Chapter 14
On the Fractal Distribution of Brain Synapses

Richard Crandall†

Abstract Herein we present mathematical ideas for assessing the fractal character of distributions of brain synapses. Remarkably, laboratory data are now available in the form of actual three-dimensional coordinates for millions of mouse-brain synapses (courtesy of Smithlab at Stanford Medical School). We analyze synapse datasets in regard to statistical moments and fractal measures. It is found that moments do not behave as if the distributions are uniformly random, and this observation can be quantified. Accordingly, we also find that the measured fractal dimension of each of two synapse datasets is 2.8 ± 0.05. Moreover, we are able to detect actual neural layers by generating what we call probagrams, paramegrams, and fractagrams—these are surfaces one of whose support axes is the y-depth (into the brain sample). Even the measured fractal dimension is evidently neural-layer dependent.

Key words: Brain • Fractals • Neural science • Synapses

Mathematics Subject Classifications (2010): Primary 28Axx, 28A80, 28A25; Secondary 65Cxx, 65C20.

COMMUNICATED BY HEINZ H. BAUSCHKE.

R. Crandall† (1947–2012) (✉)

D.H. Bailey et al. (eds.), *Computational and Analytical Mathematics*, Springer Proceedings in Mathematics & Statistics 50, DOI 10.1007/978-1-4614-7621-4_14,

14.1 Motivation

Those who study or delight in fractals know full well that often the fractal nature is underscored by structural rules. When the author was informed by colleagues[1] that 3D synapse data is now available in numerical form, it loomed natural that mathematical methods should be brought to bear.

Thus we open the discussion with the following disclaimer: The present paper is not a neurobiological treatise of any kind. It is a mathematical treatise. Moreover, there is no medical implication here, other than the possibility of using such measures as we investigate for creation of diagnostic tools.[2]

There is some precedent for this kind of mathematical approach. Several of many fractal studies on neurological structures and signals include [8–10]. on random point-clouds per se have even been suggested for the stringent testing of random-number generators [7]. Some researchers have attempted to attribute notions of context-dependent processing, or even competition to the activity within neural layers [1]. Indeed, it is known that dendrites—upon which synapses subsist—travel through layers. Some good rendition graphics are found in [16]. Again, our input datasets do not convey any information about dendritic structure; although, it could be that deeper analysis will ultimately be able to suggest dendritic presence [17].

14.2 Synapse Data for Mathematical Analysis

Our source data is in the section Appendix: Synapse datasets. It is important to note that said data consists exclusively of triples (x, y, z) of integers, each triple locating a single brain synapse, and we rescale to nanometers to yield physically realistic point-clouds. There is no neurological structure per se embedded in the data. This lack of structural information actually allows straightforward comparison to random point-clouds (Fig. 14.1).

To be clear, each synapse dataset has the form

$$x_0 \;\; y_0 \;\; z_0$$

$$x_1 \;\; y_1 \;\; z_1$$

[1] From Smithlab, of Stanford Medical School [15].

[2] Indeed, one motivation for high-level brain science in neurobiology laboratories is the understanding of such conditions as Alzheimer's syndrome. One should not rule out the possibility of "statistical" detection of some brain states and conditions—at least, that is our primary motive for bringing mathematics into play.

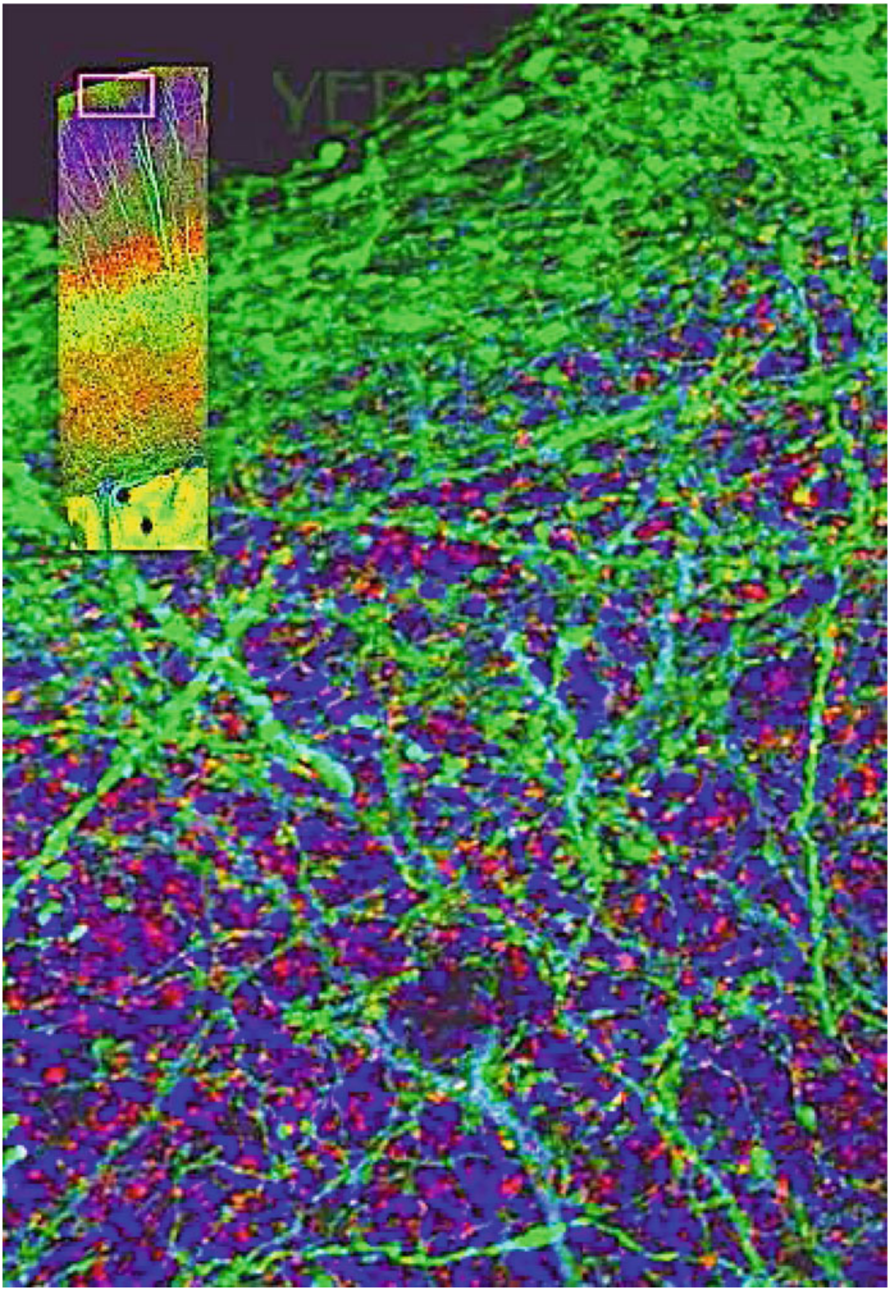

Fig. 14.1 Frame from video: The beginning (*top layer*, $y \sim 0$) of a mouse-brain section. Synapses (our present data of interest) are *red points*. The *vertical strip at upper left* represents the complete section—the *small light-pink rectangle* indicates the region we are currently seeing in the video (courtesy of Smithlab, Stanford medical school [15])

$$\cdots$$

$$x_j \;\; y_j \;\; z_j \qquad (= \mathbf{r})$$

$$\cdots$$

$$x_k \;\; y_k \;\; z_k \qquad (= \mathbf{q})$$

$$\cdots$$

$$x_{N-1} \;\; y_{N-1} \;\; z_{N-1},$$

where each x, y, z is an integer (Appendix 1 gives the nanometer quantization). There are N points, and we have indicated symbolically here that we envision some row as point $\mathbf{r}$ and some other row as point $\mathbf{q}$, for the purposes of statistical analysis (Fig. 14.2). (A point $\mathbf{r}$ may or may not precede a $\mathbf{q}$ on the list, although in our calculations we generally enforce $\mathbf{r} \neq \mathbf{q}$ to avoid singularities in some moments.)

14.3 The Modern Theory of Box Integrals

Box integrals—essentially statistical expectations, also called moments, over a unit box rather than over all of space—have a rich, decades-long history (see [2,3,5] and historical references therein). The most modern results involve such functions as

$$\Delta_n(s) \;:=\; \langle|\mathbf{r}-\mathbf{q}|\rangle|_{\mathbf{r},\mathbf{q}\in[0,1]^n}$$

$$= \int_0^1 \cdots \int_0^1 \left(\sum_{k=1}^{n} (r_k - q_k)^2 \right)^{s/2} \mathrm{d}r_1 \mathrm{d}q_1 \mathrm{d}r_2 \mathrm{d}q_2 \cdots \mathrm{d}r_n \mathrm{d}q_n.$$

This can be interpreted physically as the expected value of v^s, where separation $v = |\mathbf{v}|$, $\mathbf{v} := \mathbf{r} - \mathbf{q}$ is the distance between two uniformly random points each lying in the unit n-cube (Fig. 14.3).

It is of theoretical interest that $\Delta_n(s)$ can be given a closed form for every integer s, in the cases $n = 1,2,3,4,5$ [5]. For example, the expected distance between two points in the unit 3-cube is given exactly by

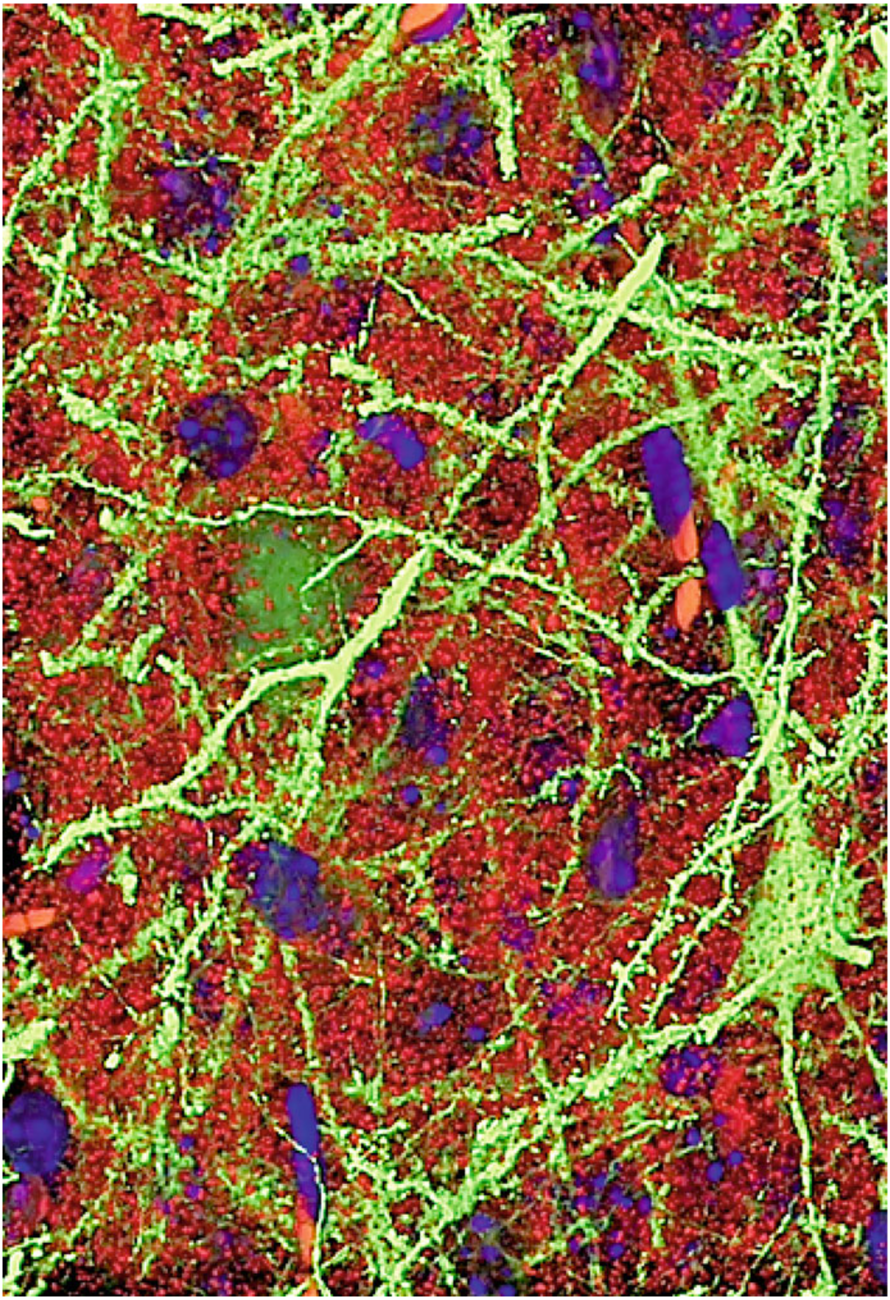

Fig. 14.2 A subsection in neural layer 5b. The chemical color-coding is as follows. *Green*: Thy1-H-YFP (layer 5B neuron subset); *Red*: Synapsin I (synapses); *Blue*: DAPI (DNA in all nuclei). All of our present analyses involve only the synapsin-detected synapses

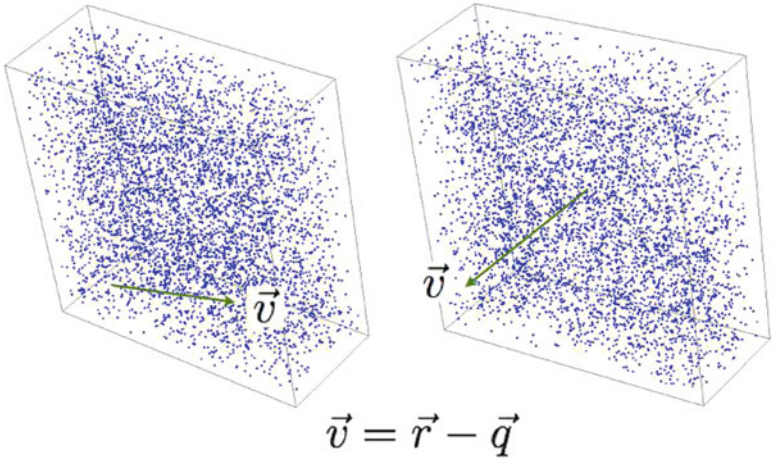

Fig. 14.3 Views of 5,000 random points (*left*) and 5,000 actual synapses (*right*) in a cuboid of given sides as follows (all in nanometers): $a = \Delta x \sim 103,300; b = \Delta y \sim 78,200; c = \Delta z \sim 11,400$, for *horizontal*, *vertical*, and *transverse* (angled into page), respectively. To convey an idea of scale, a millimeter is about 10x the horizontal span of either point-cloud. It is hard to see visual differences between the random points at *left* and the actual brain points at *right*. Nevertheless, sufficiently delicate statistical measures such as moments $\langle |\mathbf{v}|^s \rangle$ as well as fractal measurement do reveal systematic, quantifiable differences

$$\begin{aligned}\Delta_3(1) &= -\frac{118}{21} - \frac{2}{3}\pi + \frac{34}{21}\sqrt{2} - \frac{4}{7}\sqrt{3} + 2\log\left(1+\sqrt{2}\right) + 8\log\left(\frac{1+\sqrt{3}}{\sqrt{2}}\right) \\ &= 0.6617071822671762351558311332484135817464001357909 5\ldots.\end{aligned}$$

The exact formula allows a comparison between a given point-cloud and a random cloud: One may calculate the empirical expectation $\langle |\mathbf{r}-\mathbf{q}| \rangle$, where $\mathbf{r}, \mathbf{q}$ each runs over the point-cloud and compares with the exact expression $\Delta_3(1) \approx \ldots$. Similarly it is known that the expected inverse separation in the 3-cube is

$$\Delta_3(-1) := \left\langle \frac{1}{|\mathbf{r}-\mathbf{q}|} \right\rangle$$

$$= \frac{2}{5} - \frac{2}{3}\pi + \frac{2}{5}\sqrt{2} - \frac{4}{5}\sqrt{3} + 2\log\left(1+\sqrt{2}\right) + 12\log\left(\frac{1+\sqrt{3}}{\sqrt{2}}\right) - 4\log\left(2+\sqrt{3}\right)$$

$$= 1.8823126443896601601056008388683675878524628803107 0\ldots.$$

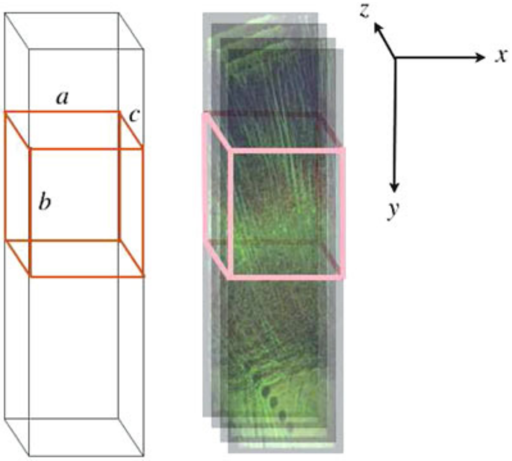

Fig. 14.4 Pictorial of the role of cuboid calculus in our analysis scenario. The *right-hand entity* pictorializes an array-tomography section of mouse brain (see Appendix: Synapse datasets for details). At the *left* is an idealized, *long cuboid* representing the full brain sample, inside of which is a chosen subsection as an (a,b,c)-cuboid. The idea is to statistically compare the synapse distribution within an (a,b,c)-cuboid against a random distribution having the same cuboid population. By moving the (a,b,c) cuboid downward, along the y-axis, one can actually detect neural layers

Such exact forms do not directly apply in our analysis of the brain data, because we need volume sections that are not necessarily cubical. For this reason, we next investigate a generalization of box integrals to cuboid volumes (Fig. 14.4).

14.4 Toward a Theory of Cuboid Integrals

In the present study we shall require a more general three-dimensional box integral involving a cuboid of sides (a,b,c).[3] Consider therefore an expectation for two points $\mathbf{r},\mathbf{q}$ lying in the same cuboid (Fig. 14.5):

$$\begin{aligned}\Delta_3(s;a,b,c) \;&:=\; \langle|\mathbf{r}-\mathbf{q}|\rangle|_{\mathbf{r},\mathbf{q}\in[0,a]\times[0,b]\times[0,c]}\\ &= \frac{1}{a^2b^2c^2}\int_0^a\int_0^a\int_0^b\int_0^b\int_0^c\int_0^c |\mathbf{r}-\mathbf{q}|^s\,\mathrm{d}r_1\,\mathrm{d}q_1\,\mathrm{d}r_2\,\mathrm{d}q_2\,\mathrm{d}r_3\,\mathrm{d}q_3.\end{aligned}$$

This agrees with the standard box integral $\Delta_3(s)$ when $(a,b,c)=(1,1,1)$.

[3] A cuboid being a parallelepiped with all faces rectangular—essentially a "right parallelepiped."

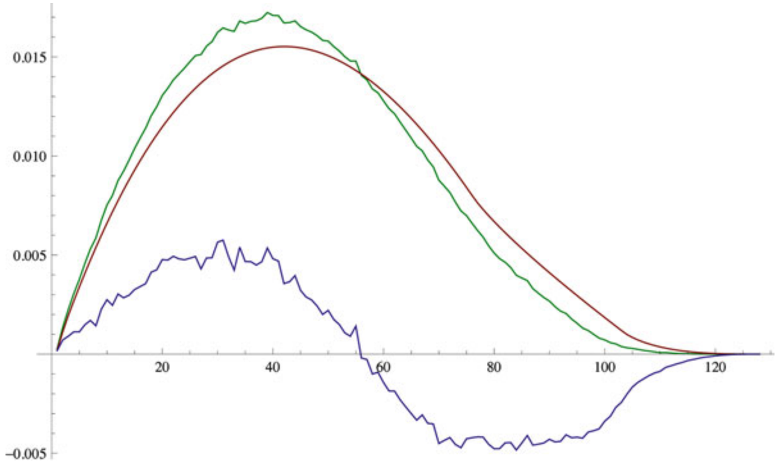

Fig. 14.5 Probability density curves for the separation $v = |\mathbf{r} - \mathbf{q}|$ (*horizontal axis*), taken over a cuboid of data, in the spirit of Fig. 14.4. The *green curve* (with highest peak) is extracted from subsegment 2 of dataset I, under the segmentation paradigm $\{12, 1, 128, \{1, 128\}\}$. The *red curve* (with rightmost peak) is theoretical—calculated from the Philip formula for $F_3(v; 146700, 107900, 2730)$. The *blue* "excess curve" is the point-wise curve difference (amplified $3\times$) and can be used in our "probagram" plots to show excess as a function of section depth y. The expected separations within this cuboid turn out to be $\langle v \rangle = 62018, 66789$ for brain, random, respectively

Figure 14.6 shows the result of empirical assessment of cuboid expectations for dataset I.

We introduce a generalized box integral, as depending on fixed parameters k, a_1, a_2, a_3 (we use a_i here rather than a, b, c just for economy of notation):

$$G_3(k; a_1, a_2, a_3) \;:=\; \langle \mathrm{e}^{-k|\mathbf{p}-\mathbf{q}|^2} \rangle$$

$$= \frac{1}{\prod a_i^2} \int_0^{a_1} \int_0^{a_1} \cdots \int_0^{a_3} \int_0^{a_3} \mathrm{e}^{-k|\mathbf{r}-\mathbf{q}|^2} \, \mathrm{d}r_1 \, \mathrm{d}r_2 \, \mathrm{d}r_3 \, \mathrm{d}q_1 \, \mathrm{d}q_2 \, \mathrm{d}q_3,$$

which, happily, can be given a closed form

$$G_3(k; a_1, a_2, a_3) \;=\; \frac{1}{k^3} \prod_i \frac{\mathrm{e}^{-a_i^2 k} + a_i \sqrt{\pi k} \, \mathrm{erf}\left(a_i \sqrt{k}\right) - 1}{a_i^2},$$

where $\mathrm{erf}(z) := 2/\sqrt{\pi} \int_0^z \mathrm{e}^{-t^2} \, \mathrm{d}t$ denotes the error function. The closed form here is quite useful, and by expanding the erf() in a standard series, we obtain for example a three-dimensional summation for G_3. The question is, can one write a summation that is of lower dimension? One possible approach is to expand the Gaussian in even

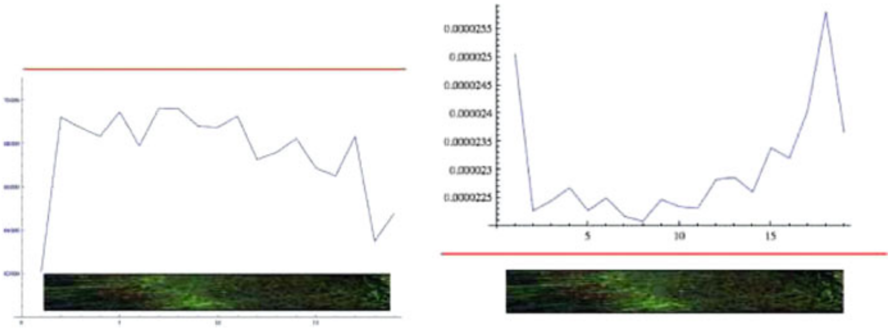

Fig. 14.6 Results for cuboid expectations of separation v and $1/v$ for cuboids of the type in Fig. 14.4, running over all y-depth. (The *dark green horizontal strip* represents the full sample, oriented *left-right* for these plots.) In both *left- and right-hand plots*, the *horizontal red line* is calculated from the exact formula for $\Delta_3(1;a,b,c)$. The segmentation paradigm here is $\{12,2,80,\{1,32\}\}$, dataset I

powers of $|\mathbf{p}-\mathbf{q}|$ and leverage known results in regard to box integrals Δ_n of Bailey et al. [2, 3]. Such dimensionality reduction remains an open problem.

Yet another expectation that holds promise for point-cloud analysis is what one might call a Yukawa expectation:

$$Y_3(k;a_1,a_2,a_3) \ := \left\langle \frac{\mathrm{e}^{-k|\mathbf{r}-\mathbf{q}|}}{|\mathbf{r}-\mathbf{q}|} \right\rangle.$$

This is the expected Yukawa potential—of nuclear physics lore—between two points within the cuboid. The reason such potentials are of interest is that being "short-range" (just like nuclear forces) means that effects of closely clustered points will be amplified. Put another way: The boundary effects due to finitude of a cuboid can be rejected to some degree in this way.

14.4.1 Cuboid Statistics

Not just the exact expectation $\Delta_3(1;a,b,c)$ but the very probability density $F_3(v;a,b,c)$ has been worked out by Philip [14]. Both exact expressions in terms of a,b,c are quite formidable—see Appendix: Exact-density code for a programmatic way to envision the complexity. By probability density, we mean

$$\mathrm{Prob}\{|\mathbf{r}-\mathbf{q}| \in (v,v+\mathrm{d}v)\} \ = \ F_3(v;a,b,c)\,\mathrm{d}v;$$

hence we have a normalization integral with upper limit being the long cuboid diagonal:

$$\int_0^{\sqrt{a^2+b^2+c^2}} F_3(v;a,b,c)\,\mathrm{d}v \;=\; 1.$$

More generally we can represent the moment Δ_3 in the form

$$\Delta_3(s;a,b,c) \;=\; \int_0^{\sqrt{a^2+b^2+c^2}} v^s\, F_3(v;a,b,c)\,\mathrm{d}v.$$

The Philip density for separation v can also be used directly to obtain the density for a power of v, so

$$f_3(X := v^s;a,b,c) \;=\; \frac{1}{|s|} X^{\frac{1}{s}-1} F_3(X^{\frac{1}{s}};a,b,c).$$

For example, if we wish to plot the density of inverse separation $X := 1/v$ for a random point-cloud, we simply plot $X^{-2}F_3(1/X;a,b,c)$ for X running from $1/\sqrt{a^2+b^2+c^2}$ up to infinity; the area under this density will be 1.

14.5 Fractal Dimension

For the present research we used two fractal-measurement methods: The classical box-counting method, and a new, space-fill method. For a survey of various fractal-dimension definitions, including estimates for point-cloud data, see [6].

As for box-counting, we define a box dimension

$$\delta \;:=\; \lim_{\varepsilon\to 0} \frac{\log \#(\varepsilon)}{-\log \varepsilon}\ ,$$

where for a given side ε of a microbox, $\#(\varepsilon)$ is the number of microboxes that minimally, collectively contain all the points of the point-cloud. Of course, our clouds are always finite, so the limit does not exist. But it has become customary to develop a #-vs.ε curve, such as the two curves atop Fig. 14.7, and report in some sense "best slope" as the measured box dimension.

There are two highly important caveats at this juncture: We choose to *redefine* the box-count number, as

$$\# \;\to\; \#\cdot \frac{1}{1-\mathrm{e}^{-N\varepsilon^3}},$$

when the cloud has N total points. This statistical warp factor attempts to handle the scenario in which microboxes are so small that the finitude of points causes many empty microboxes. Put another way: The top curve of the top part of

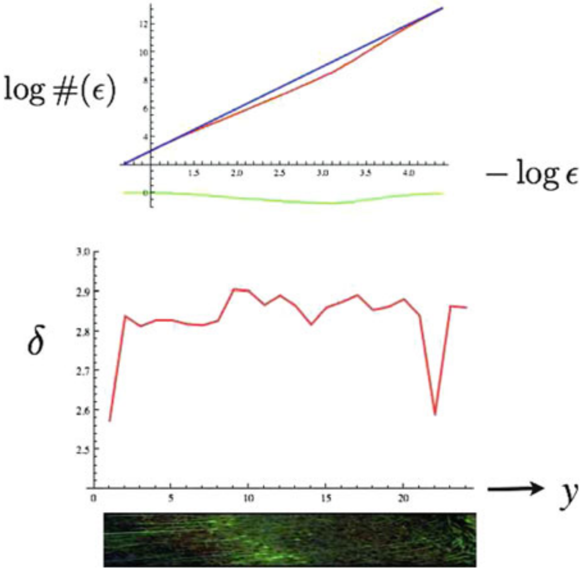

Fig. 14.7 Fractal-dimension measurement. Within a given cuboid we use the standard box-counting method, namely, in the *upper* figure is plotted log# vs. $\log(1/\varepsilon)$ for random points (*upper, blue curve*), then for the actual synapse points (*lower, red curve*), and with the excess as the *green* (*lowest*) plot. In the *bottom* figure, we use the excess to estimate fractal dimension for each cuboid in a segmentation paradigm $\{12, 2, 80, \{2, 80\}\}$. Evidently, the fractal dimension fluctuates depending on layer characteristics at depths *y*, with an average fractal dimension of ~ 2.8 for the whole of dataset I

Fig. 14.7—which curve should have slope 3 for N random points—stays straight and near slope 3 for a longer dynamic range because of the warp factor.

The second caveat is that we actually use not ε-microboxes but microcuboids. When the segment being measured is originally of sides (a, b, c), we simply rescale the cuboid to be in a unit box, which is equivalent to using a "microbrick" whose aspect ratios are that of the cuboid, and transform that microbrick to a cube of side $\varepsilon := (abc)^{1/3}$.

14.5.1 *Space-Fill Method for Fractal Measurement*

During this research, we observed that a conveniently stable fractal-measurement scheme exists for point-cloud datasets. We call this method the "space-fill" algorithm, which runs like so[4]:

[4] The present author devised this method in 1997, in an attempt to create "$1/f$" noise by digital means, which attempt begat the realization that fractal dimension could be measured with a Hilbert space-fill.

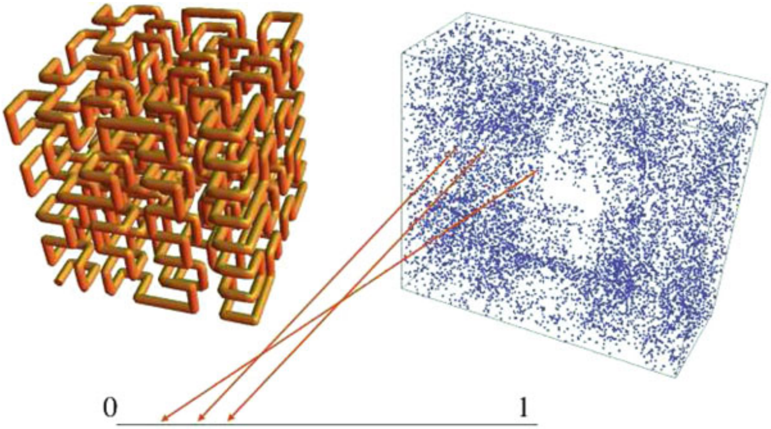

Fig. 14.8 The "space-fill" method for measuring point-cloud dimension. This algorithm as described in the text yields similar results to the more standard box-counting method, yet preliminary research reveals the space-fill method to be rather more stable with respect to graph noise. The basic idea is to create a set of pullbacks on the line $[0,1)$ and then use a quick sort and a simple one-dimensional fractal assessment

1. Assume a three-dimensional unit cube containing a point-cloud and construct a Hilbert space-filling curve, consisting of discrete visitation points $\mathbf{H}(t)$, where t runs over the integers in $[0, 2^{3b} - 1]$. (The resolution of this curve will be b binary bits per coordinate, therefore.)
2. Create a list of "pullback" rationals $t_k/2^{3b}$, corresponding to the points r_k of the point-cloud data.
3. Perform a one-dimensional sort on the set of pullbacks and measure the fractal dimension δ_1 using a simple interval counter.
4. Report fractal dimension of the point-cloud data as $\delta = 3 \cdot \delta$.

We do not report space-fill measurements herein—all of the results and figures employ the box-counting method—except to say (a) the space-fill method appears to be quite stable, with the fractagram surfaces being less noisy, and (b) the dimensions obtained in preliminary research with the space-fill approach are in good agreement with the box-counting method. Figure 14.8 pictorializes the space-fill algorithm.

14.6 Probagrams, Paramegrams, and Fractagrams

Our "grams" we have so coined to indicate their three-dimensional-embedding character.[5] Each 'gram is a surface, one of whose support dimensions is the section

[5] As in "sonogram"—which these days can be a medical ultrasound image, but originally was a moving spectrum, like a fingerprint of sound that would fill an entire sheet of strip-chart.

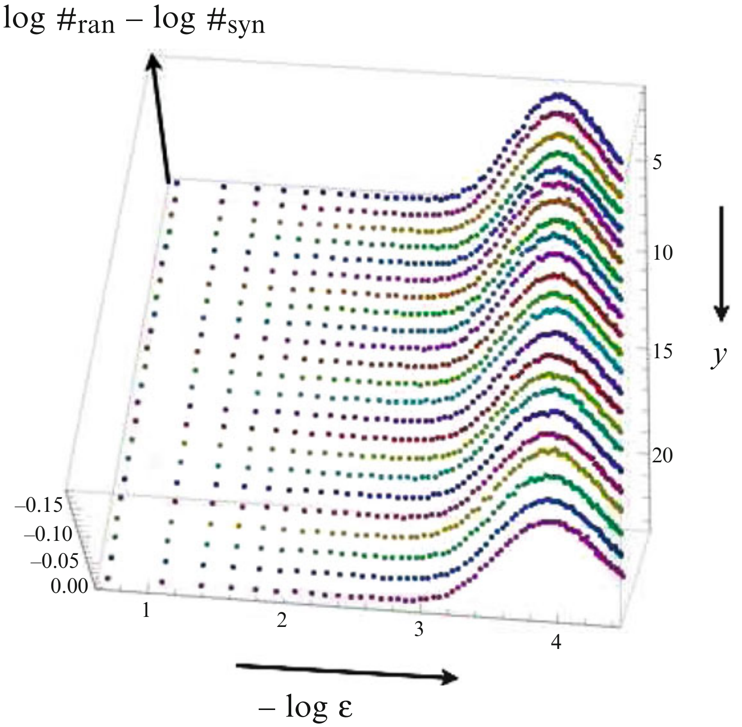

Fig. 14.9 The "fractagram" concept—which is similar for probagrams and paramegrams. For each cuboid in a given segmentation paradigm (here, paradigm $\{12, 2, 80, \{2, 80\}\}$) we generate the fractal-slope excess as in Fig. 14.7. The resulting "strands" of fractal data vs. y-depth in the dataset (here, dataset I) are much easier to interpret if plotted as a surface, which surface we then call a fractagram as pictured in Fig. 14.11

depth y. In our "grams", as in the original synapse datasets, $y = 0$ is the outside (pial) surface, while y increases into the brain sample. We typically have, in our "grams", *downward increasing* y, so that the top of a 'gram pictorial is the outside surface.

Precise definitions are:

- Probagram: Surface whose height is probability density of a given variable within a cuboid, horizontal axis is the variable, and the vertical axis is the y-depth into the sample.
- Paramegram: Surface whose height is a parameterized expectation (such as our function $G_3(k; a, b, c)$), horizontal axis is the parameter (such as k), and the vertical axis is y-depth.
- Fractagram: Surface whose height is the excess between the fractal-slope curve for a random cloud in a cuboid and the actual data cloud's fractal-slope curve, horizontal axis is $-\log \varepsilon$, and vertical axis is as before the y-depth (Fig. 14.9).

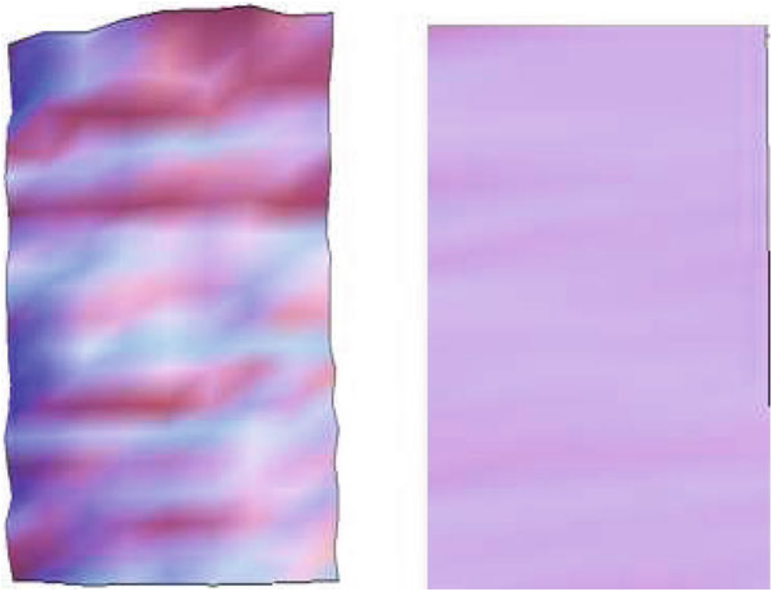

Fig. 14.10 A baseline experiment. At *left* is the probagram for dataset I and density $f_3(X := 1/v^2; a, b, c)$ under segmentation paradigm $\{12, 2, 16, \{2, 10\}\}$. At the *right* is the result of using the same number of points ($N = 1,119,299$) randomly placed within the full sample cuboid. This kind of experiment shows that the brain synapses are certainly *not* randomly distributed

In general, we display these "grams" looking down onto the surface or possible at a small tilt to be able to understand the surface visually.

What we shall call a segmentation paradigm is a set P of parameters that determine the precise manner in which we carve (a,b,c)-cuboids out of a full synapse dataset (Fig. 14.10). Symbolically,

$$P := \{M, G, H, \{b, e\}\},$$

where

- M is the "magnification" factor—the y-thickness of a cuboid divided into the full y-span of the dataset.
- G is the "grain"—which determines the oversampling; $1/G$ is the number of successively overlapping cuboids in one cuboid.
- H is the number of histogram bins in a 'gram plot, and we plot from bin b to bin e.

We generally use $g < 1$ to avoid possible alias effects at cuboid boundaries. The total number of cuboids analyzed in a 'gram thus turns out to be

$$S = 1 + G(M - 1).$$

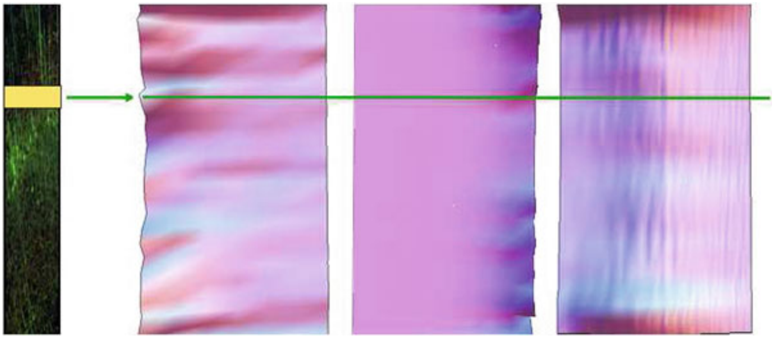

Fig. 14.11 Typical set of three "grams": At *far left* is a pictorialized a full-section sample, with a *small box* indicating a cuboid subsection. As said section is moved downward (increasing y), we obtain, *left-to-right* and for separation, $v := |\mathbf{r} - \mathbf{q}|$, the probagram for v^{-1}, then the paramegram for $\langle \exp(-kv^2) \rangle$, then the fractalgram. The phenomenon of neural layering is evident and qualitatively consistent (either correlated or anticorrelated) across all three "grams" for this sample (dataset I, detailed in Appendix: Synapse datasets)

For example, with grain $G = 3$ and $M = 10$, we calculate over a total of 28 cuboids. This is because there are generally $G = 3$ cuboids overlapping a given cuboid. In any case, one may take cuboid dimensions a, b, c as

$$a \ = \ x_{\max} - x_{\min}; \ b = \frac{y_{\max} - y_{\min}}{M} \ ; c \ = \ z_{\max} - z_{\min},$$

where min, max coordinates are deduced from the data (Fig. 14.11). (In our "grams", we continually recompute the min, max for every cuboid to guard against such as corner holes in the data.)

14.7 How Do We Explain the Observed Fractal Dimension?

Let us give an heuristic argument for the interaction of cuboid expectations and fractal-dimension estimates. Whereas the radial volume element in 3-space is $4\pi r^2 dr$, imagine a point-cloud having the property that the number of points a distance r from a given point scales as $r^{\delta-1}$ where $\delta < 3$, say. Then, if the characteristic size of a point sample is R (here we are being rough, avoiding discussion of the nature of the region boundaries), we might estimate an expectation for point-separation v to the sth power as

$$\langle v^s \rangle \ \sim \ \frac{\int_0^R u^s u^{\delta-1} \, \mathrm{d}u}{\int_0^R u^{\delta-1} \, \mathrm{d}u}.$$

Note that we can avoid calculation of a normalization constant by dividing this way, to enforce $\langle v^0 \rangle = 1$. This prescription gives the estimate

$$\langle v^s \rangle \sim \frac{\delta}{s+\delta} R^s,$$

showing a simple dependence on the fractal dimension δ. In fact, taking the left-hand plot of Fig. 14.6, we can right off estimate the fractal dimension of the whole dataset as

$$\delta \sim 2.6,$$

not too off the mark from our more precise fractal measurements that we report as 2.8 ± 0.05.

So one way to explain our discovered fractal dimension $\sim 2.8 < 3$ for both datasets is to surmise that the distance metric is weighted in some nonuniform fashion (Fig. 14.12).

14.7.1 Generalized Cantor Fractals

One aspect undertaken during the present research was to attempt to fit the observed fractal properties of the datasets to some form of Cantor fractal. There is a way to define a generalized Cantor fractal in n dimensions so that virtually any desired fractal dimension in the interval $[n\frac{\log 2}{\log 3}, n]$ (see [4]).[6] Such generalized Cantor fractals were used to fine-tune our fractal measurement machinery.

Interestingly, the cuboid expectations for dataset II seem qualitatively resonant with the corresponding expectations for a certain generalized Cantor set called $C_3(\overline{33111111})$ having dimension $\delta = 2.795\ldots$. However, dataset I does *not* have similar expectations on typical cuboids. For one thing, the highest-peak curve in Fig. 14.5—which is from a cuboid within dataset I—shows $\langle v \rangle$ for the laboratory data being less than the same expectation for random data; yet, a Cantor fractal tends to have such expectation *larger* than random data.

We shall soon turn to a different fractal model that appears to encompass the features of both datasets. But first, a word is appropriate here as to the meaning of "holes" in a dataset. Clearly, holes in the laboratory point-clouds will be caused

[6] Mathematically, the available fractal dimensions for the generalized Cantor fractals are dense in said interval.

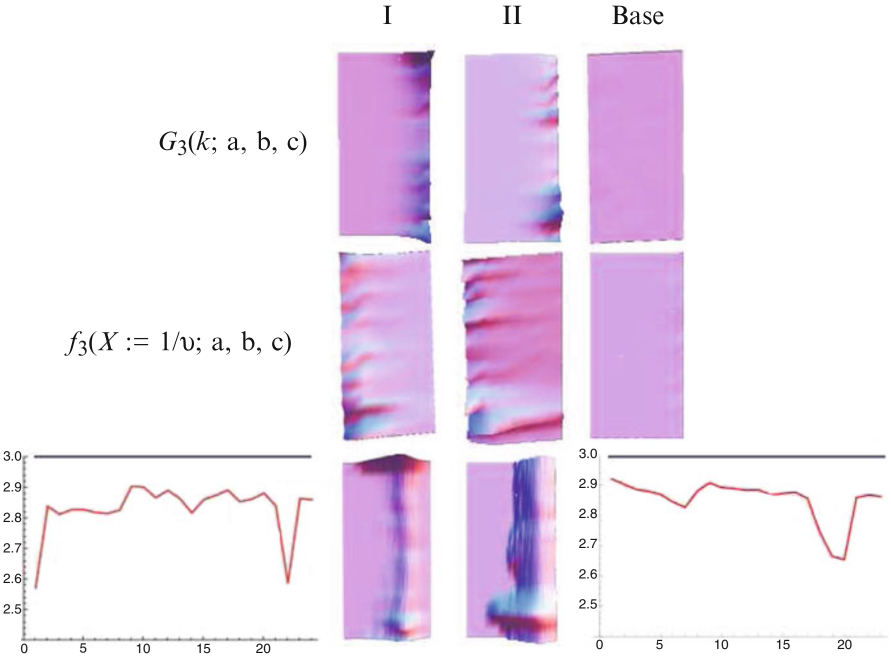

Fig. 14.12 The "grams" for the synapse-location datasets I, II. The *top row* shows G_3 paramegrams and baseline test for segmentation paradigm $\{12,2,32,\{1,10\}\}$. The *second row* shows probagrams for inverse separation $1/v$, in the same segmentation paradigm. The two 3D plots at *bottom* are the fractagrams. At *far-left* and *far-right bottom* are graphical displays of the per-cuboid fractal-dimension estimate. Note that the baseline test here is for a randomly filled cuboid; the *horizontal lines* at dimension 3.0 really are less noisy than one pixel width. Thus the datasets I, II can be said both to have overall fractal dimension 2.8 ± 0.5, although the dimension is evidently neural-layer dependent

by the simple fact of synapses not subsisting within large bodies.[7] So, too, Cantor fractals can be created by successive removal of holes that scale appropriately. But here is the rub: The existence of holes *does not in itself necessarily alter fractal dimension*.[8] For example, take a random cloud and remove large regions, to create essentially a swiss-cheese structure in between whose holes are equidistributed points. The key is, fractal-measurement machinery will still give a dimension very close to $\delta = 3$.

[7] Synapses live on dendrites, exterior to actual neurons.

[8] Of course, the situation is different if hole existence is connected with microscopic synapse distribution, e.g., if synapses were to concentrate near surfaces of large bodies.

14.7.2 *"Bouquet" Fractal as a Possible Synapse-Distribution Model*

We did find a kind of fractal that appears to lend itself well to comparison with synapse distributions.[9] We shall call such artificial constructs "bouquet" fractals. A generating algorithm to create a bouquet point-cloud having N points runs as follows:

1. In a unit 3-cube, generate N_0 random points (N_0 and other parameters can be used to "tune" the statistics of a bouquet fractal). Thus the point-cloud starts with population N_0.
2. Choose an initial radius $r = r_0$, a multiplicity number m, and a scale factor $c < 1$.
3. For each point in the point-cloud, generate m new points a mean distance r away (using, say, a normal distribution with deviation r away from a given point). At this juncture the point-cloud population will be $N_0 \cdot m^k$ for k being the number of times this step 3 has been executed. If this population is $\geq N$, go to step 5.
4. Reduce r by $r = c \cdot r$ and go to step 3.
5. Prune the point-cloud population so that the exact population is achieved.

The bouquet fractal will have fractal dimension on the order of

$$\delta \sim \frac{\log m}{-\log c},$$

but this is an asymptotic heuristic; in practice, one should simply tune all parameters to obtain experimental equivalencies.[10] For example, our dataset I corresponds interestingly to bouquet parameters

$$\{N_0, r_0, m, c\} = \{1000, N_0^{-1/3}, 23, 1/3\}.$$

The measured fractal dimension of the resulting bouquet for population $N = 1,119,299$ is $\delta \sim 2.85$ and statistical moments also show some similarity.

Once again, something like a bouquet fractal may not convey any neurophysiological understanding of synapses locations, but there could be a diagnostic parameter set, namely that set for which chosen statistical measures come out quantitatively similar.

[9] Again, we are not constructing here a neurophysiological model; rather, a phenomenological model whose statistical measures have qualitative commonality with the given synapse data.

[10] The heuristic form of dimension δ here may not be met if there are not enough total points. This is because the fractal-slope paradigm has low-resolution box counts that depend also on parameters N_0, r.

14.7.3 Nearest-Neighbor Calculus

Another idea that begs for further research is to perform nearest-neighbor calculus on synapse cuboids. This is yet a different way to detect departure from randomness.

In an n-dimensional unit volume, the asymptotic behavior of the nearest-pair distance for N uniformly randomly placed points, namely

$$\mu_1 \; := \; \langle \min |\mathbf{r} - \mathbf{q}| \rangle_{\mathbf{r},\mathbf{q} \in V},$$

is given—in its first asymptotic term—by

$$\mu_1 \sim \Gamma\left(1+\frac{1}{n}\right)\frac{2^{1/n}}{\sqrt{\pi}}\Gamma^{1/n}\left(1+\frac{n}{2}\right)\frac{1}{N^{2/d}} + \ldots .$$

In our $(n = 3)$-dimensional scenarios, we thus expect the nearest-pair separation to be

$$\mu_1 \sim \frac{\Gamma(4/3)}{\pi^{1/3}}\frac{1}{N^{2/3}} \approx \frac{0.6097}{N^{2/3}}.$$

It is interesting that this expression can be empirically verified with perhaps less inherent noise than one might expect.

Presumably a nearest-pair calculation on the synapse distributions will reveal once again significant departures from randomness. What we expect is a behavior like so

$$\mu_1 \sim \frac{\text{constant}}{N^{2/\delta}}$$

for fractal dimension δ. Probably the best research avenue, though, is to calculate the so-called k-nearest-pairs, meaning ordered k-tuples of successively more separate pairs, starting with the minimal pair, thus giving a list of expected ordered distances $\mu_1, \mu_2, \ldots, \mu_k$.

Acknowledgments The author is grateful to S. Arch of Reed College, as well as N. Weiler, S. Smith, and colleagues of Smithlab at Stanford Medical School, for their conceptual and algorithmic contributions to this project. T. Mehoke aided this research via statistical algorithms and preprocessing of synapse files. Mathematical colleague T. Wieting supported this research by being a selfless, invaluable resource for the more abstract fractal concepts. D. Bailey, J, Borwein, and M. Rose aided the author in regard to experimental mathematics on fractal sets. This author benefitted from productive discussions with the Advanced Computation Group at Apple, Inc.; in particular, D. Mitchell provided clutch statistical tools for various of the moment analyses herein.

Appendix 1: Synapse Datasets

Referring to Table 14.1: Both datasets I, II are from adult-mouse “barrel cortex” which is a region of the somatosensory neocortex involved in processing sensation from the facial whiskers (one of the mouse’s primary sensory modalities). The long y-axis of the volumes crosses all 6 layers of the neocortex (these are layers parallel to the cortical surface, and the long axis is perpendicular to the surface).

Neurophysiological considerations including array-tomography technology are discussed in [11–13] and web URL [15]; we give a brief synopsis:

Array tomography (AT) is a new high-throughput proteomic imaging method offering unprecedented capabilities for high-resolution imaging of tissue molecular architectures. AT is based on (1) automated physical, ultrathin sectioning of tissue specimens embedded in a hydrophilic resin, (2) construction of planar arrays of these serial sections on optical coverslips, (3) staining and imaging of these two-dimensional arrays, and (4) computational reconstruction into three dimensions, followed by (5) volumetric image analysis. The proteomic scope of AT is enhanced enormously by its unique amenability to high-dimensional immunofluorescence multiplexing via iterative cycles of antibody staining, imaging and antibody elution.

Appendix 2: Exact-Density Code

```
(* Evaluation of the exact Philip density F3[v,a,b,c]
 for an (a,b,c)-cuboid. *)

h11[u_, a_, b_, c_] := 1/(3 a^2 b^2 c^2) *
        If[u <= b^2, -3 Pi b c u + 4 b u^(3/2),
           If[u <= c^2,
                4 b^4 + 6 b^2 c Sqrt[u - b^2] -
                6 b c u ArcSin[b/Sqrt[u]],
                If[u <= b^2 + c^2, 4 b^4 + 6 b^2 c *
                      Sqrt[u - b^2] +
                      6 b c u (ArcCos[c/Sqrt[u]] -
                      ArcSin[b/Sqrt[u]]) -
```

Table 14.1 Synapse dataset characteristics

File, voxel nm×nm×nm	N	$(x_{\min}, x_{\max})$	$(y_{\min}, y_{\max})$	$(z_{\min}, z_{\max})$
I				
KDM-100824B 100×100×70	$1,119,299$	$(2800, 151300)$	$(2300, 1298000)$	$(105, 2835)$
II				
mMos3_Syn 100×100×200	$1,732,051$	$(100, 103400)$	$(100, 1252600)$	$(105, 4095)$

The point-cloud population N exceeds 10^6 for each dataset. The min, max parameters have been converted here to nm

```
                      2 b (2 u + c^2) Sqrt[u - c^2],
                      0
                ]
             ]
         ];

h12[u_, a_, b_, c_] := 1/(6 a^2 b^2 c^2) *
        If[u <= a^2,
           12 Pi a b c Sqrt[u] - 6 Pi a (b + c) u +
           8 (a + c) u^(3/2) - 3 u^2,
             If[u <= c^2,
                 5 a^4 - 6 Pi a^3 b +
                 12 Pi a b c Sqrt[u] +
                 8 c u^(3/2) - 12 Pi a b c *
                 Sqrt[u - a^2] - 8 c *
                 (u - a^2)^(3/2) -
                 12 a c u ArcSin[a/Sqrt[u]],
                 If[u <= a^2 + c^2,
                      5 a^4 - 6 Pi a^3 b +
                      6 Pi a b c^2 -
                      c^4 + 6 (Pi a b + c^2) u +
                      3 u^2 - 12 Pi a b c *
                      Sqrt[u - a^2] -
                      8 c (u - a^2)^(3/2) -
                      4 a (2 u + c^2)*
                      Sqrt[u - c^2] +
                      12 a c u *
              (ArcCos[c/Sqrt[u]]-ArcSin[a/Sqrt[u]]),
                 0
               ]
             ]
        ];

h22[u_, a_, b_, c_] := 1/(3 a^2 b^2 c^2) *
        If[u <= a^2, 0,
           If[u <= a^2 + b^2,
                3 Pi a^2 b (a + c) - 3 a^4 -
                6 Pi a b c Sqrt[u] +
                3 (a^2 + Pi b c) u +
                (6 Pi a b c - 2 (b + 3 c) a^2-4 b u)*
                Sqrt[u - a^2] -
                6 a b u ArcSin[a/Sqrt[u]],
                If[u <= a^2 + c^2,
                     3 a^2 b (Pi a - b) - 4 b^4-
                     12 a b c Sqrt[u]*
```

```
ArcSin[b Sqrt[u]/(Sqrt[a^2 + b^2] * Sqrt[u - a^2])]-
                     6 a c (a - Pi b) Sqrt[u - a^2]-
                     6 c (b^2 - a^2 +
                  2 a b ArcSin[a/Sqrt[a^2 + b^2]])*
                     Sqrt[u - a^2 - b^2] -
                     6 a b (a^2 + b^2)*
                     ArcSin[a/Sqrt[a^2 + b^2]] +
                     6 b c (a^2 + u) *
                     ArcSin[b/Sqrt[u - a^2]],
                     3 a^2 (a^2 - b^2 - c^2)-4 b^4-
                     3 a^2 u - 12 a b c  Sqrt[u]  *
(ArcSin[b Sqrt[u]/(Sqrt[a^2 + b^2] * Sqrt[u-a^2])]-
ArcCos[a c/(Sqrt[u - c^2] Sqrt[u - a^2])]) +
                     2 b (a^2 + c^2 + 2 u) *
                     Sqrt[u - a^2 - c^2] -
                     6 c *
(b^2 - a^2 + 2 a b ArcSin[a/Sqrt[a^2 + b^2]]) *
                     Sqrt[u - a^2 - b^2] -
                     6 a b (a^2 + b^2) *
                     ArcSin[a/Sqrt[a^2 + b^2]] +
                     6 b c (a^2 + u) *
(ArcSin[b/Sqrt[u - a^2]] - ArcCos[c/Sqrt[u - a^2]])+
                     6 a b (c^2 + u) *
                     ArcSin[a/Sqrt[u - c^2]]
               ]
             ]
        ];

h32[u_, a_, b_, c_] := h22[u, b, a, c];

h33[u_, a_, b_, c_] := 1/(6 a^2 b^2 c^2) *
        If[u <= b^2, 0,
           If[u <= a^2 + b^2,
                3 (2 Pi a b + b^2 + u) (u - b^2) -
                4 c (b^2 + 3 Pi a b + 2 u) *
                Sqrt[u - b^2],
                If[u <= b^2 + c^2, 3 (a^2 + b^2)^2 -
                   3 b^4 + 6 Pi a^3 b -
                   4 c (b^2 + 3 Pi a b + 2 u) *
                   Sqrt[u - b^2] +
                   4 c (a^2 + b^2 + 3 Pi a b + 2 u)*
                   Sqrt[u - a^2 - b^2],
                3 (a^2 + b^2)^2 + c^4 +
                   6 Pi a b (a^2 + b^2 - c^2) -
                   6 (Pi a b + c^2) u - 3 u^2 +
                   4 c (a^2 + b^2 + 3 Pi a b + 2 u)*
                   Sqrt[u - a^2 - b^2]
              ]
             ]
       ];

(* Next, the Philip density function for separation v.
    It must be arranged that a <= b <= c. *)

F3[v_, a_, b_, c_] :=
  2 v (h11[v^2, a, b, c] + h12[v^2, a, b, c] +
  h22[v^2, a, b, c] + h32[v^2, a, b, c] +
  h33[v^2, a, b, c]);
```

References

1. Adesnik, H., Scanziani, M.: Lateral competition for cortical space by layer-specific horizontal circuits. Nature **464**, 1155–1160 (2010)
2. Bailey, D., Borwein, J., Crandall, R.: Box integrals. J. Comput. Appl. Math. **206**, 196–208 (2007)
3. Bailey, D., Borwein, J., Crandall, R.: Advances in the theory of box integrals. Math. Comp. **79**, 1839–1866 (2010)
4. Bailey, D., Borwein, J., Crandall, R., Rose, M.: Expectations on fractal sets. Appl. Math. Comput. (2013)
5. Borwein, J., Chan, O-Y, Crandall, R.: Higher-dimensional box integrals. Exp. Math. **19**(3), 431–445 (2010)
6. Clarkson, K.J.: Nearest-neighbor searching and metric space dimensions. In: Shakhnarovich, G., et al. (eds.) Nearest-Neighbor Methods in Learning and Vision: Theory and Practice. MIT Press, Cambridge (2005)
7. Fischler, M.: Distribution of minimum distance among N random points in d dimensions. Technical Report FERMILAB-TM-2170 (2002)
8. Georgiev, S., et al.: EEG fractal dimension measurement before and after human auditory stimulation. Bioautomation **12**, 70–81 (2009)
9. Grski, A., Skrzat, J.: Error estimation of the fractal dimension measurements of cranial sutures. J. Anat. **208**(3), 353–359 (2006)
10. Lynnerup, N., Jacobsen, J.C.: Brief communication: age and fractal dimensions of human sagittal and coronal sutures. Am. J. Phys. Anthropol. **121**(4), 332–6 (2003)
11. Micheva, K.D., Smith, S.J.: Array tomography: A new tool for imaging the molecular architecture and ultrastructure of neural circuits. Neuron **55**, 25–36 (2007)
12. Micheva, K.D., Busse, B., Weiler, N.C., O'Rourke, N., Smith, S.J.: Single-synapse analysis of a diverse synapse population: proteomic imaging methods and markers. Neuron **68**(4), 639–53 (2010)
13. Micheva, K.D., O'Rourke, N., Busse, B., Smith, S.J.: Array tomography: high-resolution three-dimensional immunofluorescence. In: Imaging: A Laboratory Manual, Ch. 45, pp. 697–719, 3rd edn. Cold Spring Harbor Press, New York (2010)
14. Philip, J.: The probability distribution of the distance between two random points in a box. TRITA MAT 07 MA 10 (2007)
15. Smithlab: Array tomography. http://smithlab.stanford.edu/Smithlab/Array_Tomography.html

16. Spruston, N.: Pyramidal neurons: dendritic structure and synaptic integration. Nat. Rev. Neurosci. **9**, 206–221 (2008)
17. Teich, M., et al.: Fractal character of the neural spike train in the visual system of the cat. J. Opt. Soc. Am. A **14**(3), 529–546 (1997)