Validating the Correctness of Hardware Implementations of the NBS Data Encryption Standard, 1980, NBS Special Publication 500–20 (Washington: U.S. Department of Commerce, National Bureau of Standards). [3]

Meyer, C.H. and Matyas, S.M. 1982, *Cryptography: A New Dimension in Computer Data Security* (New York: Wiley). [4]

Knuth, D.E. 1973, *Sorting and Searching*, vol. 3 of *The Art of Computer Programming* (Reading, MA: Addison-Wesley), Chapter 6. [5]

Vitter, J.S., and Chen, W-C. 1987, *Design and Analysis of Coalesced Hashing* (New York: Oxford University Press). [6]

7.6 Simple Monte Carlo Integration

Inspirations for numerical methods can spring from unlikely sources. “Splines” first were flexible strips of wood used by draftsmen. “Simulated annealing” (we shall see in §10.9) is rooted in a thermodynamic analogy. And who does not feel at least a faint echo of glamor in the name “Monte Carlo method”?

Suppose that we pick N random points, uniformly distributed in a multidimensional volume V. Call them $x_1, \ldots, x_N$. Then the basic theorem of Monte Carlo integration estimates the integral of a function f over the multidimensional volume,

$$\int f\,dV \approx V\,\langle f\rangle \;\pm V\sqrt{\frac{\langle f^2\rangle - \langle f\rangle^2}{N}} \tag{7.6.1}$$

Here the angle brackets denote taking the arithmetic mean over the N sample points,

$$\langle f\rangle \equiv \frac{1}{N}\sum_{i=1}^{N} f(x_i) \qquad \left\langle f^2\right\rangle \equiv \frac{1}{N}\sum_{i=1}^{N} f^2(x_i) \tag{7.6.2}$$

The “plus-or-minus” term in (7.6.1) is a one standard deviation error estimate for the integral, not a rigorous bound; further, there is no guarantee that the error is distributed as a Gaussian, so the error term should be taken only as a rough indication of probable error.

Suppose that you want to integrate a function g over a region W that is not easy to sample randomly. For example, W might have a very complicated shape. No problem. Just find a region V that *includes* W and that *can* easily be sampled (Figure 7.6.1), and then define f to be equal to g for points in W and equal to zero for points outside of W (but still inside the sampled V). You want to try to make V enclose W as closely as possible, because the zero values of f will increase the error estimate term of (7.6.1). And well they should: points chosen outside of W have no information content, so the effective value of N, the number of points, is reduced. The error estimate in (7.6.1) takes this into account.

General purpose routines for Monte Carlo integration are quite complicated (see §7.8), but a worked example will show the underlying simplicity of the method. Suppose that we want to find the weight and the position of the center of mass of an

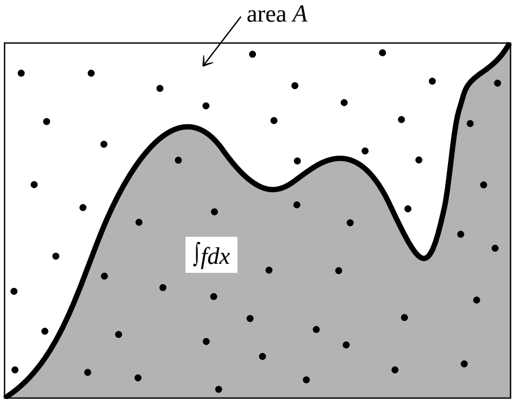


Figure 7.6.1. Monte Carlo integration. Random points are chosen within the area A. The integral of the function f is estimated as the area of A multiplied by the fraction of random points that fall below the curve f. Refinements on this procedure can improve the accuracy of the method; see text.

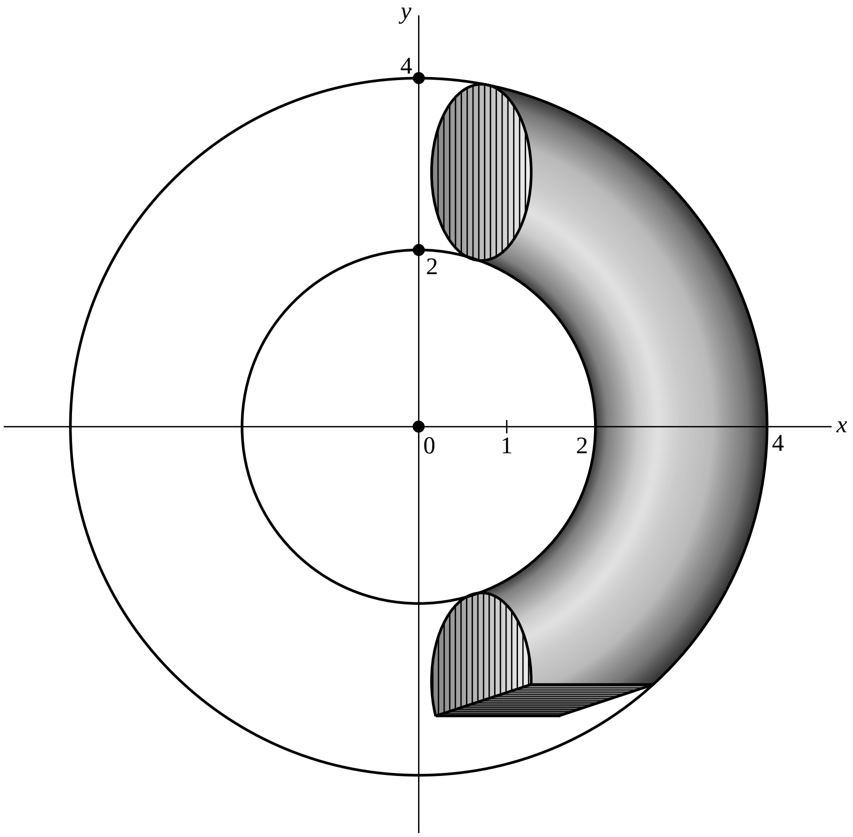


Figure 7.6.2. Example of Monte Carlo integration (see text). The region of interest is a piece of a torus, bounded by the intersection of two planes. The limits of integration of the region cannot easily be written in analytically closed form, so Monte Carlo is a useful technique.

object of complicated shape, namely the intersection of a torus with the edge of a large box. In particular let the object be defined by the three simultaneous conditions

$$z^2 + \left(\sqrt{x^2+y^2} - 3\right)^2 \le 1 \tag{7.6.3}$$

(torus centered on the origin with major radius $= 4$, minor radius $= 2$)

$$x \ge 1 \qquad y \ge -3 \tag{7.6.4}$$

(two faces of the box, see Figure 7.6.2). Suppose for the moment that the object has a constant density ρ.

We want to estimate the following integrals over the interior of the complicated object:

$$\int \rho\, dx\, dy\, dz \qquad \int x\rho\, dx\, dy\, dz \qquad \int y\rho\, dx\, dy\, dz \qquad \int z\rho\, dx\, dy\, dz \tag{7.6.5}$$

The coordinates of the center of mass will be the ratio of the latter three integrals (linear moments) to the first one (the weight).

In the following fragment, the region V, enclosing the piece-of-torus W, is the rectangular box extending from 1 to 4 in x, -3 to 4 in y, and -1 to 1 in z.

```
n=                                      Set to the number of sample points desired.
den=                                    Set to the constant value of the density.
sw=0.                                   Zero the various sums to be accumulated.
swx=0.
swy=0.
swz=0.
varw=0.
varx=0.
vary=0.
varz=0.
vol=3.*7.*2.                            Volume of the sampled region.
do 11 j=1,n
    x=1.+3.*ran2(idum)                  Pick a point randomly in the sampled region.
    y=-3.+7.*ran2(idum)
    z=-1.+2.*ran2(idum)
    if (z**2+(sqrt(x**2+y**2)-3.)**2.le.1.)then       Is it in the torus?
        sw=sw+den                       If so, add to the various cumulants.
        swx=swx+x*den
        swy=swy+y*den
        swz=swz+z*den
        varw=varw+den**2
        varx=varx+(x*den)**2
        vary=vary+(y*den)**2
        varz=varz+(z*den)**2
    endif
enddo 11
w=vol*sw/n                              The values of the integrals (7.6.5),
x=vol*swx/n
y=vol*swy/n
z=vol*swz/n
dw=vol*sqrt((varw/n-(sw/n)**2)/n)       and their corresponding error estimates.
dx=vol*sqrt((varx/n-(swx/n)**2)/n)
dy=vol*sqrt((vary/n-(swy/n)**2)/n)
dz=vol*sqrt((varz/n-(swz/n)**2)/n)
```

A change of variable can often be extremely worthwhile in Monte Carlo integration. Suppose, for example, that we want to evaluate the same integrals, but for a piece-of-torus whose density is a strong function of z, in fact varying according to

$$\rho(x, y, z) = e^{5z} \tag{7.6.6}$$

One way to do this is to put the statement

```
        den=exp(5.*z)
```

inside the `if...then` block, just before `den` is first used. This will work, but it is a poor way to proceed. Since (7.6.6) falls so rapidly to zero as z decreases (down to its lower limit -1), most sampled points contribute almost nothing to the sum of the weight or moments. These points are effectively wasted, almost as badly as those that fall outside of the region W. A change of variable, exactly as in the transformation methods of §7.2, solves this problem. Let

$$ds = e^{5z}dz \qquad \text{so that} \qquad s = \frac{1}{5}e^{5z}, \quad z = \frac{1}{5}\ln(5s) \tag{7.6.7}$$

Then $\rho dz = ds$, and the limits $-1 < z < 1$ become $.00135 < s < 29.682$. The program fragment now looks like this

```
n=                                      Set to the number of sample points desired.
sw=0.
swx=0.
swy=0.
swz=0.
varw=0.
varx=0.
vary=0.
varz=0.
ss=(0.2*(exp(5.)-exp(-5.)))             Interval of s to be random sampled.
vol=3.*7.*ss                            Volume in x,y,s-space.
do 11 j=1,n
    x=1.+3.*ran2(idum)
    y=-3.+7.*ran2(idum)
    s=.00135+ss*ran2(idum)              Pick a point in s.
    z=0.2*log(5.*s)                     Equation (7.6.7).
    if (z**2+(sqrt(x**2+y**2)-3.)**2.lt.1.)then
        sw=sw+1.                        Density is 1, since absorbed into definition of s.
        swx=swx+x
        swy=swy+y
        swz=swz+z
        varw=varw+1.
        varx=varx+x**2
        vary=vary+y**2
        varz=varz+z**2
    endif
enddo 11
w=vol*sw/n                              The values of the integrals (7.6.5),
x=vol*swx/n
y=vol*swy/n
z=vol*swz/n
dw=vol*sqrt((varw/n-(sw/n)**2)/n)       and their corresponding error estimates.
dx=vol*sqrt((varx/n-(swx/n)**2)/n)
dy=vol*sqrt((vary/n-(swy/n)**2)/n)
dz=vol*sqrt((varz/n-(swz/n)**2)/n)
```

If you think for a minute, you will realize that equation (7.6.7) was useful only because the part of the integrand that we wanted to eliminate (e^{5z}) was both integrable analytically, and had an integral that could be analytically inverted. (Compare §7.2.) In general these properties will not hold. Question: What then? Answer: Pull out of the integrand the "best" factor that *can* be integrated and inverted. The criterion for "best" is to try to reduce the remaining integrand to a function that is as close as possible to constant.

The limiting case is instructive: If you manage to make the integrand f *exactly* constant, and if the region V, of known volume, *exactly* encloses the desired region W, then the average of f that you compute will be exactly its constant value, and the error estimate in equation (7.6.1) will exactly vanish. You will, in fact, have done the integral exactly, and the Monte Carlo numerical evaluations are superfluous. So, backing off from the extreme limiting case, *to the extent* that you are able to make f approximately constant by change of variable, and *to the extent* that you can sample a region only slightly larger than W, you will increase the accuracy of the Monte Carlo integral. This technique is generically called *reduction of variance* in the literature.

The fundamental disadvantage of simple Monte Carlo integration is that its accuracy increases only as the square root of N, the number of sampled points. If your accuracy requirements are modest, or if your computer budget is large, then the technique is highly recommended as one of great generality. In the next two sections we will see that there are techniques available for "breaking the square root of N barrier" and achieving, at least in some cases, higher accuracy with fewer function evaluations.

CITED REFERENCES AND FURTHER READING:

Hammersley, J.M., and Handscomb, D.C. 1964, *Monte Carlo Methods* (London: Methuen).

Shreider, Yu. A. (ed.) 1966, *The Monte Carlo Method* (Oxford: Pergamon).

Sobol', I.M. 1974, *The Monte Carlo Method* (Chicago: University of Chicago Press).

Kalos, M.H., and Whitlock, P.A. 1986, *Monte Carlo Methods* (New York: Wiley).

7.7 Quasi- (that is, Sub-) Random Sequences

We have just seen that choosing N points uniformly randomly in an n-dimensional space leads to an error term in Monte Carlo integration that decreases as $1/\sqrt{N}$. In essence, each new point sampled adds linearly to an accumulated sum that will become the function average, and also linearly to an accumulated sum of squares that will become the variance (equation 7.6.2). The estimated error comes from the square root of this variance, hence the power $N^{-1/2}$.

Just because this square root convergence is familiar does not, however, mean that it is inevitable. A simple counterexample is to choose sample points that lie on a Cartesian grid, and to sample each grid point exactly once (in whatever order). The Monte Carlo method thus becomes a deterministic quadrature scheme — albeit a simple one — whose fractional error decreases at least as fast as N^{-1} (even faster if the function goes to zero smoothly at the boundaries of the sampled region, or is periodic in the region).

15.6 Confidence Limits on Estimated Model Parameters

Several times already in this chapter we have made statements about the standard errors, or uncertainties, in a set of M estimated parameters **a**. We have given some formulas for computing standard deviations or variances of individual parameters (equations 15.2.9, 15.4.15, 15.4.19), as well as some formulas for covariances between pairs of parameters (equation 15.2.10; remark following equation 15.4.15; equation 15.4.20; equation 15.5.15).

In this section, we want to be more explicit regarding the precise meaning of these quantitative uncertainties, and to give further information about how quantitative confidence limits on fitted parameters can be estimated. The subject can get somewhat technical, and even somewhat confusing, so we will try to make precise statements, even when they must be offered without proof.

Figure 15.6.1 shows the conceptual scheme of an experiment that "measures" a set of parameters. There is some underlying true set of parameters $\mathbf{a}_{\rm true}$ that are known to Mother Nature but hidden from the experimenter. These true parameters are statistically realized, along with random measurement errors, as a measured data set, which we will symbolize as $\mathcal{D}_{(0)}$. The data set $\mathcal{D}_{(0)}$ *is* known to the experimenter. He or she fits the data to a model by χ^2 minimization or some other technique, and obtains measured, i.e., fitted, values for the parameters, which we here denote $\mathbf{a}_{(0)}$.

Because measurement errors have a random component, $\mathcal{D}_{(0)}$ is not a unique realization of the true parameters $\mathbf{a}_{\rm true}$. Rather, there are infinitely many other realizations of the true parameters as "hypothetical data sets" each of which *could* have been the one measured, but happened not to be. Let us symbolize these by $\mathcal{D}_{(1)}, \mathcal{D}_{(2)}, \ldots$. Each one, had it been realized, would have given a slightly different set of fitted parameters, $\mathbf{a}_{(1)}, \mathbf{a}_{(2)}, \ldots$, respectively. These parameter sets $\mathbf{a}_{(i)}$ therefore occur with some probability distribution in the M-dimensional space of all possible parameter sets **a**. The actual measured set $\mathbf{a}_{(0)}$ is one member drawn from this distribution.

Even more interesting than the probability distribution of $\mathbf{a}_{(i)}$ would be the distribution of the difference $\mathbf{a}_{(i)} - \mathbf{a}_{\rm true}$. This distribution differs from the former one by a translation that puts Mother Nature's true value at the origin. If we knew *this* distribution, we would know everything that there is to know about the quantitative uncertainties in our experimental measurement $\mathbf{a}_{(0)}$.

So the name of the game is to find some way of estimating or approximating the probability distribution of $\mathbf{a}_{(i)} - \mathbf{a}_{\rm true}$ without knowing $\mathbf{a}_{\rm true}$ and without having available to us an infinite universe of hypothetical data sets.

Monte Carlo Simulation of Synthetic Data Sets

Although the measured parameter set $\mathbf{a}_{(0)}$ is not the true one, let us consider a fictitious world in which it *was* the true one. Since we hope that our measured parameters are not *too* wrong, we hope that that fictitious world is not too different from the actual world with parameters $\mathbf{a}_{\rm true}$. In particular, let us hope — no, let us *assume* — that the shape of the probability distribution $\mathbf{a}_{(i)} - \mathbf{a}_{(0)}$ in the fictitious world is the same, or very nearly the same, as the shape of the probability distribution

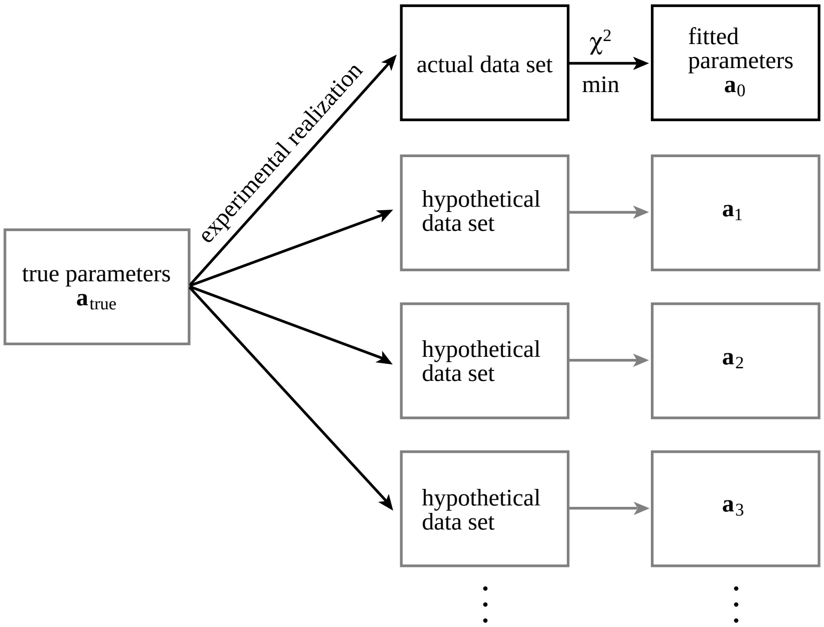


Figure 15.6.1. A statistical universe of data sets from an underlying model. True parameters $\mathbf{a}_{\rm true}$ are realized in a data set, from which fitted (observed) parameters $\mathbf{a}_0$ are obtained. If the experiment were repeated many times, new data sets and new values of the fitted parameters would be obtained.

$\mathbf{a}_{(i)} - \mathbf{a}_{\rm true}$ in the real world. Notice that we are not assuming that $\mathbf{a}_{(0)}$ and $\mathbf{a}_{\rm true}$ are equal; they are certainly not. We are only assuming that the way in which random errors enter the experiment and data analysis does not vary rapidly as a function of $\mathbf{a}_{\rm true}$, so that $\mathbf{a}_{(0)}$ can serve as a reasonable surrogate.

Now, often, the distribution of $\mathbf{a}_{(i)} - \mathbf{a}_{(0)}$ in the fictitious world *is* within our power to calculate (see Figure 15.6.2). If we know something about the process that generated our data, given an assumed set of parameters $\mathbf{a}_{(0)}$, then we can usually figure out how to *simulate* our own sets of "synthetic" realizations of these parameters as "synthetic data sets." The procedure is to draw random numbers from appropriate distributions (cf. §7.2–§7.3) so as to mimic our best understanding of the underlying process and measurement errors in our apparatus. With such random draws, we construct data sets with exactly the same numbers of measured points, and precisely the same values of all control (independent) variables, as our actual data set $\mathcal{D}_{(0)}$. Let us call these simulated data sets $\mathcal{D}^S_{(1)}, \mathcal{D}^S_{(2)}, \ldots$. By construction these are supposed to have exactly the same statistical relationship to $\mathbf{a}_{(0)}$ as the $\mathcal{D}_{(i)}$'s have to $\mathbf{a}_{\rm true}$. (For the case where you don't know enough about what you are measuring to do a credible job of simulating it, see below.)

Next, for each $\mathcal{D}^S_{(j)}$, perform exactly the same procedure for estimation of parameters, e.g., χ^2 minimization, as was performed on the actual data to get the parameters $\mathbf{a}_{(0)}$, giving simulated measured parameters $\mathbf{a}^S_{(1)}, \mathbf{a}^S_{(2)}, \ldots$. Each simulated measured parameter set yields a point $\mathbf{a}^S_{(i)} - \mathbf{a}_{(0)}$. Simulate enough data sets and enough derived simulated measured parameters, and you map out the desired probability distribution in M dimensions.

In fact, the ability to do *Monte Carlo simulations* in this fashion has revo-

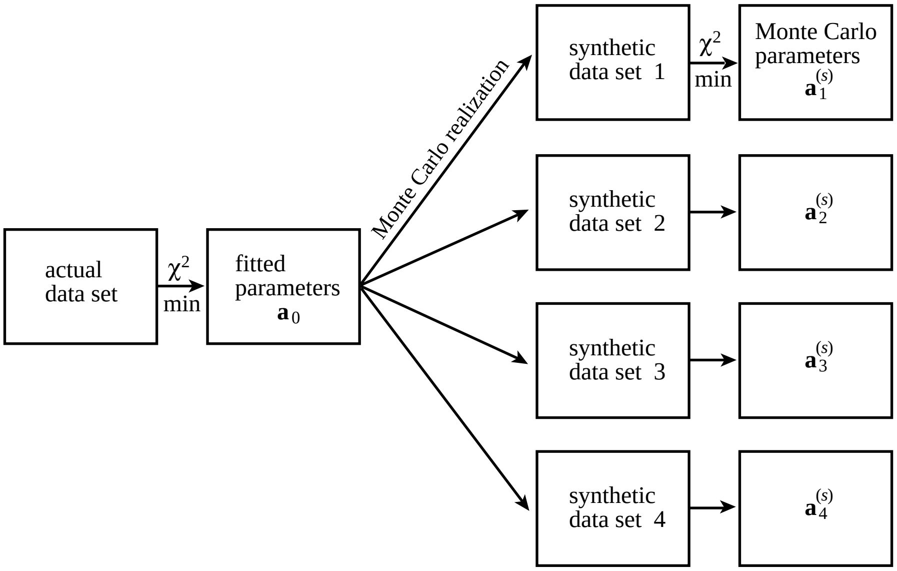


Figure 15.6.2. Monte Carlo simulation of an experiment. The fitted parameters from an actual experiment are used as surrogates for the true parameters. Computer-generated random numbers are used to simulate many synthetic data sets. Each of these is analyzed to obtain its fitted parameters. The distribution of these fitted parameters around the (known) surrogate true parameters is thus studied.

lutionized many fields of modern experimental science. Not only is one able to characterize the errors of parameter estimation in a very precise way; one can also try out on the computer different methods of parameter estimation, or different data reduction techniques, and seek to minimize the uncertainty of the result according to any desired criteria. Offered the choice between mastery of a five-foot shelf of analytical statistics books and middling ability at performing statistical Monte Carlo simulations, we would surely choose to have the latter skill.

Quick-and-Dirty Monte Carlo: The Bootstrap Method

Here is a powerful technique that can often be used when you don't know enough about the underlying process, or the nature of your measurement errors, to do a credible Monte Carlo simulation. Suppose that your data set consists of N *independent and identically distributed* (or *iid*) "data points." Each data point probably consists of several numbers, e.g., one or more control variables (uniformly distributed, say, in the range that you have decided to measure) and one or more associated measured values (each distributed however Mother Nature chooses). "Iid" means that the sequential order of the data points is not of consequence to the process that you are using to get the fitted parameters **a**. For example, a χ^2 sum like (15.5.5) does not care in what order the points are added. Even simpler examples are the mean value of a measured quantity, or the mean of some function of the measured quantities.

The *bootstrap method* [1] uses the actual data set $\mathcal{D}_{(0)}^S$, with its N data points, to generate any number of synthetic data sets $\mathcal{D}_{(1)}^S, \mathcal{D}_{(2)}^S, \ldots$, also with N data points. The procedure is simply to draw N data points at a time *with replacement* from the set $\mathcal{D}^S_{(0)}$. Because of the replacement, you do not simply get back your original data set each time. You get sets in which a random fraction of the original points, typically $\sim 1/e \approx 37\%$, are replaced by *duplicated* original points. Now, exactly as in the previous discussion, you subject these data sets to the same estimation procedure as was performed on the actual data, giving a set of simulated measured parameters $\mathbf{a}^S_{(1)}, \mathbf{a}^S_{(2)}, \ldots$. These will be distributed around $\mathbf{a}_{(0)}$ in close to the same way that $\mathbf{a}_{(0)}$ is distributed around $\mathbf{a}_{\rm true}$.

Sounds like getting something for nothing, doesn't it? In fact, it has taken more than a decade for the bootstrap method to become accepted by statisticians. By now, however, enough theorems have been proved to render the bootstrap reputable (see [2] for references). The basic idea behind the bootstrap is that the actual data set, viewed as a probability distribution consisting of delta functions at the measured values, is in most cases the best — or only — available estimator of the underlying probability distribution. It takes courage, but one can often simply use *that* distribution as the basis for Monte Carlo simulations.

Watch out for cases where the bootstrap's "iid" assumption is violated. For example, if you have made measurements at evenly spaced intervals of some control variable, then you can *usually* get away with pretending that these are "iid," uniformly distributed over the measured range. However, some estimators of **a** (e.g., ones involving Fourier methods) might be particularly sensitive to all the points on a grid being present. In that case, the bootstrap is going to give a wrong distribution. Also watch out for estimators that look at anything like small-scale clumpiness within the N data points, or estimators that sort the data and look at sequential differences. Obviously the bootstrap will fail on these, too. (The theorems justifying the method are still true, but some of their technical assumptions are violated by these examples.)

For a large class of problems, however, the bootstrap does yield easy, *very quick*, Monte Carlo estimates of the errors in an estimated parameter set.

Confidence Limits

Rather than present all details of the probability distribution of errors in parameter estimation, it is common practice to summarize the distribution in the form of *confidence limits*. The full probability distribution is a function defined on the M-dimensional space of parameters **a**. A *confidence region* (or *confidence interval*) is just a region of that M-dimensional space (hopefully a small region) that contains a certain (hopefully large) percentage of the total probability distribution. You point to a confidence region and say, e.g., "there is a 99 percent chance that the true parameter values fall within this region around the measured value."

It is worth emphasizing that you, the experimenter, get to pick both the *confidence level* (99 percent in the above example), and the shape of the confidence region. The only requirement is that your region does include the stated percentage of probability. Certain percentages are, however, customary in scientific usage: 68.3 percent (the lowest confidence worthy of quoting), 90 percent, 95.4 percent, 99 percent, and 99.73 percent. Higher confidence levels are conventionally "ninety-nine point nine ... nine." As for shape, obviously you want a region that is compact and reasonably centered on your measurement $\mathbf{a}_{(0)}$, since the whole purpose of a confidence limit is to inspire confidence in that measured value. In one dimension,

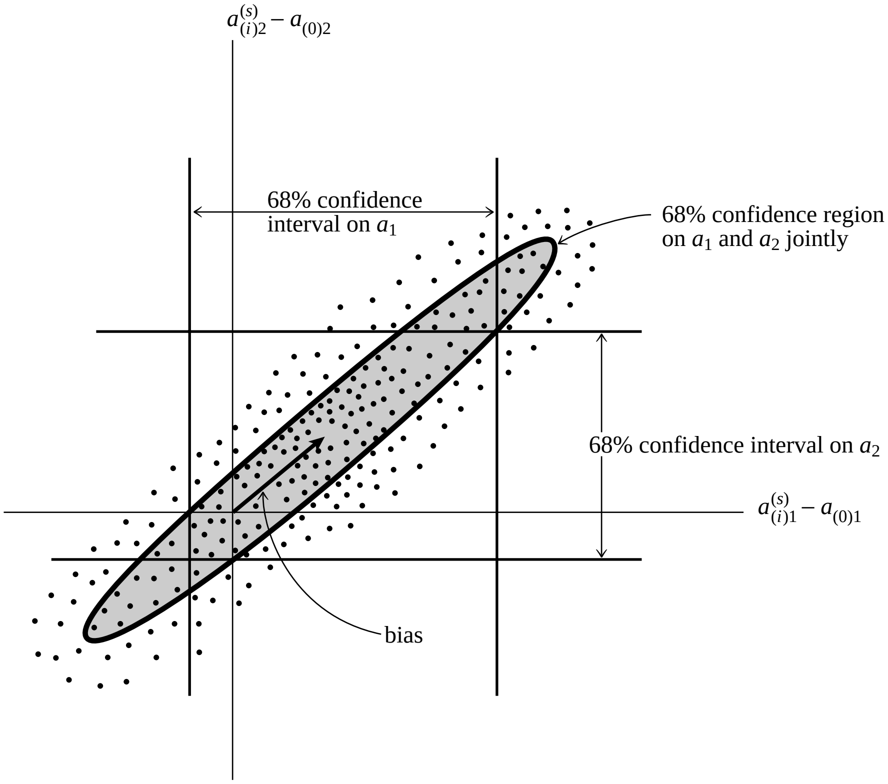

$a^{(s)}_{(i)2} - a_{(0)2}$
68% confidence interval on a_1
68% confidence region on a_1 and a_2 jointly
68% confidence interval on a_2
$a^{(s)}_{(i)1} - a_{(0)1}$
bias

Figure 15.6.3. Confidence intervals in 1 and 2 dimensions. The same fraction of measured points (here 68%) lies (i) between the two vertical lines, (ii) between the two horizontal lines, (iii) within the ellipse.

the convention is to use a line segment centered on the measured value; in higher dimensions, ellipses or ellipsoids are most frequently used.

You might suspect, correctly, that the numbers 68.3 percent, 95.4 percent, and 99.73 percent, and the use of ellipsoids, have some connection with a normal distribution. That is true historically, but not always relevant nowadays. In general, the probability distribution of the parameters will not be normal, and the above numbers, used as levels of confidence, are purely matters of convention.

Figure 15.6.3 sketches a possible probability distribution for the case $M = 2$. Shown are three different confidence regions which might usefully be given, all at the same confidence level. The two vertical lines enclose a band (horizontal interval) which represents the 68 percent confidence interval for the variable a_1 without regard to the value of a_2. Similarly the horizontal lines enclose a 68 percent confidence interval for a_2. The ellipse shows a 68 percent confidence interval for a_1 and a_2 jointly. Notice that to enclose the same probability as the two bands, the ellipse must necessarily extend outside of both of them (a point we will return to below).

Constant Chi-Square Boundaries as Confidence Limits

When the method used to estimate the parameters $\mathbf{a}_{(0)}$ is chi-square minimization, as in the previous sections of this chapter, then there is a natural choice for the

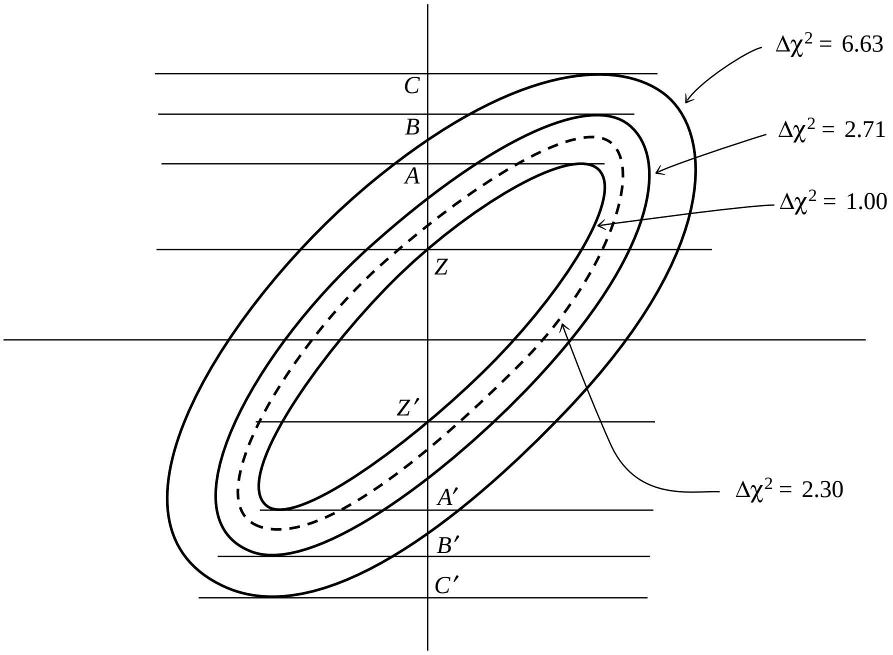


Figure 15.6.4. Confidence region ellipses corresponding to values of chi-square larger than the fitted minimum. The solid curves, with $\Delta\chi^2 = 1.00, 2.71, 6.63$ project onto one-dimensional intervals AA', BB', CC'. These intervals — not the ellipses themselves — contain 68.3%, 90%, and 99% of normally distributed data. The ellipse that contains 68.3% of normally distributed data is shown dashed, and has $\Delta\chi^2 = 2.30$. For additional numerical values, see accompanying table.

shape of confidence intervals, whose use is almost universal. For the observed data set $\mathcal{D}_{(0)}$, the value of χ^2 is a minimum at $\mathbf{a}_{(0)}$. Call this minimum value $\chi^2_{\rm min}$. If the vector **a** of parameter values is perturbed away from $\mathbf{a}_{(0)}$, then χ^2 increases. The region within which χ^2 increases by no more than a set amount $\Delta\chi^2$ defines some M-dimensional confidence region around $\mathbf{a}_{(0)}$. If $\Delta\chi^2$ is set to be a large number, this will be a big region; if it is small, it will be small. Somewhere in between there will be choices of $\Delta\chi^2$ that cause the region to contain, variously, 68 percent, 90 percent, etc. of probability distribution for **a**'s, as defined above. These regions are taken as the confidence regions for the parameters $\mathbf{a}_{(0)}$.

Very frequently one is interested not in the full M-dimensional confidence region, but in individual confidence regions for some smaller number ν of parameters. For example, one might be interested in the confidence interval of each parameter taken separately (the bands in Figure 15.6.3), in which case $\nu = 1$. In that case, the natural confidence regions in the ν-dimensional subspace of the M-dimensional parameter space are the *projections* of the M-dimensional regions defined by fixed $\Delta\chi^2$ into the ν-dimensional spaces of interest. In Figure 15.6.4, for the case $M = 2$, we show regions corresponding to several values of $\Delta\chi^2$. The one-dimensional confidence interval in a_2 corresponding to the region bounded by $\Delta\chi^2 = 1$ lies between the lines A and A'.

Notice that the projection of the higher-dimensional region on the lower-dimension space is used, not the intersection. The intersection would be the band between Z and Z'. It is *never* used. It is shown in the figure only for the purpose of making this cautionary point, that it should not be confused with the projection.

Probability Distribution of Parameters in the Normal Case

You may be wondering why we have, in this section up to now, made no connection at all with the error estimates that come out of the χ^2 fitting procedure, most notably the covariance matrix C_{ij}. The reason is this: χ^2 minimization is a useful means for estimating parameters even if the measurement errors are not normally distributed. While normally distributed errors are required if the χ^2 parameter estimate is to be a maximum likelihood estimator (§15.1), one is often willing to give up that property in return for the relative convenience of the χ^2 procedure. Only in extreme cases, measurement error distributions with very large "tails," is χ^2 minimization abandoned in favor of more robust techniques, as will be discussed in §15.7.

However, the formal covariance matrix that comes out of a χ^2 minimization has a clear quantitative interpretation only if (or to the extent that) the measurement errors actually are normally distributed. In the case of *non*normal errors, you are "allowed"

- to fit for parameters by minimizing χ^2
- to use a contour of constant $\Delta\chi^2$ as the boundary of your confidence region
- to use Monte Carlo simulation or detailed analytic calculation in determining *which* contour $\Delta\chi^2$ is the correct one for your desired confidence level
- to give the covariance matrix C_{ij} as the "formal covariance matrix of the fit."

You are *not* allowed

- to use formulas that we now give for the case of normal errors, which establish quantitative relationships among $\Delta\chi^2$, C_{ij}, and the confidence level.

Here are the key theorems that hold when (i) the measurement errors are normally distributed, and either (ii) the model is linear in its parameters or (iii) the sample size is large enough that the uncertainties in the fitted parameters **a** do not extend outside a region in which the model could be replaced by a suitable linearized model. [Note that condition (iii) does not preclude your use of a nonlinear routine like `mqrfit` to *find* the fitted parameters.]

Theorem A. $\chi^2_{\min}$ is distributed as a chi-square distribution with $N - M$ degrees of freedom, where N is the number of data points and M is the number of fitted parameters. This is the basic theorem that lets you evaluate the goodness-of-fit of the model, as discussed above in §15.1. We list it first to remind you that unless the goodness-of-fit is credible, the whole estimation of parameters is suspect.

Theorem B. If $\mathbf{a}^S_{(j)}$ is drawn from the universe of simulated data sets with actual parameters $\mathbf{a}_{(0)}$, then the probability distribution of $\delta\mathbf{a} \equiv \mathbf{a}^S_{(j)} - \mathbf{a}_{(0)}$ is the multivariate normal distribution

$$P(\delta\mathbf{a})\ da_1 \dots da_M = \text{const.} \times \exp\left(-\frac{1}{2}\delta\mathbf{a}\cdot[\alpha]\cdot\delta\mathbf{a}\right)\ da_1 \dots da_M$$

where $[\alpha]$ is the curvature matrix defined in equation (15.5.8).

Theorem C. If $\mathbf{a}^S_{(j)}$ is drawn from the universe of simulated data sets with actual parameters $\mathbf{a}_{(0)}$, then the quantity $\Delta\chi^2 \equiv \chi^2(\mathbf{a}_{(j)}) - \chi^2(\mathbf{a}_{(0)})$ is distributed as a chi-square distribution with M degrees of freedom. Here the χ^2's are all evaluated using the fixed (actual) data set $\mathcal{D}_{(0)}$. This theorem makes the connection between particular values of $\Delta\chi^2$ and the fraction of the probability distribution that they enclose as an M-dimensional region, i.e., the confidence level of the M-dimensional confidence region.

Theorem D. Suppose that $\mathbf{a}^S_{(j)}$ is drawn from the universe of simulated data sets (as above), that its first ν components $a_1, \ldots, a_\nu$ are held fixed, and that its remaining $M - \nu$ components are varied so as to minimize χ^2. Call this minimum value χ^2_ν. Then $\Delta\chi^2_\nu \equiv \chi^2_\nu - \chi^2_{\min}$ is distributed as a chi-square distribution with ν degrees of freedom. If you consult Figure 15.6.4, you will see that this theorem connects the *projected* $\Delta\chi^2$ region with a confidence level. In the figure, a point that is held fixed in a_2 and allowed to vary in a_1 minimizing χ^2 will seek out the ellipse whose top or bottom edge is tangent to the line of constant a_2, and is therefore the line that projects it onto the smaller-dimensional space.

As a first example, let us consider the case $\nu = 1$, where we want to find the confidence interval of a single parameter, say a_1. Notice that the chi-square distribution with $\nu = 1$ degree of freedom is the same distribution as that of the square of a single normally distributed quantity. Thus $\Delta\chi^2_\nu < 1$ occurs 68.3 percent of the time (1-σ for the normal distribution), $\Delta\chi^2_\nu < 4$ occurs 95.4 percent of the time (2-σ for the normal distribution), $\Delta\chi^2_\nu < 9$ occurs 99.73 percent of the time (3-σ for the normal distribution), etc. In this manner you find the $\Delta\chi^2_\nu$ that corresponds to your desired confidence level. (Additional values are given in the accompanying table.)

Let $\delta\mathbf{a}$ be a change in the parameters whose first component is arbitrary, δa_1, but the rest of whose components are chosen to minimize the $\Delta\chi^2$. Then Theorem D applies. The value of $\Delta\chi^2$ is given in general by

$$\Delta\chi^2 = \delta\mathbf{a} \cdot [\alpha] \cdot \delta\mathbf{a} \qquad (15.6.1)$$

which follows from equation (15.5.8) applied at $\chi^2_{\min}$ where $\beta_k = 0$. Since $\delta\mathbf{a}$ by hypothesis minimizes χ^2 in all but its first component, the second through Mth components of the normal equations (15.5.9) continue to hold. Therefore, the solution of (15.5.9) is

$$\delta\mathbf{a} = [\alpha]^{-1} \cdot \begin{pmatrix} c \\ 0 \\ \vdots \\ 0 \end{pmatrix} = [C] \cdot \begin{pmatrix} c \\ 0 \\ \vdots \\ 0 \end{pmatrix} \qquad (15.6.2)$$

where c is one arbitrary constant that we get to adjust to make (15.6.1) give the desired left-hand value. Plugging (15.6.2) into (15.6.1) and using the fact that $[C]$ and $[\alpha]$ are inverse matrices of one another, we get

$$c = \delta a_1 / C_{11} \qquad \text{and} \qquad \Delta\chi^2_\nu = (\delta a_1)^2 / C_{11} \qquad (15.6.3)$$

or

$$\delta a_1 = \pm\sqrt{\Delta\chi^2_\nu}\,\sqrt{C_{11}} \qquad (15.6.4)$$

At last! A relation between the confidence interval $\pm\delta a_1$ and the formal standard error $\sigma_1 \equiv \sqrt{C_{11}}$. Not unreasonably, we find that the 68 percent confidence interval is $\pm\sigma_1$, the 95 percent confidence interval is $\pm 2\sigma_1$, etc.

$\Delta\chi^2$ as a Function of Confidence Level and Degrees of Freedom						
	ν					
p	1	2	3	4	5	6
68.3%	1.00	2.30	3.53	4.72	5.89	7.04
90%	2.71	4.61	6.25	7.78	9.24	10.6
95.4%	4.00	6.17	8.02	9.70	11.3	12.8
99%	6.63	9.21	11.3	13.3	15.1	16.8
99.73%	9.00	11.8	14.2	16.3	18.2	20.1
99.99%	15.1	18.4	21.1	23.5	25.7	27.8

These considerations hold not just for the individual parameters a_i, but also for any linear combination of them: If

$$b \equiv \sum_{k=1}^{M} c_i a_i = \mathbf{c} \cdot \mathbf{a} \qquad (15.6.5)$$

then the 68 percent confidence interval on b is

$$\delta b = \pm\sqrt{\mathbf{c} \cdot [C] \cdot \mathbf{c}} \qquad (15.6.6)$$

However, these simple, normal-sounding numerical relationships do *not* hold in the case $\nu > 1$ [3]. In particular, $\Delta\chi^2 = 1$ is not the boundary, nor does it project onto the boundary, of a 68.3 percent confidence region when $\nu > 1$. If you want to calculate not confidence intervals in one parameter, but confidence ellipses in two parameters jointly, or ellipsoids in three, or higher, then you must follow the following prescription for implementing Theorems C and D above:

- Let ν be the number of fitted parameters whose joint confidence region you wish to display, $\nu \leq M$. Call these parameters the "parameters of interest."
- Let p be the confidence limit desired, e.g., $p = 0.68$ or $p = 0.95$.
- Find Δ (i.e., $\Delta\chi^2$) such that the probability of a chi-square variable with ν degrees of freedom being less than Δ is p. For some useful values of p and ν, Δ is given in the table. For other values, you can use the routine gammq and a simple root-finding routine (e.g., bisection) to find Δ such that gammq$(\nu/2, \Delta/2) = 1 - p$.
- Take the $M \times M$ covariance matrix $[C] = [\alpha]^{-1}$ of the chi-square fit. Copy the intersection of the ν rows and columns corresponding to the parameters of interest into a $\nu \times \nu$ matrix denoted $[C_{\text{proj}}]$.
- Invert the matrix $[C_{\text{proj}}]$. (In the one-dimensional case this was just taking the reciprocal of the element C_{11}.)
- The equation for the elliptical boundary of your desired confidence region in the ν-dimensional subspace of interest is

$$\Delta = \delta\mathbf{a}' \cdot [C_{\text{proj}}]^{-1} \cdot \delta\mathbf{a}' \qquad (15.6.7)$$

where $\delta\mathbf{a}'$ is the ν-dimensional vector of parameters of interest.

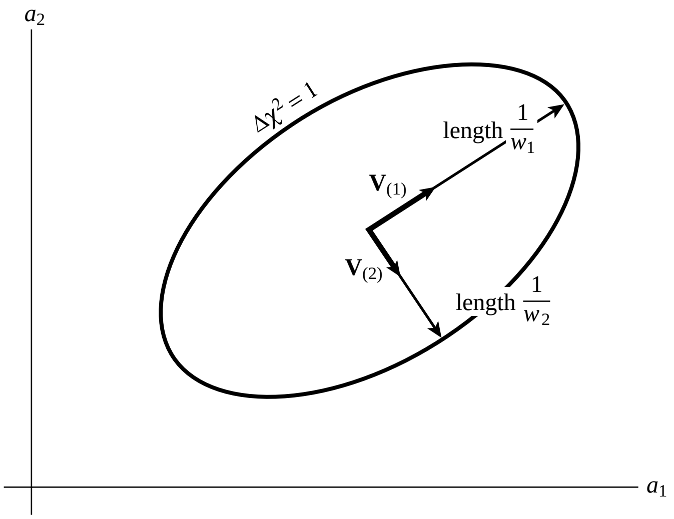


Figure 15.6.5. Relation of the confidence region ellipse $\Delta\chi^2 = 1$ to quantities computed by singular value decomposition. The vectors $\mathbf{V}_{(i)}$ are unit vectors along the principal axes of the confidence region. The semi-axes have lengths equal to the reciprocal of the singular values w_i. If the axes are all scaled by some constant factor α, $\Delta\chi^2$ is scaled by the factor α^2.

If you are confused at this point, you may find it helpful to compare Figure 15.6.4 and the accompanying table, considering the case $M = 2$ with $\nu = 1$ and $\nu = 2$. You should be able to verify the following statements: (i) The horizontal band between C and C' contains 99 percent of the probability distribution, so it is a confidence limit on a_2 alone at this level of confidence. (ii) Ditto the band between B and B' at the 90 percent confidence level. (iii) The dashed ellipse, labeled by $\Delta\chi^2 = 2.30$, contains 68.3 percent of the probability distribution, so it is a confidence region for a_1 and a_2 jointly, at this level of confidence.

Confidence Limits from Singular Value Decomposition

When you have obtained your χ^2 fit by singular value decomposition (§15.4), the information about the fit's formal errors comes packaged in a somewhat different, but generally more convenient, form. The columns of the matrix $\mathbf{V}$ are an orthonormal set of M vectors that are the principal axes of the $\Delta\chi^2 =$ constant ellipsoids. We denote the columns as $\mathbf{V}_{(1)} \dots \mathbf{V}_{(M)}$. The lengths of those axes are inversely proportional to the corresponding singular values $w_1 \dots w_M$; see Figure 15.6.5. The boundaries of the ellipsoids are thus given by

$$\Delta\chi^2 = w_1^2(\mathbf{V}_{(1)} \cdot \delta\mathbf{a})^2 + \cdots + w_M^2(\mathbf{V}_{(M)} \cdot \delta\mathbf{a})^2 \tag{15.6.8}$$

which is the justification for writing equation (15.4.18) above. Keep in mind that it is *much* easier to plot an ellipsoid given a list of its vector principal axes, than given its matrix quadratic form!

The formula for the covariance matrix $[C]$ in terms of the columns $\mathbf{V}_{(i)}$ is

$$[C] = \sum_{i=1}^{M} \frac{1}{w_i^2} \mathbf{V}_{(i)} \otimes \mathbf{V}_{(i)} \tag{15.6.9}$$

or, in components,

$$C_{jk} = \sum_{i=1}^{M} \frac{1}{w_i^2} V_{ji} V_{ki} \qquad (15.6.10)$$

CITED REFERENCES AND FURTHER READING:

Efron, B. 1982, *The Jackknife, the Bootstrap, and Other Resampling Plans* (Philadelphia: S.I.A.M.). [1]

Efron, B., and Tibshirani, R. 1986, *Statistical Science* vol. 1, pp. 54–77. [2]

Avni, Y. 1976, *Astrophysical Journal*, vol. 210, pp. 642–646. [3]

Lampton, M., Margon, M., and Bowyer, S. 1976, *Astrophysical Journal*, vol. 208, pp. 177–190.

Brownlee, K.A. 1965, *Statistical Theory and Methodology*, 2nd ed. (New York: Wiley).

Martin, B.R. 1971, *Statistics for Physicists* (New York: Academic Press).

15.7 Robust Estimation

The concept of *robustness* has been mentioned in passing several times already. In §14.1 we noted that the median was a more robust estimator of central value than the mean; in §14.6 it was mentioned that rank correlation is more robust than linear correlation. The concept of outlier points as exceptions to a Gaussian model for experimental error was discussed in §15.1.

The term "robust" was coined in statistics by G.E.P. Box in 1953. Various definitions of greater or lesser mathematical rigor are possible for the term, but in general, referring to a statistical estimator, it means "insensitive to small departures from the idealized assumptions for which the estimator is optimized." [1,2] The word "small" can have two different interpretations, both important: either fractionally small departures for all data points, or else fractionally large departures for a small number of data points. It is the latter interpretation, leading to the notion of outlier points, that is generally the most stressful for statistical procedures.

Statisticians have developed various sorts of robust statistical estimators. Many, if not most, can be grouped in one of three categories.

M-estimates follow from maximum-likelihood arguments very much as equations (15.1.5) and (15.1.7) followed from equation (15.1.3). M-estimates are usually the most relevant class for model-fitting, that is, estimation of parameters. We therefore consider these estimates in some detail below.

L-estimates are "linear combinations of order statistics." These are most applicable to estimations of central value and central tendency, though they can occasionally be applied to some problems in estimation of parameters. Two "typical" L-estimates will give you the general idea. They are (i) the median, and (ii) *Tukey's trimean,* defined as the weighted average of the first, second, and third quartile points in a distribution, with weights 1/4, 1/2, and 1/4, respectively.

R-estimates are estimates based on rank tests. For example, the equality or inequality of two distributions can be estimated by the *Wilcoxon test* of computing the mean rank of one distribution in a combined sample of both distributions. The Kolmogorov-Smirnov statistic (equation 14.3.6) and the Spearman rank-order

Statistical Data Analysis in the Computer Age

Bradley Efron; Robert Tibshirani

Science, New Series, Vol. 253, No. 5018. (Jul. 26, 1991), pp. 390-395.

Stable URL:
http://links.jstor.org/sici?sici=0036-8075%2819910726%293%3A253%3A5018%3C390%3ASDAITC%3E2.0.CO%3B2-2

Science is currently published by American Association for the Advancement of Science.

Articles

Statistical Data Analysis in the Computer Age

BRADLEY EFRON AND ROBERT TIBSHIRANI

Most of our familiar statistical methods, such as hypothesis testing, linear regression, analysis of variance, and maximum likelihood estimation, were designed to be implemented on mechanical calculators. Modern electronic computation has encouraged a host of new statistical methods that require fewer distributional assumptions than their predecessors and can be applied to more complicated statistical estimators. These methods allow the scientist to explore and describe data and draw valid statistical inferences without the usual concerns for mathematical tractability. This is possible because traditional methods of mathematical analysis are replaced by specially constructed computer algorithms. Mathematics has not disappeared from statistical theory. It is the main method for deciding which algorithms are correct and efficient tools for automating statistical inference.

Most scientists face problems of data analysis: What data should I collect? What can I conclude from my data? How far can I trust the conclusions? Statistics is the mathematical science that deals with these questions. Some statistical methods, such as linear regression, hypothesis testing, standard errors, and confidence intervals, have become familiar in the scientific literature over time. Most of the "classical" methods were developed between 1920 and 1950, by scientists such as R. A. Fisher, J. Neyman, and H. Hotelling, who were senior colleagues to statisticians still active today.

The 1980s produced a rising curve of new statistical theory and methods based on the power of electronic computation. Today's data analyst can afford to expend more computation on a single problem than the world's yearly total of statistical computation in the 1920s. How can such computational wealth be spent wisely, in a way that genuinely adds to the classical methodology without merely elaborating it? Answering this question has become a dominant theme of modern statistical theory.

Some promising developments in computer-intensive statistical methodology are described in this article. The examples involve bootstrap methods, nonparametric regression, generalized additive models, and classification and regression trees. The presentation here is mainly descriptive, without much mathematical development. However, we will try to indicate the crucial role that mathematics plays in tying the new statistical methods to their classical antecedents.

The Bootstrap

In almost every statistical data analysis, on the basis of a data set x we calculate a statistic $t(\mathbf{x})$ for the purpose of estimating some quantity of interest. Box 1 shows the cholesterol reduction scores of nine men after taking cholestyramine; the scores are an ordered random sample from the scores of 164 men (*1*). The data set x could be these nine scores, and $t(\mathbf{x})$ could be their mean value $\bar{x} = 28.58$, intended as an estimate of the true mean value of the cholesterol reduction scores. (The true mean value is the mean we would obtain if we observed a much larger set of scores.) The following fundamental question arises: how accurate is $t(\mathbf{x})$?

−21.0	3.25	10.75	13.75	32.50	39.50	41.75	56.75	80.0

Box 1

This question has a simple answer if $t(\mathbf{x})$ is the mean $\bar{x}$ of numbers $x_1, x_2, \ldots, x_n$. Then the standard error of $\bar{x}$, its root- mean-square error, is estimated by a formula made famous in elementary statistics courses

$$se(\bar{x}) = \left\{ \sum_{i=1}^{n} (x_i - \bar{x})^2/[n(n-1)] \right\}^{1/2} \qquad (1)$$

For the nine numbers in Box 1, Eq. 1 gives 10.13. The estimate of the true cholesterol reduction mean would usually be expressed as 28.58 ± 10.13, or perhaps 28.58 ± 10:13z, where z is some constant, such as 1.645 or 1.960, relating to areas under a bell-shaped curve. With $z = 1.645$, the interval has approximately 90% chance of containing the true mean value. In other words, it is an approximate 90% confidence interval.

The bootstrap (*2*) was introduced primarily as a device for extending Eq. 1 to estimators other than the mean. For example suppose $t(\mathbf{x})$ is the 25% trimmed mean, $\bar{x}\{0.25\}$, defined as the average of the middle 50% of the data. We order the observations $x_1, x_2, \ldots, x_n$, discard the lower and upper 25% of them, and take the mean of the remaining 50%. Interpolation is required for cases where $0.25n$ is not an integer. For the cholesterol data

$$\bar{x}(0.25) = \frac{3/4(10.75) + (13.75) + (32.5) + (39.5) + 3/4(41.25)}{3/4 + 1 + 1 + 1 + 3/4} = 27.81 \qquad (2)$$

There is no neat algebraic formula such as Eq. 1 for the standard error of a trimmed mean or for almost any estimate other than the mean. That is why the mean is so popular in statistics courses. In lieu of a formula, the bootstrap uses computational power to get a numerical estimate of the standard error. The bootstrap algorithm depends on the notion of a bootstrap sample, which is a sample of

B. Efron is in the Department of Statistics, Stanford University, Stanford, CA 94305. R. Tibshirani is in the Department of Preventive Medicine and Biostatistics and Department of Statistics, University of Toronto, Toronto, Ontario, Canada M5S 1A8.

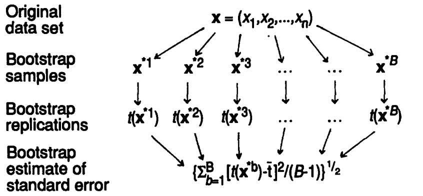


Fig. 1. A diagram of the bootstrap algorithm for estimating the standard error of a statistic $t(\mathbf{x})$; each of the B bootstrap samples is a random sample of size n drawn with replacement from the original data set $(x_1, x_2, \ldots, x_n)$; $\bar{t}$ is the average of the B bootstrap replications $t(\mathbf{x}^{*b})$. Most of the computation occurs at step 2, bootstrap replications.

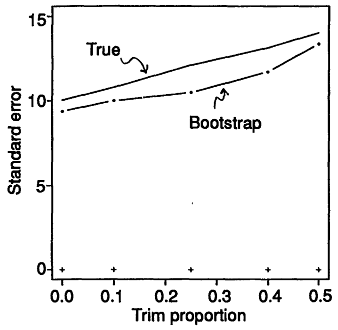


Fig. 2. Bootstrap estimates of standard error for five different trimmed means $\bar{x}\{p\}$, $p = 0, 0.10, 0.25, 0.40, 0.5$, applied to the cholesterol data of Box 1, based on $B = 400$ bootstrap samples. Also shown is the true standard error of $\bar{x}\{p\}$, obtained by taking random samples of size 9 from the population of 164 cholesterol reduction scores. In this case, the bootstrap correctly indicates that $\bar{x}\{0\}$, the ordinary mean, gives the smallest standard error.

size n drawn with replacement from the original data set $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. The bootstrap sample is denoted $\mathbf{x}^* = (x_1^*, x_2^*, \ldots, x_n^*)$. Each x_1^* is one of the original x values, randomly selected (perhaps $x_1^* = x_7, x_2^* = x_5, x_3^* = x_5, x_4^* = x_9, x_5^* = x_7$, and so forth). The name "bootstrap" refers to the use of the original data set to generate new data sets $\mathbf{x}^*$.

The bootstrap estimate of standard error for $\bar{x}\{0.25\}$ is computed as follows: (i) a large number B of independent bootstrap samples, each of size n, is generated using a random number device, (ii) the 25% trimmed mean is calculated for each bootstrap sample, and (iii) the empirical standard deviation of the B bootstrap trimmed means is the bootstrap estimate of standard error for $\bar{x}\{0.25\}$. A schematic diagram of the bootstrap algorithm, applied to a general statistic $t(\mathbf{x})$, is shown in Fig. 1.

These bootstrap estimates of standard error for the 25% trimmed mean, applied to the cholesterol data, were obtained for different values of B: $B = 25$, bootstrap estimate $= 12.44$; $B = 50$, bootstrap estimate $= 9.71$; $B = 100$, bootstrap estimate $= 11.50$; $B = 200$, bootstrap estimate $= 10.70$; $B = 400$, bootstrap estimate $= 10.48$. Ideally, B would go to infinity. However, randomness in the bootstrap standard error that comes from using a finite value of B is usually negligible for B greater than 200; that is, this randomness would be small relative to the randomness caused by variations in the original data set $\mathbf{x}$. Even values of B as small as 25 often give satisfactory results. This can be important if the statistic $t(\mathbf{x})$ is difficult to compute because the bootstrap algorithm requires about B times as much computation as $t(\mathbf{x})$.

The bootstrap algorithm can be applied to almost any statistical estimation problem: (i) The individual data points x_i need not be single numbers; they can be vectors, matrices, or more general quantities, such as maps or graphs. (ii) The statistic $t(\mathbf{x})$ can be anything at all, as long as we can compute $t(\mathbf{x}^*)$ for every bootstrap data set $\mathbf{x}^*$. (iii) The data set $\mathbf{x}$ does not have to be a simple random sample from a single distribution. Other data structures, for example, regression models, time series, or stratified samples, can be accommodated by appropriate changes in the definition of a bootstrap sample. (iv) Measures of statistical accuracy other than the standard error, for instance, biases, mean absolute value errors, and confidence intervals, can be calculated at the final stage of the algorithm (*3*). The example below illustrates some of these points.

There is one statistic $t(\mathbf{x})$ for which one does not need the computer to calculate the bootstrap standard error, namely the mean $\bar{x}$. In this case, it can be proved that, as B goes to infinity, the bootstrap standard error estimate goes to $\sqrt{(n - 1)/n}$ times Eq. 1. The factor $\sqrt{(n - 1)/n}$, which equals 0.943 for $n = 9$, could be removed by redefinition of the last step of the bootstrap algorithm, but there is no general advantage to doing so. For the statistic $\bar{x}$, using the bootstrap algorithm gives about the same result as Eq. 1.

At a deeper level, the logic that makes Eq. 1 a reasonable assessment of standard error for $\bar{x}$ applies equally well to the bootstrap as an assessment of standard error for a general statistic $t(\mathbf{x})$. In both cases, the standard error of the statistic of interest is assessed by the true standard error that would apply if the unknown probability distribution yielding the data exactly equaled the empirical distribution of the data. The efficacy of this simple estimation principle has been verified by a large amount of theoretical work in the statistics literature of the past decade; see (*3*–*5*) and references within.

Why use a trimmed mean rather than $\bar{x}$? The theory of robust statistics, developed since 1960, shows that if the data $\mathbf{x}$ comes from a long-tailed probability distribution, then the trimmed mean can be substantially more accurate than $\bar{x}$. That is, it can have substantially smaller standard error (*6*, *7*). In practice, however, one does not know a priori if the true probability distribution is long-tailed. The bootstrap can help answer this question.

The bootstrap estimates of standard error for five different trimmed means $\bar{x}\{p\}$, where p is the proportion of the data trimmed off each end of the sample before the mean is taken are shown in Fig. 2. (So $\bar{x}\{0\}$ is $\bar{x}$, the usual mean, whereas $\bar{x}\{0.5\}$ is the median.) These were computed with the use of the bootstrap algorithm in Fig. 1 ($B = 400$), except that at step 2, bootstrap replication, five different statistics were evaluated for each bootstrap sample $\mathbf{x}^*$, namely $\bar{x}\{0\}$, $\bar{x}\{0.10\}$, $\bar{x}\{0.25\}$, $\bar{x}\{0.40\}$, and $\bar{x}\{0.50\}$.

According to the bootstrap standard errors in Fig. 2, the ordinary mean has the smallest standard error among the five trimmed means. This seems to indicate that there is no advantage to trimming for this particular data set.

The nine cholesterol reduction scores in Box 1 were a random sample from a larger data set: 164 scores, corresponding to the 164 men in the Stanford arm of a large clinical trial designed to test the efficiency of the cholesterol-reducing drug cholestyramine (*8*). With all of this extra data available, the bootstrap standard errors can be checked. The solid line in Fig. 2 indicates the true standard errors for each of the five trimmed means, that is, the standard errors of random samples of size 9 taken from the population of 164 scores.

We see that the true standard errors confirm the bootstrap conclusion that the ordinary mean is the estimator of choice in this case. The main point here is that the bootstrap estimates use only the data in Box 1, whereas the true standard errors require extra data that usually is not available in a real data analysis problem.

Theoretical work on properties of the bootstrap is proceeding at a vigorous pace (*4*, *5*). We have emphasized standard errors here, but the main theoretical thrust has been toward confidence intervals. Getting dependable confidence intervals from bootstrap calculations is challenging, in theory and in practice, but progress on both fronts has been considerable.

Nonparametric Regression

The data for all 164 men in the Stanford arm of the cholestyramine experiment are shown in Fig. 3. The vertical axis plots the cholesterol reduction scores, nine of which appear in Box 1. The horizontal axis plots compliance, the proportion of the intended dose each man actually took (measured by counting of the packets of unused cholestyramine returned to the clinic). Better compliance tends to be associated with a greater reduction in cholesterol, as might be hoped.

The smooth curve in Fig. 3 is a quadratic regression curve fit to the 164 data points. In other words, it is the quadratic function of compliance that minimizes the sum of the 164 squared distances from the curve to the data points, where distance is measured in the vertical direction. Least-squares regression is a classical estimation method dating back to Gauss and Legendre in the early 1800s (*9*). The height of the quadratic curve at 60% compliance is 27.72 ± 3.08. The standard error 3.08 is provided by a formula much like Eq. 1, which is not surprising because the average $\bar{x}$ is the simplest example of a least-squares estimate.

The value 27.72 estimates the true amount of cholesterol reduction at the average compliance (60%), a quantity of particular importance in assessing the true cholesterol-reducing powers of cholestyramine (*8*). One might worry that a quadratic function of compliance does not accurately model cholesterol reduction as a function of compliance. If not, the estimate 27.72 will be biased, a form of statistical error not included in the formula that gave 3.08.

The irregular curve in Fig. 3 was obtained using loess (pronounced "low ess") (*10*), a computer-based fitting method that does not attempt to fit a simple model, like a quadratic curve, over the entire compliance range. Instead, loess fits a series of local regression curves for different values of compliance, in each case using only data points near the compliance value of interest.

Loess works in the following way (Fig. 4). First, a window of points (the shaded region) closest to the target point (arrow) is formed; in this case, the window contains the nearest 20% of the data points. Then a smooth weight function (dotted curve) known as the tricube function is constructed so that it is highest at the target point and falls to zero at the edges of the shaded region. Finally, a weighted linear regression (dashed line) is estimated for the points in the shaded region, with the weights determined by the tricube function. This process defines the estimate at the target point. Repeating the process for all possible target points gives the solid curve in Fig. 4. This curve is called a nonparametric regression estimate because it does not assume a particular parametric form (such as quadratic) for the regression.

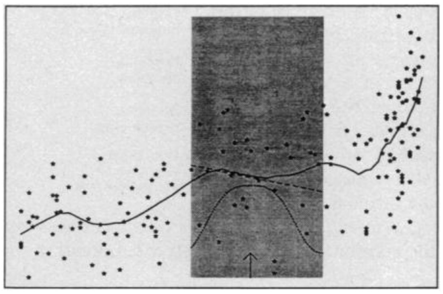


Fig. 4. How the loess smoother works. The shaded region indicates the window of points around the target point (arrow). A weighted linear regression (dashed line) is computed, with weights given by the tricube function (dotted curve). Repetition of this process for all target points gives the solid curve.

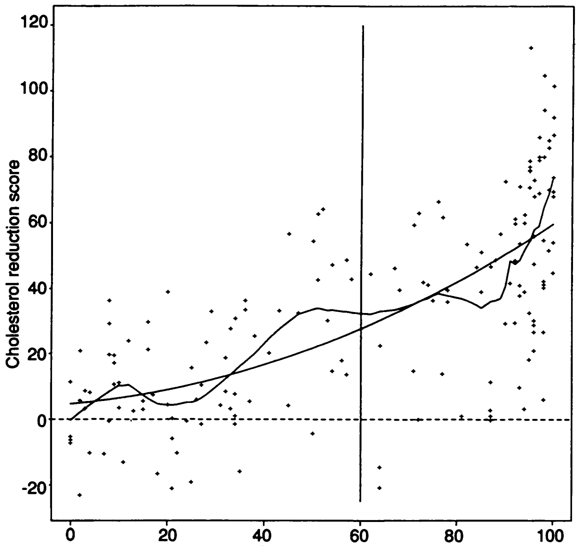


Fig. 3. Cholesterol reduction scores of 164 men in the Stanford arm of experiment LRC-CPPT plotted against compliance, measured as the percentage of intended cholestyramine dose that was actually taken. The average compliance was 60%. The smooth curve is a quadratic regression fit to the 164 points by least squares; the irregular curve is loess, a scatterplot smoother that uses local regressions fit to a moving window of 20% of the points.

The height of the loess curve at 60% compliance is 32.38, which indicates substantially greater cholesterol-reducing power than the quadratic estimate 27.72. But how dependable is the loess answer? It is bound to be less biased than the quadratic estimate because it makes fewer assumptions about the form of the dependence between compliance and cholesterol reduction. However, one cannot assess its value as an estimate without some idea of its standard error, and there is nothing like Eq. 1 for loess.

The bootstrap algorithm for standard error can be applied exactly as described in Fig. 1. Now $n = 164$, and each x_i is the pair of numbers (compliance, cholesterol reduction score) for patient i. The function $t(\mathbf{x}^*)$ takes any data set $\mathbf{x}^*$ consisting of 164 pairs, applies the loess algorithm to it, and reads off the height of the loess function evaluated at 60% compliance. Knowledge of the complicated details of the loess algorithm, is not necessary. All one need do is call the same loess subroutine that gave the estimate 32.38 for the original data set.

The bootstrap algorithm was run with $B = 50$, and the first 15 of the bootstrap loess curves are shown (Fig. 5). There is considerable variability in the intercepts of these curves at 60% compliance. The bootstrap estimate of standard error for the intercept, based on all 50 bootstrap replications, was 5.71, nearly twice the standard error of the quadratic fit. On balance, the quadratic estimate should probably be preferred in this case. It would have to have an unusually large bias to undo its superiority in standard error.

Generalized Additive Models

Nonparametric regression procedures like loess can be used to model complex data in a flexible manner. This allows the data analyst to make new discoveries about the data. As an example, Williams and colleagues from Toronto's Hospital for Sick Children collected data on the survival of 497 infants after cardiac surgery for heart defects, for the years 1983 to 1988 (*11*). This was an observational study rather than randomized clinical trials. A warm-blood cardioplegia (WBC) arrest of the heart, thought to improve chances for survival, was introduced in February 1988. The procedure was used on those infants for whom it was thought appropriate and only by those surgeons who liked the procedure. The main question was whether the introduction of WBC improved survival relative to the standard treatment; the importance of risk factors age (in days) and weight (in kilograms) was also of interest. Of the 57 infants who received WBC, 7 died; of the 440 infants who received the standard procedure, 133 died. WBC seemed to be improving the survival rate considerably.

A linear logistic model is the standard way to approach problems of this kind. This model assumes that the log of the odds ratio, probability (death)/probability (survival), is a linear function of the

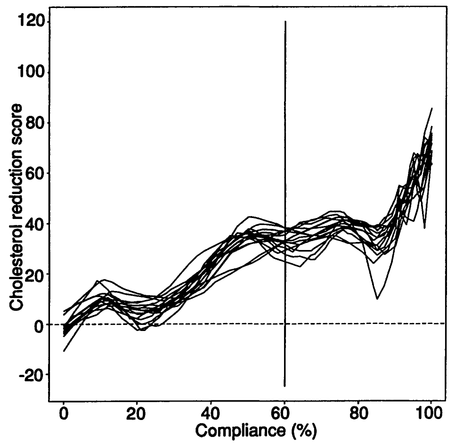


Fig. 5. The first 15 of the 50 bootstrap loess curves, based on the data for 164 men (Fig. 3). The intercept at 60% compliance has empirical standard deviation 5.71, based on all $B = 50$ bootstrap replications.

age and weight of the infant, plus a term indicating if WBC was used. The results of fitting a linear logistic model to these data suggested that WBC had a strong beneficial effect on survival, with an odds ratio of 3.8 ± 1.8. Thus the odds of dying were 3.8 times as high with the standard treatment as with WBC. Furthermore, the risk of death decreased with weight, but the age of the infant did not have a significant effect on survival.

Using nonparametric regression procedures, one can learn more from the data. Rather than assuming that the log-odds of survival is a linear function of age and weight, one assumes only that it is a sum of a smooth function of age and a smooth function of weight. This is an example of a generalized additive model (*12*). The data analyst is not required to specify the form of these smooth functions (such as linear, quadratic, or logarithmic); instead, the form of each of these functions is estimated by a computer-intensive algorithm that makes repeated use of nonparametric regression procedure such as loess.

The curves that resulted from a fit of the generalized additive model are shown in Fig. 6. The shaded regions are approximate confidence bands for the curves. The left curve, for example, represents the log-odds of death as a function of the weight of the infant. The log-odds is highest for the lighter infants (~1) and lowest for the heavier infants (~−3). Hence the odds ratio for light versus heavy infants is the exponential of $[1 - (-3)] \approx 55$. The log-odds does not start to decrease until the infant is at least 3 or 3.5 kg.

The log-odds curve for age is, perhaps, surprising. The operation is least dangerous for infants who are about 200 days old and is more risky for younger or older infants. In a traditional logistic regression, these curves might be forced to be straight lines, and one would not discover the effects seen in these pictures. The danger of oversimplified regressions becomes more acute in more complicated situations where there are large numbers of explanatory variables.

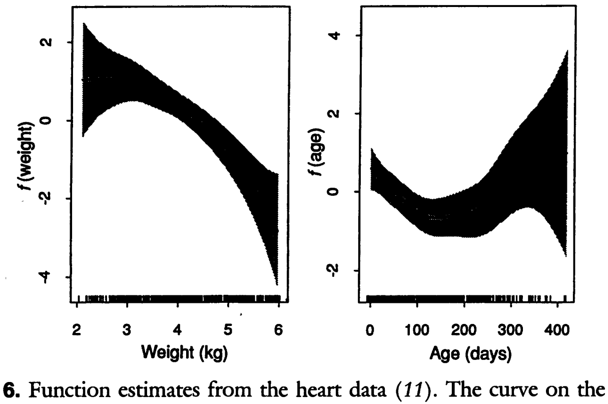


Fig. 6. Function estimates from the heart data (*11*). The curve on the left represents the log-odds of death as a function of weight; the curve on the right is the log-odds of death as a function of age. The shaded regions are approximate confidence bands.

The generalized additive model also provides an assessment of WBC. The estimated odds ratio for the standard treatment versus WBC was 4.2 ± 1.9, almost the same as the linear logistic estimate.

Modern statistical tools that are powerful and flexible also tend to be more difficult to analyze mathematically. For example, because of the complexity of the generalized additive model, many approximations were used to obtain the value 1.9 for the standard error reported above. With so much at stake medically, some additional effort to check the accuracy of this value is worthwhile. The bootstrap can be used to accomplish this. A bootstrap sample is created by random drawing of 497 patients with replacement from the original set of 497 patients. A generalized additive model is fit to the bootstrap sample and the estimated odds ratio for WBC is recorded. This entire process is repeated a large number of times, in this case 100. The standard deviation of the 100 odds ratios equaled 2.0, just slightly larger than the approximate value 1.9. The agreement of the bootstrap with the approximate standard error strengthens our belief in both of them.

Generalized additive models can be applied in a wide variety of settings, providing a flexible tool for discovering the underlying structure of scientific processes. Although the algorithm to fit these models required a mainframe computer 10 years ago, now the computations can be carried out on a personal computer. Generalized additive models are just one example of flexible modeling tools that exploit the power of the computer. The development of such tools is an active area of statistical research.

Classification and Regression Trees

In an experiment designed to provide information about the causes of duodenal ulcers (*13*), one of 56 model alkyl nucleophiles was administered to each of a sample of 745 rats. Each rat was later autopsied to check for the development of duodenal ulcer and the outcome was classified as 1, 2, or 3 in increasing order of severity. There were 535 class 1, 90 class 2, and 120 class 3 outcomes. The objective in the analysis of these data was to ascertain which of 67 characteristics of these compounds was associated with the development of duodenal ulcers.

The CART (Classification and Regression Trees) method (*14*) is a computer-intensive approach to this problem. When applied to this data, CART produced the classification tree shown in Fig. 7.

At each node of the tree a question is asked; data points for which the answer is "yes" are assigned to the left branch and other data points are assigned to the right branch. The leaves of the tree in Fig. 7 are called terminal nodes. Each observation is assigned to one of the terminal nodes on the basis of the answers to the questions. For example, a rat that received a compound with dipole moment ≤3.56 D and melting point >98.1°C would go left, then right, and would end up in the terminal node [13, 7, 41]. Triplets of numbers such as [13, 7, 41] below each terminal node number indicate the membership at that node, that is, 13 class 1, 7 class 2, and 41 class 3 observations.

In the CART procedure, each terminal node is assigned a class (1, 2, or 3). The most obvious way to assign classes to the terminal nodes is to use a majority rule and assign the class that is most numerous in the node. With a majority rule, node [13, 7, 41] would be assigned to class 3 and all of the other terminal nodes would be assigned to class 1. In this study, however, the investigators decided that it would be less desirable to misclassify an animal with a severe ulcer than one with a milder ulcer, and hence they prescribed a higher penalty to errors of the former type. Through the use of the prescribed penalties, a best rule for each terminal node can then be worked out. The assigned class is underlined at each terminal node in Fig. 7; for example, the node at the bottom left ([10, 0, 5]) has the number 5 underlined and hence is a class 3 node.

We can summarize the tree as follows. The top (root) node was split on dipole moment. A high dipole moment indicates the presence of electronegative groups. This split separates the class 1 and 2 compounds: the ratio of class 2 to class 1 in the right split (66/180) is more than five times as large as the ratio in the left split (24/355). However, the class 3 compounds are divided equally, 60 on each side of the split. If, in addition, the sum of squared atomic charges is low, then CART finds that all compounds are class 1. Hence ionization is a major determinant of biologic action in compounds with high dipole moments. Moving further down the right side of the tree, the solubility in octanol then partially separates class 3 from class 2 compounds. High octanol solubility probably reflects the ability to cross membranes and to enter the central nervous system.

On the left side of the root node, compounds with low dipole moment and high melting point were found to be class 3 (severe). Compounds at this terminal node are related to cysteamine (2-aminoethanethiol). Compounds with low melting points and high polarizability, all thiols in this study, were classified as class 2 or 3, with the partition coefficient separating these two classes. Of those chemicals with low polarizability, those of high density were classified as class 1. These chemicals have high molecular weight and volume, and this terminal node contains the highest number of observations. The low-density side of the split is composed of all short chain amines.

In statistical terminology, the data set of 745 observations is called a learning sample. It is easy to work out the misclassification rate for each class when the tree of Fig. 7 is applied to the learning sample. Looking at the terminal nodes that predict classes 2 or 3, the number of errors for class 1 is 13 + 89 + 50 + 10 + 25 + 25 = 212, so the apparent misclassification rate for class 1 is 212/535 = 39.6%. Similarly, the apparent misclassification rates for classes 2 and 3 are 56.7% and 18.3%. "Apparent" is an important qualifier here because misclassification rates in the learning sample can be badly biased downward, as discussed below.

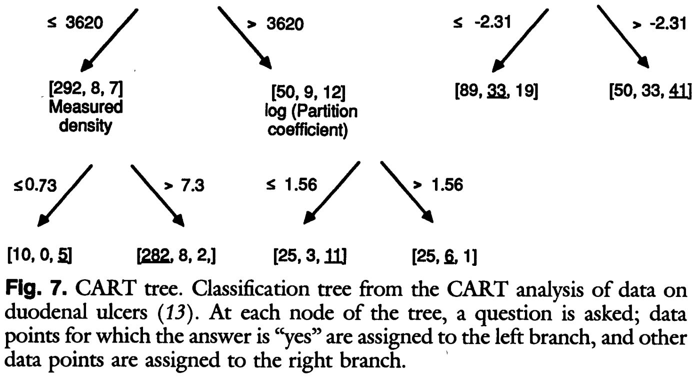


Fig. 7. CART tree. Classification tree from the CART analysis of data on duodenal ulcers (*13*). At each node of the tree, a question is asked; data points for which the answer is "yes" are assigned to the left branch, and other data points are assigned to the right branch.

How does CART build a tree like that in Fig. 7? CART is a fully automatic procedure that chooses the splitting variables and splitting points that best discriminate between the outcome classes. For example, the split "dipole moment ≤3.56" was determined to best separate the data with respect to the outcome classes. CART chose both the splitting variable, dipole moment, and the splitting value, 3.56. Having found the first splitting rule, new splitting rules are selected for each of the two resulting groups, and this process is repeated.

Rather than stopping when the tree is some reasonable size, the inventors of CART discovered a better approach: a large tree is constructed and then pruned from the bottom. This latter approach is more effective in discovering interactions that involve several variables.

How large should the tree be? If we were to build a very large tree with only one observation in each terminal node, then the apparent misclassification rate would be 0%. However, this tree would probably poorly predict the outcomes for a new sample of rats because it is too much geared to the learning sample; in statistical terminology, it is overfit.

The tree of best size would have the lowest misclassification rate for some new data. Thus, if one had a second data set available (a test sample), one could apply the trees of various sizes to it and then choose the one with lowest misclassification rate.

In most situations, one does not have extra data to work with. Data is so precious that all of it is used to estimate the best possible tree. CART uses the method of cross-validation to choose the tree size; this method attempts to mimic the use of a test sample. It works by dividing the data into ten groups of equal size, building a tree on 90% of the data, and then assessing the tree's misclassification rate on the remaining 10% of the data. This is done for each of the ten groups in turn, and the total misclassification rate is computed over the ten runs. The best tree size is determined to be that tree size giving the lowest misclassification rate. This size is used in constructing the final tree from all of the data. The crucial feature of cross-validation is the separation of data for building and assessing the trees: each one-tenth of the data acts as a test sample for the other nine-tenths.

The process of cross-validation not only provides an estimate of the best tree size, it also gives a realistic estimate of the misclassification rate of the final tree. The learning sample misclassification rates computed above are often unrealistically low because the training sample is used both for building and for assessing the tree. For the tree of Fig. 7, the cross-validated misclassification rates were about 10% higher than the learning sampling misclassification rates. It is the cross-validated rates that provide an honest assessment of how effective the tree will be in classifying a new sample of animals.

The theory underlying cross-validation is closely related to the bootstrap. Current research involves hybrids of the bootstrap and cross-validation that outperform both of them in the assessment of error rates.

Conclusion

The methods we have discussed are modern versions of traditional statistical tools. Loess, generalized additive models, and CART are different ways to expand the scope of linear regression. The bootstrap and cross-validation are improved variants of the familiar error estimate in Eq. 1. All of these developments, and a host of others we have not mentioned, differ in one important way from their classical predecessors: they substitute computer algorithms for the traditional mathematical ways of getting a numerical answer. One immediate reward is freedom from the bell-shaped curve assumptions of the traditional approach. More importantly, the new methods free the scientist to choose statistical methodology appropriate to the problem at hand, rather than choosing on the basis of mathematical tractability.

None of this means that mathematics has disappeared from statistical theory, only that it is disappearing from routine statistical applications. The question of which computer-based method to use, and when to use it, is becoming a central concern of mathematical statistics.

REFERENCES AND NOTES

1. The data in Box 1, from the Stanford arm of the LRC-CPPT experiment, is courtesy of D. Feldman and J. Farquhar, Stanford University; see (7).
2. B. Efron, *Am. Stat.* **40**, 1 (1986).
3. _____ and R. Tibshirani, *Stat. Sci.* **1**, 54 (1986).
4. T. DiCiccio and J. Romano, *J. R. Stat. Soc. B* **50**, 338 (1988).
5. D. V. Hinkley, *ibid.*, p. 321.
6. P. Huber, *Robust Statistics* (Wiley, New York, 1981), p. 5.
7. F. Hampel, E. Ronchetti, P. Rousseeuw, W. Stahel, *Robust Statistics: The Approach Based on Influence Functions* (Wiley, New York, 1986), p. 29.
8. B. Efron and D. Feldman, *J. Am. Stat. Assoc.*, in press.
9. B. Efron, *SIAM Rev.* **30**, 421 (1988).
10. W. S. Cleveland, *J. Am. Stat. Assoc.* **74**, 829 (1979).
11. W. G. Williams *et al.*, *J. Thorac. Cardiovasc. Surg.*, in press.
12. T. Hastie and R. Tibshirani, *Generalized Additive Models* (Chapman and Hall, London, 1990), p. 136.
13. C. Giampaolo, A. Gray, R. Olshen, S. Szabo, *Proc. Natl. Acad. Sci. U.S.A.*, in press.
14. L. Breiman, J. H. Friedman, R. Olshen, C. J. Stone, *Classification and Regression Trees* (Wadsworth, Belmont, CA, 1984).
15. We thank R. Olshen for allowing us to use his CART example.

Enols and Other Reactive Species

YVONNE CHIANG AND A. JERRY KRESGE

Rapid advances in the chemistry of enols and other reactive species have been made possible recently by the development of methods for generating these short-lived substances in solution under conditions where they can be observed directly and their reactions can be monitored accurately. New laboratory techniques are described and a sample of the new chemistry they have made available is provided; special attention is given to ynols and ynamines and the remarkable effects that the carbon-carbon triple bonds of these substances have on their acid-base properties.

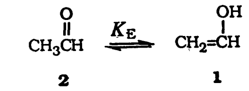
THE CHEMISTRY OF ENOLS IS CURRENTLY EXPERIENCING A renaissance (*1*) primarily because of the development of methods for generating these usually very reactive substances in solution under conditions where their reactions can be studied in detail. Such studies are worthwhile because enols and enolate ions are essential intermediates in many important reactions of carbonyl compounds, and a number of biological reactions also involve enol formation; if we wish to understand these processes, and through understanding to control them, we must understand the chemistry of enols.

We began work in this area by examining enol isomers of simple aldehydes and ketones. That work, however, soon led to the investigation of other reactive species, such as enols of carboxylic acids and their derivatives, ketenes, carbenes, ynols, and ynamines. The latter are especially fascinating substances: they are believed to exist in interstellar space and are postulated as prebiotic molecules. We have discovered that the carbon-carbon triple bond in ynols and ynamines exerts a remarkable influence on the acid-base properties of their hydroxyl and amino groups; theoretical calculations at the ab initio level have helped us understand the origins of this effect.

This article begins with an account of our work on enols and continues with a description of what we have learned about ynols and ynamines. Although the discussion is limited largely to research done in our own laboratory, we owe much to stimulation provided by the pioneering work of Guthrie *et al.* (*2*), Capon *et al.* (*3*), Dubois, Toullec, and co-workers (*4*), and Rappoport and co-workers (*5*), and we are indebted as well to an early review by Hart (*6*).

Generation of Enols

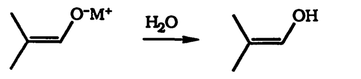
Simple enols such as vinyl alcohol, **1**, can be formed readily from their keto isomers, **2**, Eq. 1.



(1)

The reaction, however, is reversible, and the position of equilibrium generally lies strongly on the keto side; the amount of enol present at equilibrium is consequently seldom sufficient to permit direct observation, even by the most sensitive spectroscopic methods. Investigation of enol chemistry therefore requires generation of the enol in greater than the equilibrium amount in the medium of interest. We have developed a number of ways of accomplishing this in aqueous solution.

We first made enols by hydrolysis of their alkali metal salts, Eq. 2,



(2)

using solutions of these salts in aprotic solvents prepared by standard synthetic methodology (*7*). Addition of a small quantity of such a solution to a large amount of water resulted in a very fast oxygen-to-oxygen proton transfer and produced the enol in an essentially wholly aqueous medium. Conversion of the enol to its keto isomer then proceeded at a slower rate, which we could monitor accurately by following the marked change in the ultraviolet spectrum that accompanies the ketonization reaction.

This method of generating enols requires mixing two solutions and consequently cannot be applied to substances with lifetimes shorter than the mixing time. This limitation unfortunately excludes

The authors are in the Department of Chemistry, University of Toronto, Toronto, Ontario, Canada M5S 1A1.