
Olympic medals (Sydney):

log(#medals)



Olympic medals (Sydney' 00, Athens' 04):

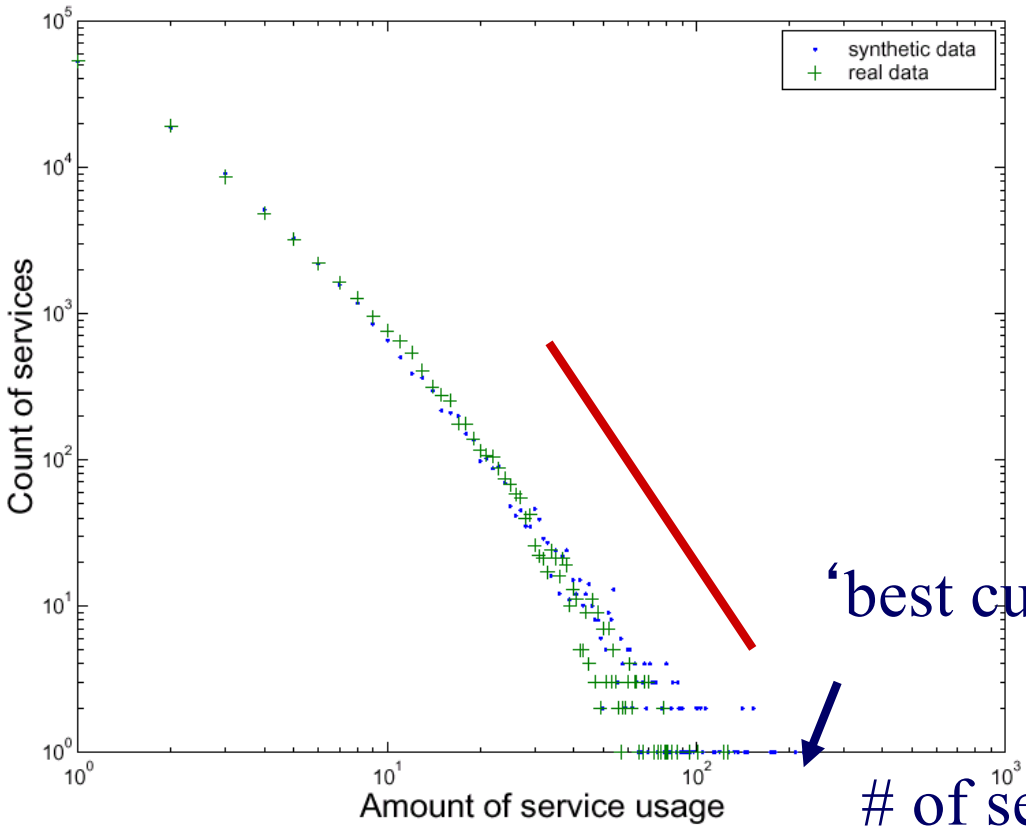
log(#medals)



log(rank)

TELCO data

count of customers



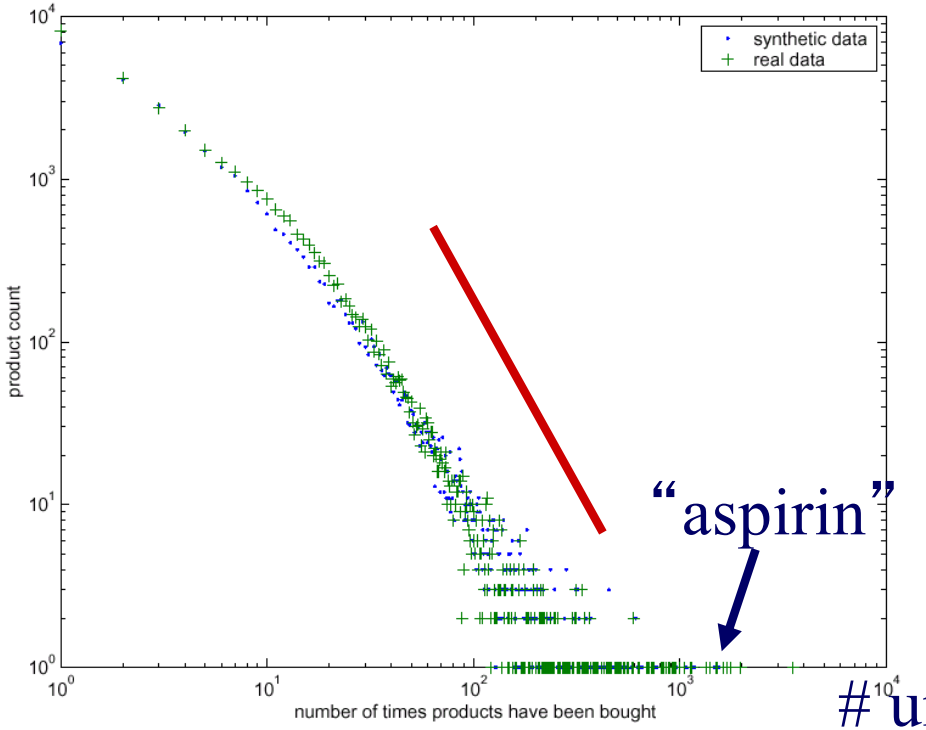
'best customer'

# of service units

Count-frequency plot of real and synthetic data

SALES data – store#96

count of products



Count-frequency plot for store no. 96.

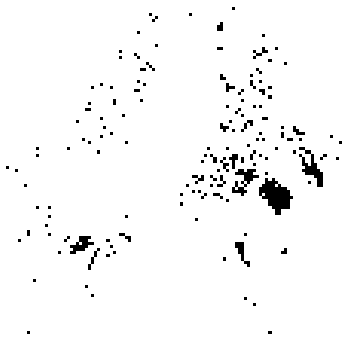
More power laws: areas – Korcak's law

Scandinavian lakes

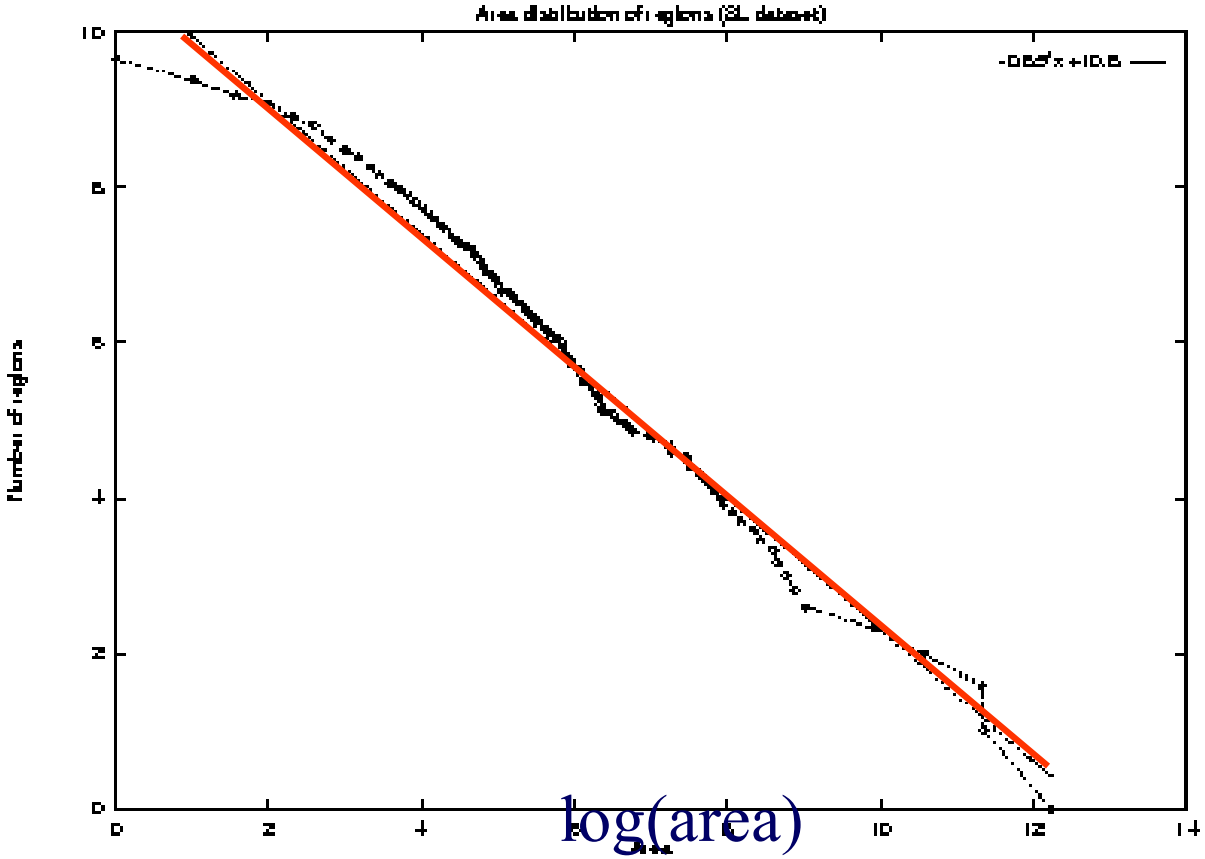
Any pattern?

More power laws: areas – Korcak's law

log(count(>= area))

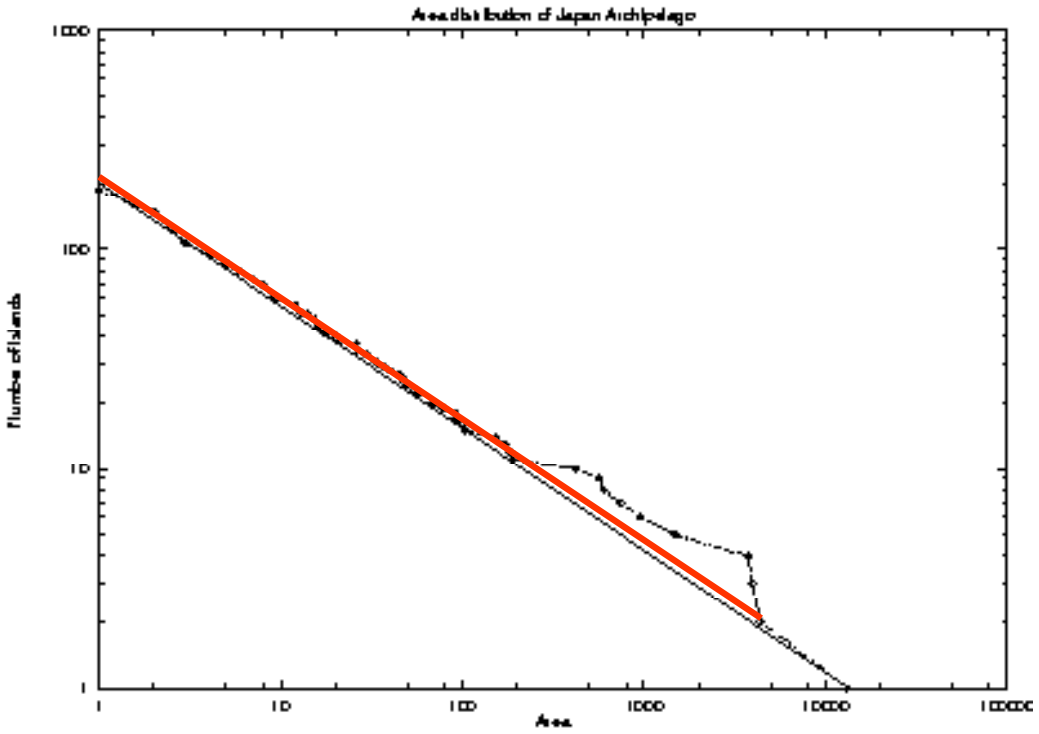


Scandinavian lakes area vs complementary cumulative count (log-log axes)



More power laws: Korcak

log(count(>= area))



Japan islands;

area vs cumulative count (log-log axes)

log(area)

(Korcak's law: Aegean islands)

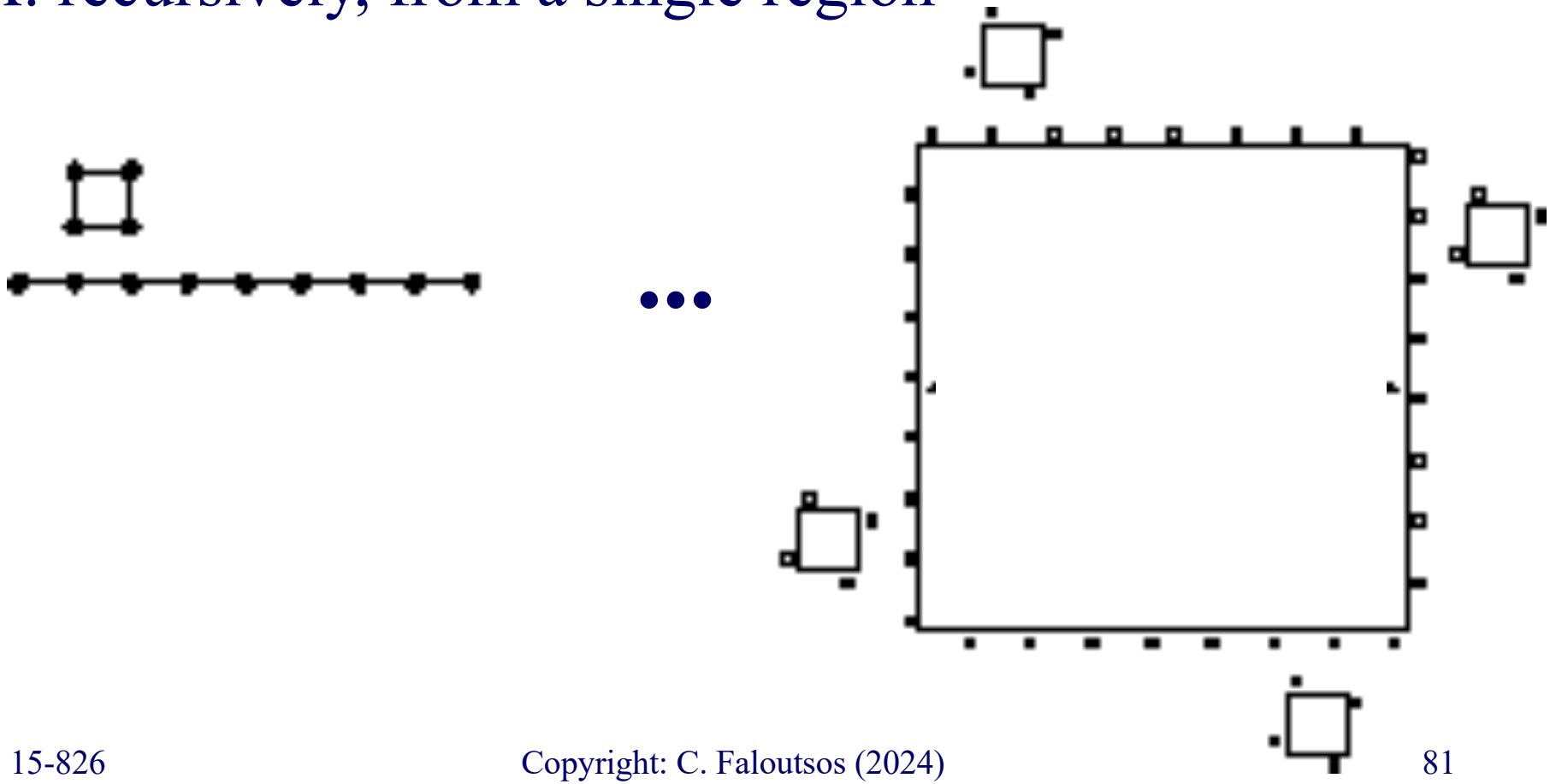
Korcak's law & "fat fractals"

How to generate such regions?

Korcak'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)

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)

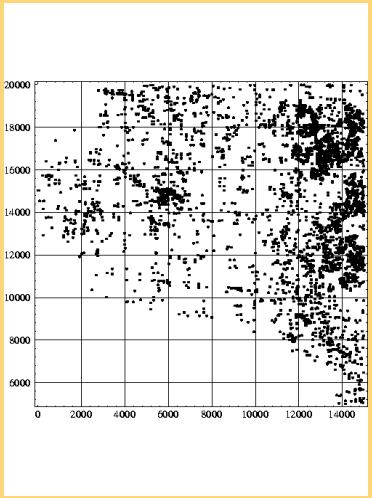
same info

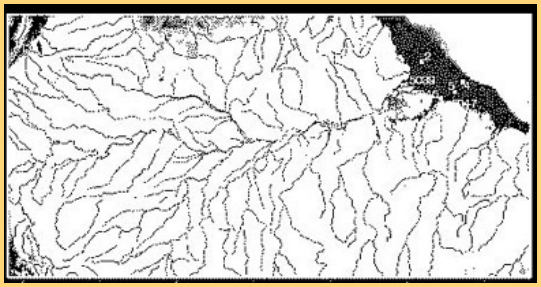
Next:

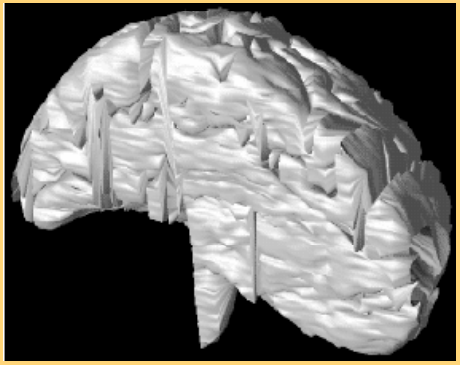
- More examples / applications
- Practitioner's guide
- Box-counting: fast estimation of correlation integral

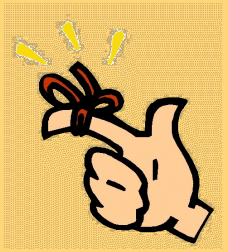
Problem

- What patterns are in real *k*-dim points?



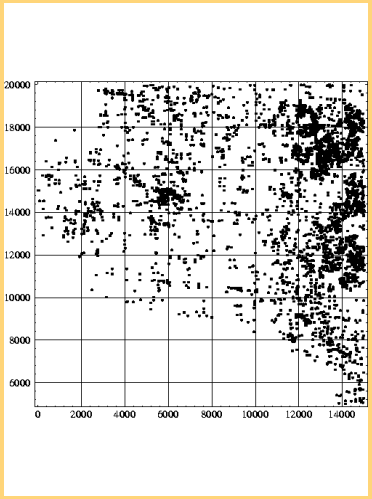


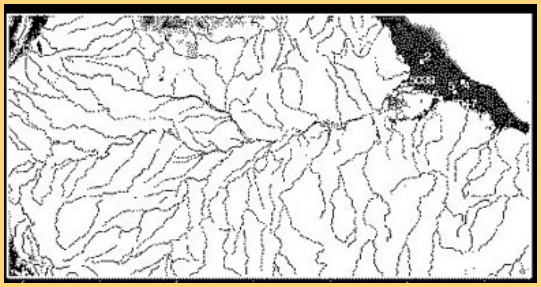


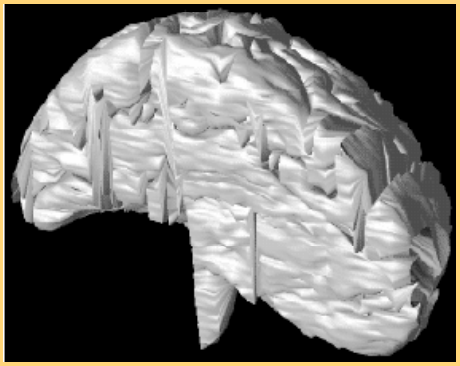


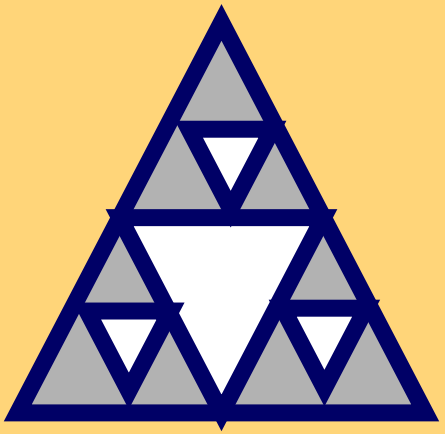
Conclusions

- What patterns are in real *k*-dim points?
- Self-similarity (= fractals -> power laws)





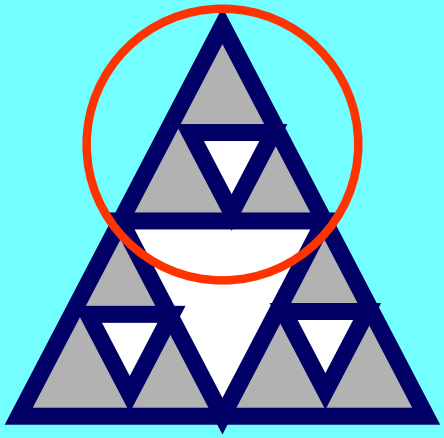




Definitions of f.d.

For mathematical fractal:

$$\text{fd} = \frac{\log(n)}{\log(f)}$$



For real set of points:
fd in the **range** (r1, r2):
Slope of corr. integral