wdv-notes 148
Stand: 23.JUN.1993 (2.)
Wiss.Datenverarbeitung
FREIE UNIVERSITÄT BERLIN

Developing Algorithms for Measuring Fractal Dimension.

© 1992–1993 Notes by *John J. Sarraille*

Math

Mit neueren Methoden der Signaldatenverarbeitung ist es heute möglich, die wahre (fraktale) Natur auch bei biologischen Systemen zu untersuchen und zu erkennen [4].

Mit dem von J. Sarraille und P. DiFalco 1992 entwickelten Programm ***FD3*** bzw. ***FDim*** [3] lassen sich beispielsweise aus geeigneten experimentellen Daten fraktale Dimensionen berechnen bzw. abschätzen.

Im folgenden ist die Beschreibung von J. Sarraille wiedergegeben, wie sie im April 1993 vom Server *lyapunov.ucsd.edu* (132.239.86.10) des Institute for Nonlinear Science (San Diego, USA) kopiert wurde.

What are fractals? Does anybody know exactly?

Ironically, the definition of the word "fractal" is somewhat fuzzy. I won't attempt to give a rigorous definition of the term.

Here is what we think fractals are.

In popular terms, fractals are sets that appear to have complex structure no matter what scale is used to examine them. One consequence of this definition is that true fractals must be infinite sets. Often, but not always, fractals have the same granularity across scales, or are even self-similar across scales, so that one tends to see the same quality of structure in a fractal as one zooms in on it. Well-behaved fractals such as these are the ones that have a well-defined fractal dimension.

Fractal curves can fill space.

Many fractals are *continuous* sets which are nevertheless very much lacking in *smoothness*, to the point that they fill up more space than one would normally expect. For example, we tend to expect a "curve" to be one-dimensional. A *fractal* curve can have a graph that appears to occupy the entire area of a square in two-dimensional space.

How do fractals relate to chaotic systems?

The phase-plane and Poincare maps of chaotic systems have a fractal structure. We can hope to recognize, classify and understand such maps of chaos by measuring their fractal properties. This in turn shows promise for increasing our understanding of the complex forces, processes, and phenomena that affect ourselves, our world, and even the universe.

Fractal dimension is a tool for recognizing chaos.

The *fractal dimension* is such a fractal property. It's use as a classification tool holds promise because it is relatively easy to estimate the fractal dimension of a set S, because we know that a chaotic system gave rise to the set S if this dimension is not an integer, and because the value of the dimension gives us some indication of what type of chaotic system gave rise to S.

These notes are an overview of methods and algorithms for computing fractal dimension.

I'll review a few methods that can be used to calculate fractal dimension. The methods are not equivalent. The different methods can give different values for the fractal dimension of a given set, even in cases where the values are absolutely precise. However, the methods are strongly related, and often do yield the same results.

What does fractal dimension measure?

In all cases that I know about, fractal dimension is an indication of how much space a given set "comes near". For example, a ball of twine appears 3-dimensional because it "comes near" enough of the volume of a sphere to deceive our eyes. The Cantor set is another example. All its points lie on a straight line, but it is so full of gaps that no complete line segment, no matter how short, is contained in it. This is reflected in the fact that its fractal dimension is about 0.63, quite a bit less than 1.

CAPACITY DIMENSION

The cardinality of a covering is inversely proportional to ε^D.

Suppose you have a line segment. Suppose that you try to cover it with squares of a certain size, and find that you need three of them to do it. If you then tried to see how many squares of half the original size were required to cover the segment, it would not surprise you to find that it requires six – twice as many. Thus the number of squares required to cover the segment is inversely proportional to the size of the squares. The covering of any smooth, continuous curve works the same way, provided that the size of the squares is small enough so that the curve is approximated well by straight line segments at that scale.

Thus, for one-dimensional objects, we see that

$$N(\varepsilon) = k/\varepsilon \qquad (1)$$

(approximately), where ε is the side of the square, $N(\varepsilon)$ is the number of squares of that size required to cover the set, and k is some constant. Now suppose that you are covering a scrap of paper with little squares. In this case, if you halve the size of the squares, it takes four of the smaller squares to cover what one of the larger squares would cover, and so you would expect $N(\varepsilon)$ to increase by a factor of four when ε is halved. This is consistent with an equation of the form:

$$N(\varepsilon) = k/\varepsilon^2 . \qquad (2)$$

From this analysis, it seems reasonable to hypothesize that for more arbitrary sets:

$$N(\varepsilon) = k/\varepsilon^D , \qquad (3)$$

where D is the (possibly fractional) dimension of the set. In other words, we can hope to measure how much of two-dimensional space some subset of it comes near by examining how efficiently the set can be covered by cells of varying size.

How can we use the relation $N = \varepsilon^D$ to find D ?

By solving the formula for D, and taking the limit as $\varepsilon \to 0$, we get the *capacity method* of estimating D. It seems to take into account the fractal's infinitesimal, or asymptotic, structure. This yields the formula:

$$D = \lim_{\varepsilon \to 0} \frac{\log[N(\varepsilon)]}{\log[1/\varepsilon]} . \qquad (4)$$

(By the way, if we assume that the set is scaled so that it fits into a square with side 1, then we get $N(1) = k = 1$. In any case, the constant k will disappear from the equation when the limit is taken).

How do we estimate D when we only have a finite set?

In the case of real data, we must always work with a finite set S (which we may or may not be able to interpret as a *sample* of the points from some infinite set). In the case of a finite set, the limit above is always *zero* because eventually ε will be so small that there will only be one point in each occupied cell. Once ε becomes sufficiently small, $N(\varepsilon)$ becomes equal to $\|S\|$, the number of points in the set, and the limit above is the same as

$$D = \lim_{\varepsilon \to 0} \frac{\log \|S\|}{\log[1/\varepsilon]} , \qquad (5)$$

which equals zero, because $\log(\|S\|)$ is a constant and $\log(1/\varepsilon)$

grows without bound as ε tends to zero. This is just a reflection of the fact that all finite sets are zero dimensional – they have the same dimension as a single point. To estimate fractal dimension with a computer algorithm, we digitize as large a sample of the set as is practical, calculate (or estimate) the values of $N(\varepsilon)$ for some *range* of values of ε, and take as our estimate of the fractal dimension some sort of average of those values of

$$\frac{\log[N(\varepsilon)]}{\log[1/\varepsilon]} \tag{6}$$

(In cases where the scaling has not made the proportionality constant k in the relation

$$N(\varepsilon) = k \cdot \varepsilon^{D} \tag{7}$$

equal to 1, the trick is to average instead the slope of a plot of $\log[N(\varepsilon)]$ versus $\log(1/\varepsilon)$. This effectively corrects for the constant, the slope being the desired estimate of D).

Disadvantages of capacity dimension.

1. It does not account for the *frequency* with which the set in question might "visit" the covering cells, and thus local properties of the set – properties pertaining to neighborhoods of individual points – are not distinguishable.
2. Calculating truly minimal coverings of a set is computationally intensive.

Advantages of capacity dimension (for our group).

1. We have a simplified algorithm [2] that has been found accurate by empirical measures, is easy to program, and is potentially very fast.
2. We have been able to adapt the algorithm to give us other versions of fractal dimension with nearly the same speed, and apparently the same accuracy.

INFORMATION DIMENSION

Background: How is it that -log(*P*) measures information or surprise?

A measure of the amount of information (surprise) associated with the occurrence of an event E with probability $P(E)$ is

$$I(E) = -\log[P(E)] = \log[1/P(E)]. \tag{8}$$

To understand why $I(E)$ is a good measure of "surprise", note that if $P(E) = 1$, then the event is certain, and $I(E) = 0$ – there is no surprise at all when the event occurs. If $P(E) = 0$, then since infinity is the limit of $I(E)$ as $P(E)$ approaches 0, this expresses our infinite surprise to witness the impossible. The information function $I(E)$ also has the nice property that when an event E is the result of the simultaneous occurrence of two independent events F and G, $I(E) = I(F) + I(G)$. This agrees with our intuitive feeling that the information conveyed by the simultaneous occurrence of two independent events is the sum of what is conveyed by each event individually.

Formulation of information dimension.

In this method, one covers the sample set S with a minimal number $N(\varepsilon)$ of cells of size ε. Let $||S||$ denote the number of elements in S, and for each of the $N(\varepsilon)$ cells, let $N(\varepsilon, i)$ be the number of points in cell number i. Set $P(\varepsilon, i) = N(\varepsilon, i)/||S||$ – the sample probability that a point of S is in the i'th cell.

I(ε) measures average surprise in learning which ε-cell a point is in.

The quantity $-\log[P(\varepsilon, i)]$ measures the information conveyed by knowing that a point of S is in the i'th cell of the covering, and the *average*

$$I(\varepsilon) = \sum_{i=1}^{N(\varepsilon)} -P(\varepsilon, i) \cdot \log[P(\varepsilon, i)] \tag{9}$$

measures the average information conveyed by knowing what cell a point of S is in.

Information dimension is a generalization of capacity dimension.

Now note that when all the probabilities are equal, each $P(\varepsilon, i)$ is equal to $1/N(\varepsilon)$, and the sum above is just $I(\varepsilon) = \log[N(\varepsilon)]$. This is, in fact, the maximum possible value of $I(\varepsilon)$. We can interpret the lower values that might occur as quantifying the non-uniformity of the distribution of the points, or alternatively, as correcting the dimension estimate by giving less weight to the cells that contain relatively few points.

Thus if we define a fractal dimension by:

$$D = \lim_{\varepsilon \to 0} \frac{I(\varepsilon)}{\log[1/\varepsilon]}, \tag{10}$$

we get a formulation of fractal dimension that is similar to the capacity dimension:

$$D = \lim_{\varepsilon \to 0} \frac{\log[N(\varepsilon)]}{\log[1/\varepsilon]}, \tag{11}$$

but which gives a value different from the capacity dimension when the distribution of the set over the covering is non-uniform.

How is the formula for information dimension used in practice to yield an estimate of fractal dimension?

This is done just as described previously — averaging techniques are used in place of the limit calculation, which is not feasible when working with a finite sample (the calculation of the limit may be computationally intractable anyway, even when we know how to specify the infinite set precisely).

CORRELATION DIMENSION

Formulation of correlation dimension.

Correlation dimension can be calculated using the distances between each pair of points in the set S,

$$s(i, j) = \| x(i) - x(j) \|. \tag{12}$$

This gives us $||S||^2$ values, and we can calculate sample correlation functions $C(r)$ with the formula:

$$C(r) = \frac{1}{\| S \|^2} \cdot \{\#\text{ of pairs } (i, j) \text{ with } s(i, j) < r\} \tag{13}$$

These $C(r)$ have been found to exhibit a power law like the one we saw for capacity dimension: $C(r) = k\, r^D$; and we can aim to find D with estimation techniques derived from the formula:

$$D = \lim_{r \to 0} \frac{\log[C(r)]}{\log[r]} \tag{14}$$

Optical device for estimating correlation dimension!

Correlation dimension can be measured very directly with an *optical device*. Two films are prepared containing the sample of points, in such a manner that light is able to pass through the images of the points, but is blocked by the rest of the film. The two films are held parallel to each other while light is focused on one of the films, and the total quantity of light emerging from the second film is measured. Light passing through a point x on the first film shines on the second film, over the surface of a disk centered on the image of x, and having radius r determined by the distance between the films. Thus the points at distance r or less from x on the second film emanate light that came through the image of x on the first film.

It is not hard to see that the total light emanating from the second film is proportionate to $C(r)$, the number of pairs of points that are at distance r or less from each other. (This is true for wavelengths of incoherent light much smaller than the size of the apertures representing the points in the set). With such a device,

Developing Algorithms for Measuring Fractal Dimension.

© 1992–1993 Notes by *John J. Sarraille*

it is thus possible to rapidly calculate values of $C(r)$ for various values of r, and to arrive at an estimate of D. This is potentially a superb method of calculating D in terms of speed and cost effectiveness. The technique has been used by Lee and Moon [1] on an experimental device, with excellent results. I have not heard if such a device is yet commercially available.

The catch? It's only good for an embedding dimension of 1 or 2.

If the set S cannot be embedded in a plane, this optical technique is not suitable. (Is there hope for higher embedding dimensions by using projections and reflectors? We don't know).

RELATIONS AMONG THE DIMENSIONS

An inequality relates three of the dimensions.

$$\begin{matrix}\text{correlation}\\ \text{dimension}\end{matrix} \le \begin{matrix}\text{information}\\ \text{dimension}\end{matrix} \le \begin{matrix}\text{capacity}\\ \text{dimension}\end{matrix} \qquad (15)$$

Getting all three from the same formula!

Moon indicates that estimators of all three dimensions can be derived from the same formula. One takes

$$I(q,\varepsilon) = \frac{1}{1-q} \cdot \log\left[\sum_{m=1}^{N(\varepsilon)} P(m,\varepsilon)^q\right] \qquad (16)$$

where $P(i, \varepsilon)^q$ is the q'th power of the proportion of the points of S that lie in the i'th cell of a minimal covering by cells of size ε, and $N(\varepsilon)$ is the number of cells in such a covering. If we define

$$D(q) = \frac{I(q,\varepsilon)}{\log[\varepsilon]} \qquad (17)$$

then for q = 0, 1, and 2, $D(q)$ is an estimator of capacity dimension, information dimension, and correlation dimension, respectively. (The limit of $I(q, \varepsilon)$ as q–> 1 must be used in the case of q = 1).

This has made us realize that all three can be estimated in *N* log(*N*) time!

One of us noticed that the basic methodology reported in [2] can be adapted to provide an algorithm for calculating good estimates of the quantities $P(i, \varepsilon)$ in N log(N) time, where N is the number of points in the set S. This made us realize that a very fast algorithm exists for computing a good estimate of any $D(q)$ – for any embedding dimension. The ***FD3*** program [3] exploits this idea to get estimates for q = 0, 1, and 2.

We must be careful, however, not to be over-trusting of the values these algorithms give us. We do not yet understand enough about error bounds to make pronouncements about the accuracy of such algorithms. Nevertheless, these techniques appear promising to us, and we are hopeful that a combination of empirical and formal studies will prove them to be adequate.

POINTWISE DIMENSION

The formulation of pointwise dimension.

The pointwise dimension is sensitive to the behavior of the set in the vicinity of a specific point. Let S be the sample of the set whose dimension is to be estimated. Let R be the embedding space (if S is a subset of the (real) plane, then R could be the plane. R may be 3-D space, or whatever size space is required to allow S to be a subset). Let B(r; x) denote the set of all points in R whose distance from x is less than r. B(r; x) is a disk if R is 2-D, and is a filled-in sphere if R is 3-D). Let $N(r, x)$ denote the number of elements of S contained in B(r; x). Let $P(r, x) = N(r, x)/||S||$, where $||S||$ is the number of points in S. Then $P(r, x)$ is a sample probability – an estimate of the probability of finding a point of S inside B(r; x). It should be intuitively clear that $P(r, x)$ will be proportionate to r when S is a uniform sampling of a 1-dimensional set, and proportionate to r^2 when S is a uniform sampling of a 2-dimensional set. Thus, like we did in the previous section, we might settle on the formula

$$P(r,x) = k \cdot r^{D(r,x)}, \qquad (18)$$

and we might further assume that k = 1, as long as the points are scaled so that the distance between each pair of them is less than 1. Note that D is a function of x in this formulation, and that, in general, it would be possible for different values of D to exist at different points x. D is also a function of r here, but we would expect there to be a limit value of D as r tends to 0. This leads to:

$$D(x) = \lim_{r \to 0} \frac{\log[P(r,x)]}{\log[r]} \qquad (19)$$

Using pointwise dimension to get an overall fractal dimension estimate.

One may estimate $D(x)$ by suitably averaging computed values of log[$P(r, x)$]/log(r). This gives us a way to measure the local fractal properties of the set, near any of its points we care to choose. One may also estimate the overall fractal dimension of S by averaging some of the estimates of $D(x)$, or by working with average values of $P(r, x)$.

References.

[1] Moon, Francis C,: Chaotic Vibrations. New York: J. Wiley 1987. ISBN: 0-471-85685-1.

[2] Liebovitch and Toth: A Fast Algorithm To Determine Fractal Dimensions By Box Counting. Physics Letters A, 141, 386–390 (1989).

Information about the author of these notes.

John Sarraille, Professor, Computer Science Department, CSU Stanislaus, Turlock, CA 95380, USA.
e-mail: john@ishi.csustan.edu

COPYRIGHT NOTICE

Copyright © 1992 by John Sarraille and Peter DiFalco (john@ishi.csustan.edu).

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The algorithm used in this program was inspired by the paper entitled "A Fast Algorithm To Determine Fractal Dimensions By Box Counting", which was written by Liebovitch and Toth, and which appeared in the journal "Physics Letters A", volume 141, pp 386–390, (1989).

This program ***FD3*** is not warranteed: use at your own risk!

```
****************************************************************
  FRACTAL DIMENSION REPORT -- by fd software (DiFalco/Sarraille)
****************************************************************

Reporting on file named: koch.dat.

Getting the embedding dimension ...
Embedding Dimension taken to be: 2

Finding max and min values in data ...

Minimum value in input file is: -1.000000 ...
Maximum value in input file is: 1.000000 ...

32 different cell sizes will be used ...
Data will be shifted and re-scaled so that minimum coordinate
value is ZERO and maximum coordinate value is: 4294967295 ...

Allocating storage ... Done ...
Initializing queues ... Done ...
Loading data ... Done ...
Sorting the data ... Done ...
Doing sweep to get counts ... Done ...

[log(epsl)]  [CellCount]  [log(CellCount)]  [informtn]  [-log(SumSqrFreqs)]
 = [log(e)]   = [N(e)]      = [logN(e)]      = [I(e)]     = [-logSSF(e)]
        0         3072          11.58496    11.58496             11.58496
        1         3072          11.58496    11.58496             11.58496
        2         3072          11.58496    11.58496             11.58496
        3         3072          11.58496    11.58496             11.58496
        4         3072          11.58496    11.58496             11.58496
        5         3072          11.58496    11.58496             11.58496
        6         3072          11.58496    11.58496             11.58496
        7         3072          11.58496    11.58496             11.58496
        8         3072          11.58496    11.58496             11.58496
        9         3072          11.58496    11.58496             11.58496
       10         3072          11.58496    11.58496             11.58496
       11         3072          11.58496    11.58496             11.58496
       12         3072          11.58496    11.58496             11.58496
       13         3072          11.58496    11.58496             11.58496
       14         3072          11.58496    11.58496             11.58496
       15         3072          11.58496    11.58496             11.58496
       16         3072          11.58496    11.58496             11.58496
       17         3072          11.58496    11.58496             11.58496
       18         3072          11.58496    11.58496             11.58496
       19         3072          11.58496    11.58496             11.58496
       20         3072          11.58496    11.58496             11.58496
       21         3072          11.58496    11.58496             11.58496
       22         3072          11.58496    11.58496             11.58496
       23         3072          11.58496    11.58496             11.58496
       24         2660          11.37721    11.31231             11.22888
*************************************************************************
       25         1332          10.37938    10.20093             10.05683
       26          566           9.14466     8.92491              8.79624
       27          210           7.71425     7.47647              7.37632
       28           94           6.55459     6.26584              6.11553
       29           40           5.32193     5.07558              4.93563
       30           14           3.80735     3.57718              3.49477
*************************************************************************
       31            4           2.00000     2.00000              2.00000
       32            1           0.00000     0.00000              0.00000

Two-Point Estimates of FD's:
[log(e)]  [logN(e)-logN(2e)]  [I(e)-I(2e)]  [logSSF(2e)-logSSF(e)]
        0           0.00000       0.00000                0.00000
        1           0.00000       0.00000                0.00000
        2           0.00000       0.00000                0.00000
        3           0.00000       0.00000                0.00000
        4           0.00000       0.00000                0.00000
        5           0.00000       0.00000                0.00000
        6           0.00000       0.00000                0.00000
        7           0.00000       0.00000                0.00000
        8           0.00000       0.00000                0.00000
        9           0.00000       0.00000                0.00000
       10           0.00000       0.00000                0.00000
       11           0.00000       0.00000                0.00000
       12           0.00000       0.00000                0.00000
       13           0.00000       0.00000                0.00000
       14           0.00000       0.00000                0.00000
       15           0.00000       0.00000                0.00000
       16           0.00000       0.00000                0.00000
       17           0.00000       0.00000                0.00000
       18           0.00000       0.00000                0.00000
       19           0.00000       0.00000                0.00000
       20           0.00000       0.00000                0.00000
       21           0.00000       0.00000                0.00000
       22           0.00000       0.00000                0.00000
       23           0.20775       0.27265                0.35609
       24           0.99783       1.11138                1.17204
*************************************************************************
       25           1.23472       1.27602                1.26059
       26           1.43041       1.44844                1.41992
       27           1.15966       1.21063                1.26079
       28           1.23266       1.19026                1.17990
       29           1.51457       1.49840                1.44086
*************************************************************************
       30           1.80735       1.57718                1.49478
       31           2.00000       2.00000                2.00000

25 is the smallest cell size used in
the overall dimension estimates
below.  The largest cell size is 30.
Data above corresponding to
this range is between rows of aster-
isks.

Least-Square Estimates based on
Indicated Cell Range:

Fractal Dimension (Capacity)    =
1.29966

Fractal Dimension (Information) =
1.31078

Fractal Dimension (Correlation) =
1.30437
```

*Nebenstehend ist das Beispiel eines Dimension-Reports dargestellt, wie er vom Programm **FDim** (P 511) erzeugt wird* [3]. *Bei diesem Programmlauf wurden auf dem HAIRcluster die Daten von 3073 Punkten der »Snowflake« – einer bekannten fraktalen Koch-Kurve – verwendet.*

Zum „Dimension Report“

Zur Abschätzung der fraktalen Dimensionen werden lediglich die berechneten Werte zwischen den Sternlinien (***) herangezogen (linearer Bereich).

Für diese Daten wird für jede Spalte eine Ausgleichsgerade mit log(ε) als abhängiger Variablen berechnet (fit). Aus den jeweiligen Anstiegen (slope) der Ausgleichsgeraden ergeben sich dann die Abschätzungen für die Kapazitäts-Dimension, die Informations-Dimension und die Korrelations-Dimension.

In der unteren Tabelle „Two-Point Estimates of FD's“ sind jeweils die Differenzen zwischen den Werten zweier Zeilen der oberen Tabelle berechnet worden. Diese Werte geben eine „laufende“ Abschätzung der drei fraktalen Dimensionen [5] des betrachteten Signals oder Objekts an.

Die Daten für die Dimensionsanalyse mit ***FD3*** bzw. ***FDim*** [3] müssen als einfache Textdatei mit einer bestimmten Struktur vorliegen. Auf der ersten Zeile ist dabei stets die Anzahl der Datenpunkte in der Datei anzugeben. Die Koordinaten der Datenpunkte sind dann jeweils auf einer Zeile, durch Leerzeichen oder einen Tabulator getrennt, anzugeben. Weitere Informationen sind in den wdv-notes Nr. 176 [3] zu finden.

Weitere Literatur

[3] Dittberner, K.-H.: Ermittlung Fraktaler Dimensionen mit FDim (P 511). FU Berlin (IfP): wdv-notes Nr. 176, 1992–1993.

[4] Dittberner, K.-H.: Die Fraktalen Dimensionen der Natur. FU Berlin (IfP): wdv-notes Nr. 88, 1984–1992. Hier sind weitere wichtige Literaturhinweise zu finden.

[5] Hentschel, H. G. E. and Procaccia, I.: The infinite number of generalized dimensions of fractals and strange attractors. Physica 8D, 435–444 (1983).