Minimal box size for fractal dimension estimation

E. Rosenberg

AT&T Labs, Middletown, NJ 07748, USA. Email: ericr@att.com

Keywords: Box counting, Fractal dimension, Resolution, Self-similar, Scale.

Abstract: We extend Kenkel's model for determining the minimal allowable box size s^* to be used in computing the box counting dimension of a self-similar geometric fractal. This minimal size s^* is defined in terms of a specified parameter ε which is the ing dimension of a self-similar geometric fractal. This minimal size s* is defined in terms of a specified parameter ε which is the
deviation of a computed slope from the box counting dimension. We derive an exact impl the equation of a computed stope from the box counting dimension. We derive an exact implicit equation for s³ for any s. We solve
the equation using binary search, compare our results to Kenkel's, and illustrate how s^{*} code for the binary search is provided. We also derive a closed form estimate for s* having the same functional form as Kenkel's empirically obtained expression. empirically obtained expression.

Introduction

One of the most widely used fractal dimensions is the box counting dimension d_B . Suppose we wish to estimate d_B from deviates from d_B by no more the *N* points, e.g., pixels in a 2-dimensional image. We cover the *N* points with boxes of side length *s*. If the object is 2-di- or four decimal mensional, the boxes are squares; if the object is 3-dimen-decreased until
the N points of the N points with boxes of side of side of side of side \mathbb{R} . sional, the boxes are cubes; in general for an *E*-dimensional object, the boxes are *E*-dimensional hypercubes. Let $B(s)$ be box size: the number of boxes containing at least one of the *N* points. When *s* is sufficiently large, all points lie in one box, so $B(s)$ = 1. As *s* decreases, *B*(*s*) increases or remains constant. For $\frac{s_1}{10}$ all sufficiently small box sizes, each non-empty box contains a single point, so $B(s) = N$. $S(s) = N$

The box counting dimension d_B of a geometric object is defined as (Mandelbrot 1983)

$$
d_B = -\lim_{s \to 0} \frac{\log B(s)}{\log s}.
$$

There are examples for which d_B does not exist (Falconer ε , by standard ε 2003) but in this paper we assume d_B exists. In practice, d_B
is computed by evaluating $B(s)$ for a set of J values of s is computed by evaluating *B*(*s*) for a set of *J* values of *s*, which we denote by s_j , $j = 1, 2, \dots, J$. Then, typically, a line is fitted (often using linear regression) to the *J* pairs (−log 1 s_j , log $B(s_j)$), and the slope of this line is the estimate of d_B . Assume the s_j values are ordered so that $s_1 < s_2 < ... < s_J$. We want *sJ−s*1 as large as possible, so as to minimize the error in estimating d_B from the *J* pairs (log s_j , log $B(s_j)$). However, we cannot make s_1 arbitrarily small, since then, as noted above, $B(s)$ approaches *N*, and the slope of the (log *s*, log $B(s)$) curve approaches 0. The study of the minimal and maximal box sizes has received a great deal of attention; see, e.g., the large number of references in Kenkel (2013).

> A recent paper by Kenkel (2013) introduces a simple but very useful probabilistic model to obtain a relationship be-

tween N and $s₁$. The model estimates the expected value of $B(s)$ as a function of *s* and d_B . Then $s₁$ is chosen to be the $E(s)$ as a random of s and a_B . Then s_1 is shown to see the most widely used fractal dimensions is the box value at which the "local slope" of the $(\neg \log s, \log B(s))$ curve deviates from d_B by no more than 0.001, where this accuracy was chosen since estimates of d_B are often expressed to three boxes of side length *s*. If the object is 2-di-
boxes of side length *s*. If the object is 2-di-
or four decimal places. Using 28 simulations in which *s* was boxes are squares; if the object is 3-dimen-
decreased until the local slope is within 0.001 of d_B , Kenkel obtained the following approximation for the minimal usable box size:

$$
s_1 \approx \left(\frac{N}{10}\right)^{-1/d_B} \tag{1}
$$

Since (1) was obtained numerically, and only for the accubunting dimension d_B of a geometric object is racy 0.001, this result provides no guidance on how s_1 varies maeutrot 1985) as a function of the accuracy. Indeed, in many applications, α is only estimated to the analysis of neurons (Karperien 2013), d is only e.g., in the analysis of neurons (Karperien 2013), d_B is only $\log B(s)$ estimated to two decimal places. Letting ε be the desired ac- $\log s$ curacy, in this paper we generalize Kenkel's result to compute $\log s$ s_1 for arbitrary ε . We show that s_1 can be computed, for any e, by standard one-dimensional search techniques such as bi log nary search.

The original model

he slope of this line is the estimate of d_B . Following Kenkel (2013), consider first the 1-dimensional case. If we randomly select *N* points on [0, 1] then the that the probability that a given box (i.e., interval) of size *s* does not arge as possible, so as to minimize the error in probability that a given box (i.e., interval) of size *s* does not From the *J* pairs (log s_j , log $B(s_j)$). However, we contain any of these points is $(1-s)^N$. The probability that a given box of size *s* contains at least one of *N* randomly se- \sum_{s} *S* columnary similar, the expected number *B*(*s*) \sum_{s} of $B(s)$ of $B(s)$ and the slope of the (log *s*, log *B*(*s*)) curve lected points is $1 - (1 - s)N$. Hence the expected number *B*(*s*) study of the minimal and maximal box of nonempty boxes of size s is given by $B(s) = [1 - (1 - s)^{N}]/s$.

led a great deal of attention, see, e.g., the large considering next the 2-dimensional case, if we randomly rences in Kenkel (2013). select *N* points on the unit square $[0, 1] \times [0, 1]$ then the probper by Kenkel (2013) introduces a simple but ability that a given box of size s does not contain any of these babilistic model to obtain a relationship be-
points is $(1 - s^2)^N$. The probability that a given box of size *s*

contains at least one of *N* randomly selected points is $1 - (1$ s^2 ^{*N*}. Hence the expected number *B*(*s*) of nonempty boxes of size *s* is given by *B*(*s*) = $[1 - (1 - s^2)^N]/s^2$. size *s* is given by $B(s) = [1 - (1 - s^2)^N]/s^2$.

Finally, consider the case where the *N* points are sampled from a self-similar fractal set with box counting dimension d_B . For example, *N* might be the total number of pixels in an image of a real-world fractal, such as a fern leaf. The expected number $B(s)$ of nonempty boxes of size *s* is given by $(Kenkel 2013)$ $(Kenkel 2013)$ E where the *N* points are sampled
tet with box counting dimension Combining

$$
B(s) = \frac{1 - \left(1 - s^{d_B}\right)^N}{s^{d_B}}.
$$
 (2) $\frac{d_B N \left(1 - s^{d_B}\right)}{\left(1 - s^{d_B}\right)}$

This equality is the basis for the analysis of the remainder of this paper this paper. analysis of the remainder of Denoting the right behavior of $\frac{1}{2}$ Denoting the right

From (2), Kenkel numerically computes the "local slope" $F(s) > -a_B + \varepsilon$, so such box singularity computes the "local slope" estimation of d_B . Thus s^* is the of the (log *s*, log $B(s)$) curve, where the "local slope" $m(s)$ first step tow of the (log s, log $B(s)$) curve, where the "local slope" $m(s)$ first step towards
is defined by $m(s) \equiv (\log B(s + \delta) - \log B(s))/(\log (s + \delta)$ first step towards - log *s*) for some small increment δ. Then, s_1 is the value for which $m(s) + d_B = 0.001$ and empirically s_1 is found to be well approximated using (1). From (1) we obtain $s_1{}^{d}B = 10/N$. We write α rather than $\alpha(s)$. The equation $F(s) = -d_B + \varepsilon$ can
Substituting this in (2) we have Substituting this in (2), we have be obtain s_1^{a} = 10/*I*v.
now be rewritten as $\sum_{i=1}^{n}$ substituting the substituting the substituting the substituting the substituting term in $\sum_{i=1}^{n}$ as a ferm leaf of number $\frac{1}{2}$ of $\frac{1}{2}$ of $\frac{1}{2}$ s is given by (Kenkel 2013) s) for some small increment δ . Then, s_1 \mathcal{F} , we have \mathbf{B} + \mathbf{C}

$$
B(s_1) = \frac{1 - (1 - (10/N))^N}{10/N}.
$$
\n
$$
\frac{d_B N \alpha - d_B s}{10 - \alpha (1 - \alpha)}
$$

which yields, for large *N*, the estimate $B(s_1) = N/10$. This groups are how size a should be used, in the estimation of d which yields, for large *N*, the estimate $B(s_1) = N/10$. This means no box size *s* should be used, in the estimation of d_B , for which $B(s) > N/10$ (Kenkel 2013). for which $B(s) > N/10$ (Kenkel 2013). $\frac{1}{2}$ which \mathbf{w} which yields, for large N, the estimate B(s1) \sim N/10. This means no box size s should be used, in the $B(s_1) = N/10$. T \mathbf{D} ge N, the estimate $B(s_1) = N/10$. This

The generalized model $\overline{\mathcal{O}}$ is replaced by a positive parameter $\overline{\mathcal{O}}$ estimation of dB, for which B(s) > N/10 (Kenkel 2013).

In this section, we generalize the original model by deter-
Define $v \equiv (\varepsilon/d_B)$ mining the minimal box size when the constant 0.001 described
 $\alpha s^{dB} = v(1-\alpha + \alpha s^{dB})$, which simplifies to $\alpha s^{dB}(\lambda - 1 - v) =$ in Section 2 is replaced by a positive parameter ε . Rather $\alpha s^{a} = v(1-\alpha + \alpha s^{a} s)$ than work with a local slope of the $(\log s, \log B(s))$ curve, than work with a local slope of the (log s, log $B(s)$) curve,
we work with an actual derivative, which can be obtained in
closed form Hsing the chain rule for derivatives we have $\frac{Nd_B}{d} - 1 = \left(\frac{1-\alpha}{s}\right) s^{-d_B}$. closed form. Using the chain rule for derivatives, we have than work with a local slope of the (log *s*, log *B*(*s*)) curve,
we work with an actual derivative, which can be obtained in Nd_B $(1-\alpha)$ $-d$ $\frac{d}{dx}$ we the chain rule for derivatives, we have derived as \mathbf{c}_1 closed form. Using the chain rule for derivatives, we have $\frac{d\mathbf{u}_B}{d\mathbf{v}_B} - 1 = \left(\frac{1-\alpha}{\alpha}\right) s^{-d_B}$. the definition of α α β (1) and α (1) an be of the (log s, log $B(s)$) curve,

$$
\frac{d \log B(s)}{d \log s} = \left(\frac{d \log B(s)}{ds}\right) \left(\frac{ds}{d \log s}\right) = \text{Finally, from (9) and the definition (6) of } \alpha, \text{ we obtain an implicit equation for } s^*:
$$
\n
$$
\left(\frac{d \log B(s)}{ds}\right) \left(\frac{d \log s}{ds}\right)^{-1} = \left(\frac{d \log B(s)}{ds}\right)s. \qquad \left(\left(1 - s^{d_B}\right)^{1 - N} - 1\right)s^{-d_B} = \epsilon^{-1} N d_B - 1. \qquad (10)
$$

From (2), taking the derivative of $\log B(s)$ we have $\lim_{t \to 0} (2)$, taking the derivative of $\log D(s)$ Γ rom (2), taking the derivative of $\log B(s)$ we have ive of $\log D(s)$ we have

$$
\frac{d \log B(s)}{ds} =
$$
\n
$$
\frac{d \log B(s)}{d s} =
$$
\n
$$
\frac
$$

$$
\frac{d_B N s^{-1} (1 - s^{d_B})^{N-1} - d_B s^{-d_B - 1} (1 - (1 - s^{d_B})^N)}{(1 - (1 - s^{d_B})^N) s^{-d_B}}
$$
\n
$$
\frac{(1 - (1 - s^{d_B})^N) s^{-d_B}}{\text{Combining (3) and (4) yields}}
$$

Combining (3) and (4) yields $\mathcal{B}(4)$ yields

an image of a real-world fractal, such as a fern leaf. The ex-
pected number *B*(*s*) of nonempty boxes of size *s* is given by

$$
B(s) = \frac{1 - (1 - s^{d_B})^N}{s^{d_B}}.
$$
(5)

$$
B(s) = \frac{1 - (1 - s^{d_B})^N}{s^{d_B}}.
$$
(6)

$$
B(s) = \frac{1 - (1 - s^{d_B})^N}{s^{d_B}}.
$$

s for the analysis of the remainder of Denoting the right hand side of (5) by $F(s)$, our task is to find the value *s** such that $F(s^*) = -d_B + \varepsilon$. For $s < s^*$ we have $F(s) > -d + \varepsilon$, so such box sizes should not be used in the *F*(*s*) > $-d_B + \varepsilon$, so such box sizes should not be used in the imerically computes the "local slope" estimation of d_B . Thus s^* is the minimal usable box size. As a $f^{(n)}(s)$ first step towards calculating s^* , define $\frac{1}{s}$

$$
+\delta
$$
\n
$$
\alpha \equiv (1 - s^{dB})^{N-1}.
$$
\n(6)

ement 6. Then, s_1 is the value
and empirically s_1 is found to be While α is actually a function of s, for notational simplicity and empiricantly s_1 is found to be
 \therefore We write α rather than α(*s*). The equation $F(s) = -d_B + \varepsilon$ can \therefore From (1) we obtain s_1 ^{dB} = 10/*N*. ϵ ^l ω ²

$$
B(s_1) = \frac{1 - (1 - (10/N))^N}{10/N}.
$$
\n
$$
\frac{d_B N\alpha - d_B s^{-d_B} \left(1 - \alpha (1 - s^{d_B})\right)}{\left(1 - \alpha (1 - s^{d_B})\right) s^{-d_B}} + d_B = \epsilon.
$$
\n(7)

Dividing both sides by d_B yields Dividing by density $\overline{\text{Div}}$

which yields, for large *N*, the estimate
$$
B(s_1) = N/10
$$
. This
means no box size *s* should be used, in the estimation of d_B ,
for which $B(s) > N/10$ (Kenkel 2013).

$$
\frac{N\alpha - s^{-d_B} \left(1 - \alpha(1 - s^{d_B})\right)}{\left(1 - \alpha(1 - s^{d_B})\right) s^{-d_B}} = \frac{\epsilon}{d_B} - 1.
$$
(8)

Define $v \equiv (\varepsilon/d_B) - 1$. Multiplying the numerator and denomiperme v = $(\omega u_B)^{-1}$. Manaprying the numerator and denominator of the left hand side of (8) by s^{dB} yields $N\alpha s^{dB}$ = 1+ α = when the constant 0.001 described $\alpha s^{dB} = v(1-\alpha + \alpha s^{dB})$, which simplifies to $\alpha s^{dB}(N-1-v)=(v+1)$
a positive parameter ε . Rather $(1-\alpha)$ Becalling the definition of v more algebra vialds $D_{\mathcal{A}}(S_{\mathcal{A}}) = (dD) - 1$. Multiplying the numerator and denominator of \mathcal{A} e generalize the original model by deter-
Define $v \equiv (\epsilon/d_B) - 1$. Multiplying the numerator and denomi-− 1 . (8)
− 1 . (8)

of the (log s, log *B(s)*) curve,
itive, which can be obtained in
rule for derivatives, we have
$$
\frac{Nd_B}{\epsilon} - 1 = \left(\frac{1-\alpha}{\alpha}\right)s^{-d_B}.
$$
 (9)

Finally, from (9) and the definition (6) of α , we obtain an $\begin{aligned} \text{(a) } \text{log } s & \text{?} \end{aligned}$ $\text{implicit equation for } s^*$: $\frac{1}{\sqrt{2}}$ $\frac{1}{4}$ $\frac{1}{4}$ $\frac{1}{4}$ $\frac{1}{4}$ $\frac{1}{4}$

$$
\left(\frac{l \log s}{d s}\right) = \left(\frac{d \log B(s)}{d s}\right) s \tag{1 - s^{d_B} - 1} s^{-d_B} = \epsilon^{-1} N d_B - 1. \tag{10}
$$

vative of $\log B(s)$ we have Equation (10) is exact; no approximations were made. The *Equation (10)* is exact, no approximations were made. The parameter ε appears only in the right hand side, which is in- $\frac{1}{2}$ dependent of *s*. parameter ε appears only in the right hand side, which is in-
dependent of s $dependent of s.$ dependent of s .

EXECUTE: We compute a solution s^* of (10) using binary search over [0,1]; a code listing is provided in the Appendix. The binary we compute a solution s^* or (10) using binary search over

[0,1]; a code listing is provided in the Appendix. The binary $\left(4\right)$ [0,1]; a code listing is provided in the Appendix. The binary The binary search is halted when the width of the interval containing s is less than 1.0× 10[−]⁷, which takes

 ≤ 2 .

search is halted when the width of the interval containing s^*
 $\approx x^{-1}(1 + Nx + (N+1)2^{10-7}x)$ search is halted when the width of the interval containing s^* $\approx x^{-1} \left(1 + Nx + (N^2/2)x^2 + (N^3/6)x^3 - 1\right)$
is less than 1.0×10⁻⁷, which takes 24 iterations. For $N = 1000$ and $\varepsilon = 0.001$, the preceding to the state of $\sinh(\varepsilon)$ is the extended of $\sinh(\varepsilon)$ of $N + (N^2/2)x + (N^3/6)x^2$ estimate defined by (1), is plotted in Figure 1 as a function of d_B for $1 \le d_B \le 2$. The error in the approximation (1) is highest
(about 10%) when $d_B = 1$ (about 10%) when $d_B = 1$. $\begin{array}{rcl} \epsilon \text{ error in the approximation (1) is higher.} \end{array} = \epsilon^{-1} N d_B - 1 \approx \epsilon^{-1} N d_B.$ example defined by (1), is plotted in Figure 1 as a function of
estimate defined by (1), is plotted in Figure 1 as a function of
 \int for $1 \le d \le 2$. The error in the entropyimation (1) is highert \overline{D} and \overline{D} x[−]¹ 1 − x ¹−^N [−] ¹

Since, as noted in Section 1, in many applications d_B is $x \to b$ to the section of th Since, as noted in Section 1, in many applications d_B is
estimated to only two decimal places, Figure 2 compares, for Dividing by N, we obtain (N
 $N=1000$ the value s* for $s=0.1$ $s=0.01$ and $s=0.001$ all for Using 1 *N* =1000, the value *s*^{*} for ε =0.1, ε =0.01, and ε =0.001, all for Using $1 - \varepsilon^{-1}d_B \approx -\varepsilon^{-1}d_B$ we get
the same range $1 \le d_B \le 2$. the same range $1 \le d_B \le 2$.
 $-\frac{N}{2} \pm \sqrt{\frac{N^2}{4} - 4\frac{N^2}{6}(-\epsilon^{-1}d_D)}$ $N\sqrt{\frac{2}{3}\epsilon^{-1}d_D}$ $\overline{101}$ $N = 1000$ the value s^{*} for $\varepsilon = 0.1$ and ϵ

Approximating the minimal usable box size 4 Approximating the Minimal Usable Box Size

We now derive a closed form approximation to s^* .
Defining $x = s^{dg}$, we rewrite (10) as $= \frac{1}{N}$ \mathbf{D} $\frac{1}{N}$ = $\frac{1}{N}$

$$
x^{-1}\left(\left(1-x\right)^{1-N}-1\right)=\epsilon^{-1}Nd_B-1.
$$
 (11) By definition, $x = s^{dB}$, so $s^{dB} = (1/N)\sqrt{(6d_B/\epsilon)}$. Solving for *s* yields an approximation \tilde{s} for the minimal usable box size:

We have $x < 1$ since $s < 1$ and $d_B > 0$. Ignoring terms of de-
organization of (1−x)1−N $\widetilde{s} = \left(N \sqrt{\frac{6}{6d}} \right)$ gree four and higher, the Taylor series expansion of $(1-x)^{1-N}$
gree four and higher, the Taylor series expansion of $(1-x)^{1-N}$
since $(1-x)^{1-N}$ yields $(1-x)^{1-N} \approx 1 - (1-N)x + (1-N)(-N)x^2/2 - (1-N)(-N)$
(N 1)x²/6. For large N we have $(1-x)^{1-N} \approx 1 + Nx + (2N/2)x^2$. The estimate \tilde{s} which is a decreasing function $(-N-1)x^{3/6}$. For large *N* we have $(1-x)^{1-N} \approx 1 + Nx + (N^{2}/2)x^{2}$ $+(N³/6)x³$. Substituting this in (11) yields identical to (1), except to (1) is replaced box size: (1) is replaced by $s \le 1$ and $d_B > 0$. Ignoring terms of de Expansion of $(1-x)^{1-N}$

$$
x^{-1}\left(\left(1-x\right)^{1-N}-1\right)
$$

en the width of the interval containing
$$
s^*
$$

\n7, which takes 24 iterations. For $N = 1000$
\n $\left(\frac{1}{N} + Nx + \frac{N^2}{2}x^2 + \frac{N^3}{6}x^3 - 1\right)$
\n $\left(\frac{1}{N}, \frac{1}{N}\right)^* / s^*$, where s_1 is the
\n(1), is plotted in Figure 1 as a function of
\ne error in the approximation (1) is highest
\n
$$
= \epsilon^{-1}Nd_B - 1 \approx \epsilon^{-1}Nd_B.
$$

Dividing by *N*, we obtain $(N^2/6)x^2 + (N/2)x + (1 - \varepsilon^{-1}d_B) \approx 0$.
Using $1 - \varepsilon^{-1}d_B \approx -\varepsilon^{-1}d_B$ we get $= 0.01$, and $\varepsilon = 0.001$, all for Using $1 - \varepsilon^{-1} d_B \approx -\varepsilon^{-1} d_B$ we get gure 2 compares, for Dividing by N, we obtain $(N^2/6)x^2 + (N/2)x + (1 - \varepsilon^{-1}d_B) \approx 0$.
and $\varepsilon = 0.001$ all for Using $1 - \varepsilon^{-1}d_B \approx -\varepsilon^{-1}d_B$ we get 1, in many applications d_B is
laces, Figure 2 compares, for Dividing by N, we obtain $(N^2/6)x^2 + (N/2)x + (1 - \varepsilon^{-1}d_B) \approx 0$. [−]¹N d^B [−] ¹ [≈] −
−1N dB .

$$
\begin{aligned}\n\text{Approximating the minimal usable box size} \\
\text{We now derive a closed form approximation to } s^*. \\
\text{Defining } x = s^{dg}, \text{ we rewrite (10) as}\n\end{aligned}\n\quad\n\begin{aligned}\nx = \frac{-\frac{N}{2} \pm \sqrt{\frac{N^2}{4} - 4 \frac{N^2}{6} (-\epsilon^{-1} d_B)}}{(2/6) N^2} \approx \frac{N \sqrt{\frac{2}{3} \epsilon^{-1} d_B}}{(1/3) N^2} \\
\text{We now derive a closed form approximation to } s^*. \\
\text{Defining } x = s^{dg}, \text{ we rewrite (10) as}\n\end{aligned}
$$

B

 $\frac{1}{4}$

by definition, $x = s\omega$, so $s\omega = (1/y)(0dy/\epsilon)$. Solving for spields an approximation \tilde{s} for the minimal usable box size: $\frac{1}{2}$. (i) N(σa_B / ϵ). Solving for *s*
inimal usable box size: $\sum_{i=1}^{n}$

We have
$$
x < 1
$$
 since $s < 1$ and $d_B > 0$. Ignoring terms of de-
gree four and higher, the Taylor series expansion of $(1-x)^{1-N}$
yields $(1-x)^{1-N} \approx 1 - (1-N)x + (1-N)(-N)x^2/2 - (1-N)(-N)$ (12)

 $(1-N)(-N)$
 $x + (N^2/2)x^2$ The estimate \tilde{s} , which is a decreasing function of ε , looks identical to (1) , except that the empirically derived constant 0.1 in (1) is replaced by $\sqrt{\left(\varepsilon/(\delta d_B)\right)}$ in (12). However, \tilde{s}
is not a good approximation to κ^* . Each sympals with N is not a good approximation to s^* . For example, with $N = 1000$, $l = 2$, and $s = 0.001$ us have $\epsilon^* \approx 0.000$, while $= 1000, d_B = 2$, and $\varepsilon = 0.001$, we have $s^* \approx 0.099$ while

 $\widetilde{s} \approx 0.331$. The error arises from the third order Taylor series **Appendix: Bin** approximation, since for $x = (s^*)^2$ we have $(1-x)^{1-N} \approx 19700$ while $1 + Nx + (N^2/2)x^2 + (N^3/6)x^3 \approx 219$. Thus, while (12) yields the functional form obtained by Kenkel, in practice *s** 5 should be computed using (10) and binary search.

Concluding remarks

We generalized Kenkel's model for finding the minimal usable box size *s** for computing the box counting dimension. For box sizes smaller than *s**, the slope of the (−log *s*, log *B*(*s*)) curve flattens out, and deviates by more than a specified accuracy ε from d_B . Whereas Kenkel considered only the choice $\varepsilon = 0.001$, we derived an exact implicit equation for *s** for any e. Binary search was used to actually compute *s**, and a Python implementation is provided in the Appendix. We also used the implicit equation to derive a closed form approximation for *s** having the same functional form as Kenkel's empirically obtained expression.

Acknowledgments. The author thanks N. Kenkel and an anonymous reviewer for their valuable comments and suggestions.

References

- Falconer, K. 2003. *Fractal Geometry: Mathematical Foundations and Applications*, 2nd edn. Wiley, Chichester.
- Karperien A., H. Ahammer and H.F. Jelinek. 2013. Quantitating the subtleties of microglial morphology with fractal analysis. *Frontiers in Cellular Neuroscience* 7: 3, doi: 10.3389/fncel.2013.00003.
- Kenkel, N.C. 2013. Sample size requirements for fractal dimension estimation. *Community Ecol.* 14: 144–152.
- Mandelbrot, B.B. 1983. *The Fractal Geometry of Nature*. W.H. Freeman, New York.

Received November 3, 2015 Revised December 20, 2015 Accepted January 26, 2016

Appendix: Binary search code

Python code used to implement the binary search method used in Section 3 to compute the *s** solving (10). The code performs binary search over the interval [0, 1] to compute s* to an accuracy of 10^{-7} for $\varepsilon = 0.001$ and for d_B from 1 to 2, in increments of 1/200. The variable Estimate is the estimate given by (1).

