15

SIMULATION-BASED ESTIMATION AND INFERENCE AND RANDOM PARAMETER MODELS

15.1 INTRODUCTION

Simulation based methods have become increasingly popular in econometrics. They are extremely computer intensive, but steady improvements in recent years in computation hardware and software have reduced that cost enormously. The payoff has been in the form of methods for solving estimation and inference problems that have previously been unsolvable in analytic form. The methods are used for two main functions. First, simulation based methods are used to infer the characteristics of random variables, including estimators, functions of estimators, test statistics, and so on, by sampling from their distributions. Second, simulation is used in constructing estimators that involve complicated integrals that do not exist in a closed form that can be evaluated. In such cases, when the integral can be written in the form of an expectation, simulation methods can be used to evaluate it to within acceptable degrees of approximation by estimating the expectation as the mean of a random sample. The technique of maximum simulated likelihood (MSL) is essentially a classical sampling theory counterpart to the hierarchical Bayesian estimator considered in Chapter 16. Since the celebrated paper of Berry, Levinsohn, and Pakes (1995), and a related literature advocated by McFadden and Train (2000), maximum simulated likelihood estimation has been used in a large and growing number of studies.

The following are three examples from earlier chapters that have relied on sumulation methods:

Example 15.1 Inferring the Sampling Distribution of the Least Squares Estimator

In Example 4.1, we demonstrated the idea of a sampling distribution by drawing several thousand samples from a population and computing a least squares coefficient with each sample. We then examined the distribution of the sample of linear regression coefficients. A histogram suggested that the distribution appeared to be normal and centered over the true population value of the coefficient.

Example 15.2 Bootstrapping the Variance of the LAD Estimator

In Example 4.5, we compared the asymptotic variance of the least absolute deviations (LAD) estimator to that of the ordinary least squares (OLS) estimator. The form of the asymptotic variance of the LAD estimator is not known except in the special case of normally distributed disturbances. We relied, instead, on a random sampling method to approximate features of the sampling distribution of the LAD estimator. We used a device (bootstrapping) that allowed us to draw a sample of observations from the population that produces the estimator. With that random sample, by computing the corresponding sample statistics, we can infer characteristics of the distribution such as its variance and its 2.5th and 97.5th percentiles which can be used to construct a confidence interval.

Example 15.3 Least Simulated Sum of Squares

Familiar estimation and inference methods, such as least squares and maximum likelihood, rely on "closed form" expressions that can be evaluated exactly [at least in principle—likelihood equations such as (14-4) may require an iterative solution]. Model building and analysis often require evaluation of expressions that cannot be computed directly. Familiar examples include expectations that involve integrals with no closed form such as the random effects nonlinear regression model presented in Section 14.9.2. The estimation problem posed there involved nonlinear least squares estimation of the parameters of

$$E[y_{it}|\mathbf{x}_{it},u_i] = h(\mathbf{x}_{it}'\boldsymbol{\beta} + u_i).$$

Minimizing the sum of squares,

$$S(\boldsymbol{\beta}) = \Sigma_i\Sigma_t[y_{it} - h(\mathbf{x}_{it}'\boldsymbol{\beta} + u_i)]^2,$$

is not feasible because u_i is not observed. In this formulation,

$$E[y|\mathbf{x}_{it}] = E_u E[y_{it}|\mathbf{x}_{it},u_i] = \int_u E[y_{it}|\mathbf{x}_{it},u_i]f(u_i)\mathrm{d}u_i,$$

so the feasible estimation problem would involve the sum of squares,

$$S^*(\boldsymbol{\beta}) = \Sigma_i\Sigma_t[y_{it} - \int_u h(\mathbf{x}_{it}'\boldsymbol{\beta} + u_i)\, f(u_i)\mathrm{d}u_i]^2.$$

When the function is linear and u_i is normally distributed, this is a simple problem—it reduces to ordinary linear least squares. If either condition is not met, then the integral generally remains in the estimation problem. Although the integral,

$$E_u[h(\mathbf{x}_{it}'\boldsymbol{\beta} + u_i)] = \int_u h(\mathbf{x}_{it}'\boldsymbol{\beta} + u_i)f(u_i)\mathrm{d}u_i,$$

cannot be computed, if a large sample of R observations from the population of u_i, i.e., u_{ir}, $r = 1,\ldots,R$, were observed, then by virtue of the law of large numbers, we could rely on

$$\begin{aligned}\lim(1/R)\Sigma_r\, h(\mathbf{x}_{it}'\boldsymbol{\beta} + u_{ir}) &= E_u E[y_{it}|\mathbf{x}_{it},u_i] \\ &= \int_u h(\mathbf{x}_{it}'\boldsymbol{\beta} + u_i)f(u_i)\mathrm{d}u_i. \qquad \textbf{(15-1)}\end{aligned}$$

We are suppressing the extra parameter, σ_u, which would become part of the estimation problem. A convenient way to formulate the problem is to write $u_i = \sigma_u v_i$ where v_i has zero mean and variance one. By using this device, integrals can be replaced with sums that are feasible to compute. Our "simulated sum of squares" becomes

$$S_{\text{simulated}}(\boldsymbol{\beta}) = \Sigma_i\Sigma_t[y_{it} - (1/R)\Sigma_r h(\mathbf{x}_{it}'\boldsymbol{\beta} + \sigma_u v_{ir})]^2, \qquad \textbf{(15-2)}$$

which can be minimized by conventional methods. As long as (15-1) holds, then

$$\frac{1}{nT}\Sigma_i\Sigma_t[y_{it} - (1/R)\Sigma_r h(\mathbf{x}_{it}'\boldsymbol{\beta} + \sigma_u v_{ir})]^2 \rightarrow \frac{1}{nT}\Sigma_i\Sigma_t[y_{it} - \int_v h(\mathbf{x}_{it}'\boldsymbol{\beta} + \sigma_u v_i)\, f(v_i)\mathrm{d}v_i]^2 \qquad (15\text{-}3)$$

and it follows that with sufficiently increasing R, the $\boldsymbol{\beta}$ that minimizes the left hand side converges (in nT) to the same parameter vector that minimizes the probability limit of the right hand side. We are thus able to substitute a computer simulation for the intractible computation on the right hand side of the expression.

This chapter will describe some of the (increasingly) more common applications of simulation methods in econometrics. We begin in Section 15.2 with the essential tool at the heart of all the computations, random number generation. Section 15.3 describes simulation based

inference using the method of Krinsky and Robb as an alternative to the delta method (see Section 4.4.4). The method of bootstrapping for inferring the features of the distribution of an estimator is described in Section 15.4. In Section 15.5, we will use a Monte Carlo study to learn about the behavior of a test statistic and the behavior of the fixed effects estimator in some nonlinear models. Sections 15.6 to 15.9 presents simulation based estimation methods. The essential ingredient of this entire set of results is the computation of integrals. Section 15.6.1 describes an application of a simulation based estimator, a nonlinear random effects model. Section 15.6.2 discusses methods of integration. Then, the methods are applied to the estimation of the random effects model. Sections 15.7 – 15.9 describe several techniques and applications, including maximum simulated likelihood estimation for random parameter and hierarchical models. A third major (perhaps *the* major) application of simulation-based estimation in the current literature is Bayesian analysis using Markov Chain Monte Carlo (MCMC or MC^2) methods. Bayesian methods are discussed separately in Chapter 16. Sections 15.10 and 15.11 consider two remaining aspects of modeling parameter heterogeneity, estimation of individual specific parameters and a comparison of modeling with continuous distributions to modeling with discrete distributions using latent class models.

15.2 RANDOM NUMBER GENERATION

All of the techniques we will consider here rely on samples of observations from an underlying population. We will sometimes call these "random samples," though it will emerge shortly that they are never actually random. One of the important aspects of this entire body of research is the need to be able to replicate one's computations. If the samples of draws used in any kind of simulation based analysis were truly random, then this would be impossible. Although the methods we consider here will appear to be random, they are, in fact, deterministic – the "samples" can be replicated. For this reason, the sampling methods described in this section are more often labeled "pseudo-random number generators." (This does raise an intriguing question. Is it possible to generate truly random draws from a population with a computer? The answer for practical purposes is no.) This section will begin with a description of some of the mechanical aspects of random number generation. We will then detail the methods of generating particular kinds of random samples. [See Train (2009, Chapter 3) for extensive further discussion.]

15.2.1 GENERATING PSEUDO-RANDOM NUMBERS

Data are generated internally in a computer using **pseudo–random number generators.** These computer programs generate sequences of values that appear to be strings of draws from a specified probability distribution. There are many types of random number generators, but most take advantage of the inherent inaccuracy of the digital representation of real numbers. The method of generation is usually by the following steps:

1. Set a **seed.**
2. Update the seed by $\text{seed}_j = \text{seed}_{j-1} \times s$ value.
3. $x_j = \text{seed}_j \times x$ value.
4. Transform x_j if necessary, then move x_j to desired place in memory.
5. Return to step 2, or exit if no additional values are needed.

Random number generators produce sequences of values that resemble strings of random draws from the specified distribution. In fact, the sequence of values produced by the preceding method is not truly random at all; it is a deterministic **Markov chain** of values. The set of 32 bits in the random value only appear random when subjected to certain tests. [See Press et al. (1986).] Because the series is, in fact, deterministic, at any point that this type of generator produces a value it has produced before, it must thereafter replicate the entire sequence. Because modern digital computers typically use 32-bit double words to represent numbers, it follows that the longest string of values that this kind of generator can produce is $2^{32} - 1$ (about 4.3 billion). This length is the **period** of a random number generator. (A generator with a shorter period than this would be inefficient, because it is possible to achieve this period with some fairly simple algorithms.) Some improvements in the periodicity of a generator can be achieved by the method of **shuffling**. By this method, a set of, say, 128 values is maintained in an array. The random draw is used to select one of these 128 positions from which the draw is taken and then the value in the array is replaced with a draw from the generator. The period of the generator can also be increased by combining several generators. [See L'Ecuyer (1998), Gentle (2002, 2003), and Greene (2007b).]

The deterministic nature of pseudo–random number generators is both a flaw and a virtue. Many Monte Carlo studies require billions of draws, so the finite period of any generator represents a nontrivial consideration. On the other hand, being able to reproduce a sequence of values just by resetting the seed to its initial value allows the researcher to replicate a study.[1] The seed itself can be a problem. It is known that certain seeds in particular generators will produce shorter series or series that do not pass randomness tests. For example, *congruential* generators of the sort just discussed should be started from odd seeds.

[1] Readers of empirical studies are often interested in replicating the computations. In Monte Carlo studies, at least in principle, data can be replicated efficiently merely by providing the random number generator and the seed.

15.2.2 SAMPLING FROM A STANDARD UNIFORM POPULATION

The output of the generator described in Section 15.2.1 will be a pseudo-draw from the $U[0,1]$ population. (In principle, the draw should be from the closed interval [0,1]. However, the actual draw produced by the generator will be strictly between zero and one with probability just slightly below one. In the application described, the draw will be constructed from the sequence of 32 bits in a double word. All but two of the 2^{31} -1 strings of bits will produce a value in (0,1). The practical result is consistent with the theoretical one, that the probabilities attached to the terminal points are zero also.) When sampling from a standard uniform, $U[0, 1]$ population, the sequence is a kind of difference equation, because given the initial seed, x_j is ultimately a function of x_{j-1}. In most cases, the result at step 3 is a pseudo-draw from the continuous uniform distribution in the range zero to one, which can then be transformed to a draw from another distribution by using the fundamental probability transformation.

15.2.3 SAMPLING FROM CONTINUOUS DISTRIBUTIONS

One is usually interested in obtaining a sequence of draws, $x_1,\ldots,x_R$, from some particular population such as the normal with mean μ and variance σ^2. A sequence of draws from $U[0,1]$, $u_1,\ldots,u_R$, produced by the random number generator is an intermediate step. These will be transformed into draws from the desired population. A common approach is to use the **fundamental probability transformation.** For continuous distributions, this is done by treating the draw, $u_r = F_r$ as if F_r were $F(x_r)$, where $F(.)$ is the cdf of x. For example, if we desire draws from the exponential distribution with known θ, then $F(x) = 1 - \exp(-\theta x)$. The inverse transform is $x = (-1/\theta)\ln(1 - F)$. For example, for a draw of $u = 0.4$ with $\theta = 5$, the associated x would be $(-1/5)\ln(1-.4) = 0.1022$. For the logistic population with cdf $F(x) = \Lambda(x) = \exp(x)/[1+\exp(x)]$, the inverse transformation is $x = \ln[F/(1-F)]$. There are many references, for example, Evans, Hastings and Peacock (2000) and Gentle (2003) that contain tables of inverse transformations that can be used to construct random number generators.

One of the most common applications is the draws from the standard normal distribution. This is complicated because there is no closed form for $\Phi^{-1}(F)$. There are several ways to proceed. A well-known approximation to the inverse function is given in Abramovitz and Stegun (1971):

$$\Phi^{-1}(F) = x \approx T - \frac{c_0 + c_1 T + c_2 T^2}{1 + d_1 T + d_2 T^2 + d_3 T^3},$$

where $T = [\ln(1/H^2)]^{1/2}$ and $H = F$ if $F > 0.5$ and $1 - F$ otherwise. The sign is then reversed if $F < 0.5$. A second method is to transform the $U[0, 1]$ values directly to a standard normal value. The Box–Muller (1958) method is $z = (-2\ln u_1)^{1/2}\cos(2\pi u_2)$, where u_1 and u_2 are two independent $U[0, 1]$ draws. A second $N[0, 1]$ draw can be obtained from the same two values by replacing cos with sin in the transformation. The Marsaglia–Bray (1964) generator is, $z_i = x_i[-(2/v)\ln v]^{1/2}$, where $x_i = 2u_i - 1$, u_i is a random draw from $U[0, 1]$ and $v = u_1^2 + u_2^2$, $i = 1, 2$. The pair of draws is rejected and redrawn if $v \geq 1$.

Sequences of draws from the standard normal distribution can be transformed easily into draws from other distributions by making use of the results in Section B.4. For example, the square of a standard normal draw will be a draw from chi-squared[1], and the sum of K chi-squared[1]s is chi-squared [K]. From this relationship, it is possible to produce samples from the chi-squared[K], t[n], and $F[K,n]$ distributions.

A related problem is obtaining draws from the truncated normal distribution. The random variable with truncated normal distribution is obtained from one with a normal distribution by discarding the part of the range above a value U and below a value L. The density of the resulting random variable is that of a normal distribution restricted to the range $[L,U]$. The truncated normal density is

$$f(x \mid L \le x \le U) = \frac{f(x)}{\text{Prob}[L \le x \le U]} = \frac{(1/\sigma)\phi[(x-\mu)/\sigma]}{\Phi[(U-\mu)/\sigma] - \Phi[(L-\mu)/\sigma]},$$

where $\phi(t) = (2\pi)^{-1/2}\exp(-t^2/2)$ and $\Phi(t)$ is the cdf. An obviously inefficient (albeit effective) method of drawing values from the truncated normal $[\mu,\sigma^2]$ distribution in the range $[L,U]$ is simply to draw F from the $U[0, 1]$ distribution and transform it first to a standard normal variate as discussed previously and then to the $N[\mu,\sigma^2]$ variate by using $x = \mu + \sigma\Phi^{-1}(F)$. Finally, the value x is retained if it falls in the range $[L,U]$ and discarded otherwise. This rejection method will require, on average, $1/\{\Phi[(U-\mu)/\sigma] - \Phi[(L-\mu)/\sigma]\}$ draws per observation, which could be substantial. A direct transformation that requires only one draw is as follows: Let $P_j = \Phi[(j-\mu)/\sigma]$, $j = L, U$. Then

$$x = \mu + \sigma\Phi^{-1}[P_L + F \times (P_U - P_L)]. \qquad \textbf{(15-4)}$$

15.2.4 SAMPLING FROM A MULTIVARIATE NORMAL POPULATION

A common application involves draws from a multivariate normal distribution with specified mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. To sample from this K-variate distribution, we begin with a draw, $\mathbf{z}$, from the K-variate standard normal distribution. This is done by first computing K independent standard normal draws, $z_1,\ldots,z_K$ using the method of the previous section and stacking them in the vector $\mathbf{z}$. Let $\mathbf{C}$ be a square root of $\boldsymbol{\Sigma}$ such that $\mathbf{CC}' = \boldsymbol{\Sigma}$. The desired draw is then $\mathbf{x} = \boldsymbol{\mu} + \mathbf{Cz}$, which will have covariance matrix $E[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})'] = \mathbf{C}E[\mathbf{zz}']\mathbf{C}' = \mathbf{CIC}' = \boldsymbol{\Sigma}$. For the square root matrix, the usual device is the **Cholesky decomposition**, in which $\mathbf{C}$ is a lower triangular matrix. (See Section A.6.11.) For example, suppose we wish to sample from the bivariate normal distribution with mean vector $\boldsymbol{\mu}$, unit variances and correlation coefficient ρ. Then,

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \text{ and } \mathbf{C} = \begin{bmatrix} 1 & 0 \\ \rho & \sqrt{1-\rho^2} \end{bmatrix}.$$

The transformation of two draws z_1 and z_2 is $x_1 = \mu_1 + z_1$ and $x_2 = \mu_2 + [\rho z_1 + (1-\rho^2)^{1/2} z_2]$. Section 15.3 and Example 15.4 following shows a more involved application.

AU: Term "Cholesky decomposition" is not in Chap. list

Au: Term "Poisson" is not in chap list

15.2.5 SAMPLING FROM DISCRETE POPULATIONS

There is generally no inverse transformation available for discrete distributions such as the **Poisson**. An inefficient, though usually unavoidable method for some distributions is to draw the F and then search sequentially for the smallest value that has cdf equal to or greater than F. For example, a generator for the Poisson distribution is constructed as follows. The pdf is Prob[x=j] = p_j = exp(-μ)$\mu^j/j!$ where μ is the mean of the random variable. The generator will use the recursion $p_j = p_{j-1} \times \mu/j$, j = 1,... beginning with p_0 = exp(-μ). An algorithm that requires only a single random draw is as follows:

Initialize		c = exp(-μ); $p = c$; $x = 0$;
Draw		F from U[0,1];
Deliver x	*	exit with draw x if $c > F$;
Iterate		$x = x+1$; $p = p \times \mu/x$; $c = c+p$; go to *.

Au: asterisk "*" part of algorithm?

This method is based explicitly on the pdf and cdf of the distribution. Other methods are suggested by Knuth (1969) and Press et al. (1986, pp. 203–209).

The most common application of random sampling from a discrete distribution is, fortunately, also the simplest. The method of bootstrapping, and countless other applications involve random samples of draws from the **discrete uniform distribution**, Prob($x = j$) = $1/n$, j = 1,...,n. In the bootstrapping application, we are going to draw random samples of observations from the sequence of integers 1,...,n, where each value must be equally likely. In principle, the random draw could be obtained by partitioning the unit interval into n equal parts, $[0,a_1),[a_1,a_2),...,[a_{n-2},a_{n-1}),[a_{n-1},1]$; $a_j = j/n$, j = 1,...,n-1. Then, random draw F delivers $x = j$ if F falls into interval j. This would entail a search, which could be time consuming. However, a simple method that will be much faster is simply to deliver x = the integer part of ($n \times F$ + 1.0). (Once again, we are making use of the practical result that F will equal exactly 1.0 (and x will equal n+1) with ignorable probability.)

15.3 SIMULATION BASED STATISTICAL INFERENCE: THE METHOD OF KRINSKY AND ROBB

Most of the theoretical development in this text has concerned the statistical properties of estimators, that is, the characteristics of sampling distributions such as the mean (probability limits), variance (asymptotic variance), and quantiles (such as the boundaries for confidence intervals). In cases in which these properties cannot be derived explicitly, it is often possible to infer them by using random sampling methods to draw samples from the population that produced an estimator, and deduce the characteristics from the features of such a random sample. In Example 4.4, we computed a set of least squares regression coefficients, $b_1,...,b_K$, then examined the behavior of a nonlinear function $c_k = b_k/(1-b_m)$ using the **delta method**. In some cases, the asymptotic properties of nonlinear functions such as these are difficult to derive directly from the theoretical distribution of the parameters. The sampling methods described here can be used for that purpose. A second common application is learning about the behavior of test statistics. For example, at the end of Section 5.6 and in Section 14.9.1 [see (14-47)], we defined a Lagrange multiplier statistic for testing the hypothesis that certain coefficients are zero in a linear regression model. Under the assumption that the disturbances are normally distributed, the statistic has a limiting chi-squared distribution, which implies that the analyst knows what critical value to employ if they use this statistic. Whether the statistic has this distribution if the disturbances are not normally distributed is unknown. Monte Carlo methods can be helpful in determining if the guidance of the chi squared result is useful in more general cases. Finally, in Section 14.7, we defined a two step maximum likelihood estimator. Computation of the asymptotic variance of such an estimator can be challenging. Monte Carlo methods, in particular, bootstrapping methods, can be used as an effective substitute for the intractible derivation of the appropriate asymptotic distribution of an estimator. This and the next two sections will detail these three procedures, and develop applications to illustrate their use.

The method of Krinsky and Robb is suggested as a way to estimate the asymptotic covariance matrix of $\mathbf{c} = \mathbf{f}(\mathbf{b})$, where $\mathbf{b}$ is an estimated parameter vector with asymptotic covariance matrix $\boldsymbol{\Sigma}$ and $\mathbf{f}(\mathbf{b})$ defines a set of possibly nonlinear functions of $\mathbf{b}$. We assume that $\mathbf{f}(\mathbf{b})$ is a set of continuous and continuously differentiable functions that do not involve the sample size and whose derivatives do not equal zero at $\boldsymbol{\beta} = \text{plim } \mathbf{b}$. (These are the conditions underlying the Slutsky theorem in Section D.2.3.) In Section 4.4.4, we used the delta method to estimate the asymptotic covariance matrix of $\mathbf{c}$; $\text{Est.Asy.Var}[\mathbf{c}] = \mathbf{GSG}'$, where $\mathbf{S}$ is the estimate of $\boldsymbol{\Sigma}$ and $\mathbf{G}$ is the matrix of partial derivatives, $\mathbf{G} = \partial \mathbf{f}(\mathbf{b})/\partial \mathbf{b}'$. The recent literature contains some occasional skepticism about the accuracy of the delta method. The method of Krinsky and Robb (1986, 1990, 1991) is often suggested as an alternative. In a study of the behavior of estimated elasticities based on a translog model, the authors (1986) advocated an alternative approach based on Monte Carlo methods and the law of large numbers. We have consistently estimated $\boldsymbol{\beta}$ and $(\sigma^2/n)\mathbf{Q}^{-1}$, the mean and variance of the asymptotic normal distribution of the estimator $\mathbf{b}$, with $\mathbf{b}$ and $s^2(\mathbf{X}'\mathbf{X})^{-1}$. It follows that we could estimate the mean and variance of the distribution of a function of $\mathbf{b}$ by drawing a random sample of observations from the asymptotic normal population generating $\mathbf{b}$, and using the empirical mean and variance of the sample of functions to estimate the parameters of the distribution of the function. The quantiles of the sample of draws, for example, the .025th and .975th quantiles, can be used to estimate the boundaries of a confidence interval of the functions. The multivariate normal sample would be drawn using the method described in Section 15.2.4.

Krinsky and Robb (1986) reported huge differences in the standard errors produced by the delta method compared to the simulation based estimator. In a subsequent paper (1990), they reported that the entire difference could be attributed to a bug in the software they used. Upon redoing the computations, their estimates were essentially the same with the two methods. It is difficult to draw a conclusion about the effectiveness of the delta method based on the received results—it does seem at this juncture that the delta method remains an effective device that can often be employed with a hand calculator as opposed to the much more computation intensive Krinsky and Robb (1986) technique. Unfortunately, the results of any comparison will depend on the data, the model, and the functions being computed. The amount of nonlinearity in the sense of the complexity of the functions seems not to be the answer. Krinsky and Robb's case was motivated by the extreme complexity of the elasticities in a translog model. In another study, Hole (2006) examines a similarly complex problem, and finds that the delta method still appears to be the more accurate procedure.

Example 15.4 Long Run Elasticities

A dynamic version of the demand for gasoline model is estimated in Example 4.4. The model is

$$\ln(G/Pop)_t = \beta_1 + \beta_2 \ln P_{G,t} + \beta_3 \ln(Income/Pop)_t + \beta_4 \ln P_{nc,t} + \beta_5 \ln P_{uc,t} + \gamma \ln(G/Pop)_{t-1} + \varepsilon_t .$$

In this model, the short run price and income elasticities are β_2 and β_3. The long run elasticities are $\phi_2 = \beta_2/(1 - \gamma)$ and $\phi_3 = \beta_3/(1 - \gamma)$, respectively. To estimate the long run elasticities, we estimated the parameters by least squares and then computed these two nonlinear functions of the estimates. Estimates of the full set of model parameters and the estimated asymptotic covariance matrix are given in Example 4.4. The delta method was used to estimate the asymptotic standard errors for the estimates of ϕ_2 and ϕ_3. The three estimates of the specific parameters and the 3×3 submatrix of the estimated asymptotic covariance matrix are

$$Est.\begin{pmatrix}\beta_2\\ \beta_3\\ \gamma\end{pmatrix} = \begin{pmatrix}b_2\\ b_3\\ c\end{pmatrix} = \begin{pmatrix}-0.069532\\ 0.164047\\ 0.830971\end{pmatrix},$$

$$Est.\ Asy.\ Var\begin{pmatrix}b_2\\ b_3\\ c\end{pmatrix} = \begin{pmatrix}0.00021705 & 1.61265e{-5} & -0.0001109\\ 1.61265e{-5} & 0.0030279 & -0.0021881\\ -0.0001109 & -0.0021881 & 0.0020943\end{pmatrix}.$$

AU: Check these Eqs - OK?

The method suggested by Krinsky and Robb would use a random number generator to draw a large trivariate sample, $(b_2, b_3, c)_r$, $r = 1, \ldots, R$, from the normal distribution with this mean vector and covariance matrix, then compute the sample of observations on f_2 and f_3 and obtain the empirical mean and variance and the .025 and .975 quantiles from the sample. The method of drawing such a sample is shown in Section 15.2.4. We will require the square root of the covariance matrix. The Cholesky matrix is

$$\mathbf{C} = \begin{pmatrix}0.0147326 & 0 & 0\\ 0.00109461 & 0.0550155 & 0\\ -0.0075275 & -0.0396227 & 0.0216259\end{pmatrix}$$

The sample is drawn by drawn by obtaining vectors of three random draws from the standard normal population, $\mathbf{v}_r = (v_1, v_2, v_3)_r'$, $r = 1, \ldots, R$. The draws needed for the estimation are then obtained by computing $\mathbf{b}_r = \mathbf{b} + \mathbf{C}\mathbf{v}_r$ where $\mathbf{b}$ is the set of least squares estimates. We then compute the sample of estimated long-run elasticities, $f_{2r} = b_{2r}/(1 - c_r)$ and $f_{3r} = b_{3r}/(1 - c_r)$. The mean and variance of the sample observations constitute the estimates of the functions and asymptotic standard errors.

Table 15.1 shows the results of these computations based on 1,000 draws from the underlying distribution. The estimates from Example 4.4 using the delta method are shown as well. The two sets of estimates are in quite reasonable agreement. A 95% confidence interval for ϕ_2 based on the estimates, the t distribution with 51-6 = 45 degrees of freedom and the delta method would be -0.411358 ± 2.014103(0.152296). The result for ϕ_3 would be 0.970522 ± 2,014103(0.162386). These are shown in Table 15.2 with the same computation using the Krinsky and Robb estimated standard errors. The table also shows the empirical estimates of these quantiles computed using the 26th and 975th values in the samples. There is reasonable agreement in the estimates, though there is also evident a considerable amount of sample variability, even in a sample as large as 1,000.

We note, finally, that it is generally not possible to replicate results such as these across software platforms, because they use different random number generators. Within a given platform, replicability can be obtained by setting the seed for the random number generator.

TABLE 15.1 Simulation Results

	Regression Estimate		*Simulated Values*	
	Estimate	*Std.Error*	*Mean*	*Std.Dev.*
β_2	-0.069532	0.0147327	-0.068791	0.0138485
β_3	0.164047	0.0550265	0.162634	0.0558856
γ	0.830971	0.0457635	0.831083	0.0460514
ϕ_2	-0.411358	0.152296	-0.453815	0.219110
ϕ_3	0.970522	0.162386	0.950042	0.199458

TABLE 15.2 Estimated Confidence Intervals

	ϕ_2		ϕ_3	
	Lower	Upper	Lower	Upper
Delta Method	-0.718098	-0.104618	0.643460	1.297585
Krinsky and Robb	-0.895125	-0.012505	0.548313	1.351772
Sample Quantiles	-0.983866	-0.209776	0.539668	1.321617

15.4 BOOTSTRAPPING STANDARD ERRORS AND CONFIDENCE INTERVALS

The technique of **bootstrapping** is used to obtain a description of the sampling properties of empirical estimators using the sample data themselves, rather than broad theoretical results.[2] Suppose that $\hat{\theta}_n$ is an estimator of a parameter vector θ based on a sample $\mathbf{Z} = [(y_1,\mathbf{x}_1),\ldots,(y_n,\mathbf{x}_n)]$. An approximation to the statistical properties of $\hat{\theta}_n$ can be obtained by studying a sample of bootstrap estimators $\hat{\theta}(b)_m$, m, $b = 1, \ldots, B$, obtained by sampling m observations, *with replacement*, from $\mathbf{Z}$ and recomputing $\hat{\theta}$ with each sample. After a total of B times, the desired sampling characteristic is computed from

$$\hat{\Theta} = \left[\hat{\theta}(1)_m, \hat{\theta}(2)_m, \ldots \hat{\theta}(B)_m\right].$$

The most common application of bootstrapping for consistent estimators when n is reasonably large is approximating the asymptotic covariance matrix of the estimator $\hat{\theta}_n$ with

$$Est.Asy.Var\left[\hat{\theta}_n\right] = \frac{1}{B-1}\sum_{b=1}^{B}\left[\hat{\theta}(b)_m - \bar{\hat{\theta}}_B\right]\left[\hat{\theta}(b)_m - \bar{\hat{\theta}}_B\right]', \quad (15\text{-}5)$$

where $\bar{\hat{\theta}}_B$ is the average of the B bootstrapped estimates of θ. There are few theoretical prescriptions for the number of replications, B. Andrews and Buchinsky (2000) and Cameron and Trivedi (2005, pp. 361-2), make some suggestions for particular applications; Davidson and MacKinnon (2000) recommend at least 399. Several hundred is the norm; we have used 1,000 in our application to follow. This technique was developed by Efron (1979) and has been appearing with increasing frequency in the applied econometrics literature. [See, for example, Veall (1987, 1992), Vinod (1993), and Vinod and Raj (1994). Extensive surveys of uses and methods in econometrics appear in Cameron and Trivedi (2005), Horowitz (2001), and Davidson and MacKinnon(2006).] An application of this technique to the least absolute deviations estimator in the linear model is shown in the following example and in Chapter 4.

The preceding is known as a "**paired bootstrap**." The pairing is the joint sampling of y_i and $\mathbf{x}_i$. An alternative approach in a regression context would be to sample the observations on $\mathbf{x}_i$ only, then with each $\mathbf{x}_i$ sampled, generate the accompanying y_i by randomly generating the disturbance, then $\hat{y}_i(b) = \mathbf{x}_i(b)'\hat{\theta}_n + \hat{\varepsilon}_i(b)$. This would be a "**parametric bootstrap**" in that in order to simulate the disturbances, we need either to know (or assume) the data generating process that produces ε_i. In other contexts, such as in discrete choice modeling in Chapter 17, one would bootstrap sample the exogenous data in the model, then generate the dependent variable by this method using the appropriate underlying DGP. This is the approach used in 15.5.5 and in Greene (2004b) in a study of the incidental parameters problem in several limited dependent variable models. The obvious disadvantage of the parametric bootstrap is that one cannot learn of the influence of an unknown DGP for ε by assuming it is known. For example, if the bootstrap is being used to accommodate unknown heteroscedasticity in the model, a parametric bootstrap that assumes homoscedasticity would defeat the purpose. The more natural application would be a **nonparametric bootstrap**, in which both $\mathbf{x}_i$ and y_i, and, implicitly, ε_i, are sampled simultaneously.

[2] See Efron (1979), Efron and Tibshirani (1994), and Davidson and Hinkley (1997), Brownstone and Kazimi (1998), Horowitz (2001) and MacKinnon (2002).

Example 15.5 Bootstrapping the Variance of the Median

There are few cases in which an exact expression for the sampling variance of the median are known. Example 15.7 following, examines the case of the median of a sample of 500 observations from the t distribution with 10 degrees of freedom. This is one of those cases in which there is no exact formula for the asymptotic variance of the median. However, we can use the bootstrap technique to estimate one empirically. In one run of the experiment, we obtained a sample of 500 observations for which we computed the median, −0.00786. We drew 100 samples of 500 with replacement from this sample of 500 and recomputed the median with each of these samples. The empirical square root of the mean squared deviation around this estimate of −0.00786 was 0.056. In contrast, consider the same calculation for the mean. The sample mean is −0.07247. The sample standard deviation is 1.08469, so the standard error of the mean is 0.04657. (The bootstrap estimate of the standard error of the mean was 0.052.) This agrees with our expectation in that the sample mean should generally be a more efficient estimator of the mean of the distribution in a large sample. There is another approach we might take in this situation. Consider the regression model

$$y_i = \alpha + \varepsilon_i$$

where ε_i has a symmetric distribution with finite variance. The least absolute deviations estimator of the coefficient in this model is an estimator of the median (which equals the mean) of the distribution. So, this presents another estimator. Once again, the bootstrap estimator must be used to estimate the asymptotic variance of the estimator. Using the same data, we fit this regression model using the LAD estimator. The coefficient estimate is −.05397 with a bootstrap estimated standard error of 0.05872. The estimated standard error agrees with the earlier one. The difference in the estimated coefficient stems from the different computations—the regression estimate is the solution to a linear programming problem while the earlier estimate is the actual sample median.

The bootstrap estimation procedure has also been suggested as a method of reducing bias. In principle, we would compute $\hat{\theta}_n - \text{bias}(\hat{\theta}_n) = \hat{\theta}_n - \{E[\hat{\theta}_n] - \theta\}$. Since neither θ nor the exact expectation of $\hat{\theta}_n$ are known, we estimate the first with the mean of the bootstrap replications and the second with the estimator, itself. The revised estimator is

$$\hat{\theta}_{n,B} = \hat{\theta}_n - \left[\frac{1}{B}\sum_{b=1}^{B}\hat{\theta}(m) - \hat{\theta}_n\right] = 2\hat{\theta}_n - \bar{\hat{\theta}}_B. \qquad (15\text{-}6)$$

(Efron and Tibshirani (1994, p. 138) provide justification for what appears to be the wrong sign on the correction.) Davidson and MacKinnon (2006) argue that the smaller bias of the corrected estimator is offset by an increased variance compared to the uncorrected estimator. [See, as well, Cameron and Trivedi (2005).] The authors offer some other cautions for practitioners contemplating use of this technique. First, perhaps obviously, the extension of the method to samples with dependent observations presents some obstacles. For time series data, the technique makes little sense—none of the bootstrapped samples will be a time series, so the properties of the resulting estimators will not satisfy the underlying the assumptions needed to make the technique appropriate.

A second common application of bootstrapping methods is the computation of confidence intervals for parameters. This calculation will be useful when the underlying data generating process is unknown, and the bootstrap method is being used to obtain appropriate standard errors for estimated parameters. A natural approach to bootstrapping confidence intervals for parameters would be to compute the estimated asymptotic covariance matrix using (15-5), then form confidence intervals in the usual fashion. An improvement in terms of the bias of the estimator is provided by the **percentile method** [Cameron and Trivedi (2005, p. 364)]. By this technique, during each bootstrap replication, we compute

$$t_k^*(b) = \frac{\hat{\theta}_k(b) - \hat{\theta}_{n,k}}{s.e.\left(\hat{\theta}_{n,k}\right)}, \qquad (15\text{-}7)$$

where "k" indicates the kth parameter in the model, and $\hat{\theta}_{n,k}$, $s.e.\left(\hat{\theta}_{n,k}\right)$ and $\hat{\theta}_k(b)$ are the original estimator and estimated standard error from the full sample and the bootstrap replicate. Then, with all B replicates in hand, the bootstrap confidence interval is

$$\hat{\theta}_{n,k} + t_k^*[\alpha/2]\ s.e.\left(\hat{\theta}_{n,k}\right) \text{ to } \hat{\theta}_{n,k} + t_k^*[1\text{-}\alpha/2]\ s.e.\left(\hat{\theta}_{n,k}\right). \qquad (15\text{-}8)$$

(Note that $t_k^*[\alpha/2]$ is negative, which explains the plus sign in left term.) For example, in our application below, we compute the estimator and the asymptotic covariance matrix using the full sample. We compute 1,000 bootstrap replications, and compute the t-ratio in (15-7) for the education coefficient in each of the 1,000 replicates. After the bootstrap samples are accumulated, we sorted the results from (15-7), and the 25th and 975th largest values provide the values of t^*.

Example 15.6 demonstrates the computation of a confidence interval for a coefficient using the bootstrap. The application uses the Cornwell and Rupert panel data set used in Example 11.1 and several later applications. There are 595 groups of 7 observations in the data set. Bootstrapping with panel data requires an additional element in the computations. The bootstrap replications are based on sampling over i, not t. Thus, the bootstrap sample consists of n blocks of T (or T_i) observations the ith group as a whole is sampled. This produces, then, a **block bootstrap** sample.

Example 15.6 Bootstrapping Standard Errors and Confidence Intervals in a Panel

Example 11.1 presents least squares estimates and robust standard errors for the labor supply equation using Cornwell and Rupert's panel data set. There are 595 individuals and 7 periods in the data set. As seen in the results in Table 11.1 (reproduced below), using a clustering correction in a robust covariance matrix for the least squares estimator produces substantial changes in the estimated standard errors. Table 15.3 presents the least squares coefficients and the standard errors estimated with the conventional $s^2(\mathbf{X}'\mathbf{X})^{-1}$, the robust standard errors using the clustering correction, and the bootstrapped standard errors using 1,000 bootstrap replications. The resemblance between the original estimates in the leftmost column and the average of the bootstrap replications in the rightmost column is to be expected; the sample is quite large and the number of replications is large. What is striking (and reassuring) is the ability of the bootstrapping procedure to detect and mimic the effect of the clustering that is evident in the second and third column of estimated standard errors.

We also computed a confidence interval for the coefficient on *Ed* using the conventional, symmetric approach, $b_{Ed} \pm 1.96s(b_{Ed})$, and the percentile method in (15-7)-(15-8). The two intervals are

Conventional: 0.051583 to 0.061825
Percentile: 0.045560 to 0.067909

Not surprisingly (given the larger standard errors), the percentile method gives a much wider interval. Figure 15.1 shows a kernel density estimator of the distribution of the *t* statistics computed using (15-7). It is substantially wider than the (approximate) standard normal density shown with it. This demonstrates the impact of the latent effect of the clustering on the standard errors, and ultimately on the test statistic used to compute the confidence intervals.

TABLE 15.3 Bootstrap Estimates of Standard Errors for a Wage Equation

Variable	*Least Squares Estimate*	*Standard Error*	*Cluster Robust Std. Error*	*Bootstrap Std. Error*	*Bootstrap Coefficient*
Constant	5.25112	0.07129	0.1233	0.12421	5.25907
Wks	0.00422	0.00108	0.001538	0.00159	0.00409
South	-0.05564	0.01253	0.02610	0.02557	-0.05417
SMSA	0.15167	0.01207	0.02405	0.02383	0.15140
MS	0.04845	0.02057	0.04085	0.04208	0.04676
Exp	0.04010	0.00216	0.004067	0.00418	0.04017
Exp^2	-0.00067	0.00004744	0.00009111	0.00009235	-0.00067
Occ	-0.14001	0.01466	0.02718	0.02733	-0.13912
Ind	0.04679	0.01179	0.02361	0.02350	0.04728
Union	0.09263	0.01280	0.02362	0.02390	0.09126
Ed	0.05670	0.00261	0.005552	0.00576	0.05656
Fem	-0.36779	0.02510	0.04547	0.04562	-0.36855
Blk	-0.16694	0.02204	0.04423	0.04663	-0.16811

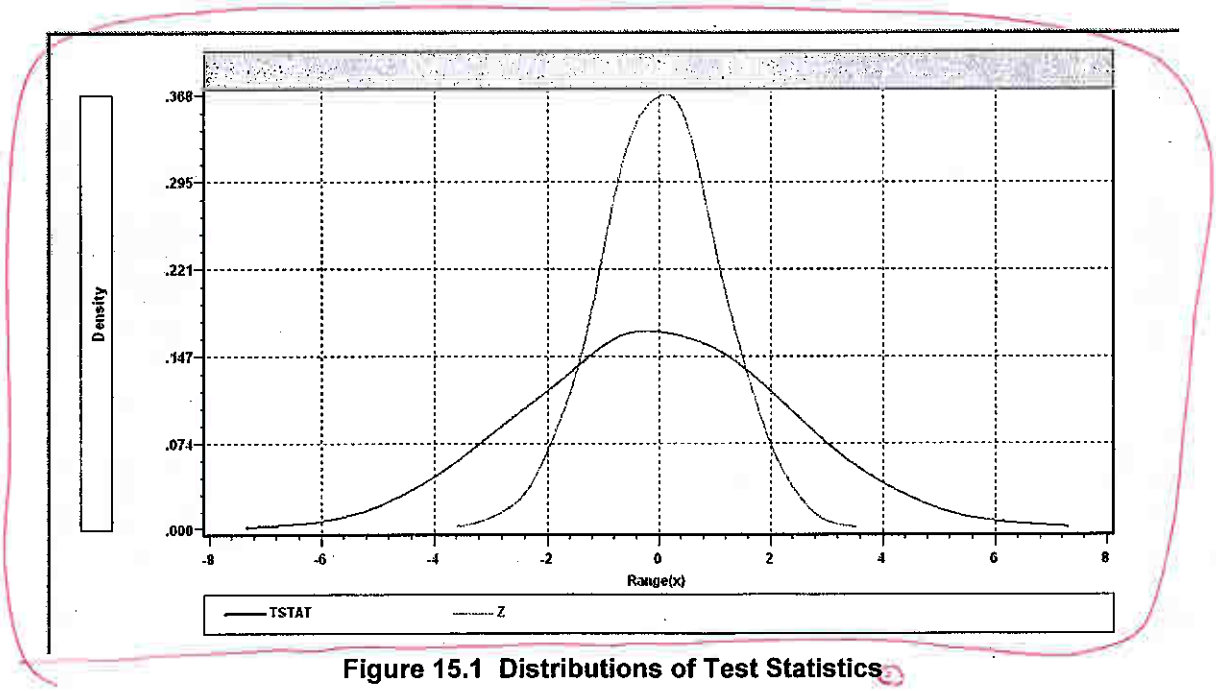


Figure 15.1 Distributions of Test Statistics

15.5 MONTE CARLO STUDIES

Simulated data generated by the methods of the preceding sections have various uses in econometrics. One of the more common applications is the derivation of the properties of estimators or in obtaining comparisons of the properties of estimators. For example, in time-series settings, most of the known results for characterizing the sampling distributions of estimators are asymptotic, large-sample results. But the typical time series is not very long, and descriptions that rely on T, the number of observations, going to infinity may not be very accurate. Exact, finite sample properties are usually intractable, however, which leaves the analyst with only the choice of learning about the behavior of the estimators experimentally.

In the typical application, one would either compare the properties of two or more estimators while holding the sampling conditions fixed or study how the properties of an estimator are affected by changing conditions such as the sample size or the value of an underlying parameter.

15.5.1 Example Monte Carlo Study of the Mean Versus the Median

In Example D.8, we compared the asymptotic distributions of the sample mean and the sample median in random sampling from the normal distribution. The basic result is that both estimators are consistent, but the mean is asymptotically more efficient by a factor of

$$\frac{\text{Asy. Var[Median]}}{\text{Asy. Var[Mean]}} = \frac{\pi}{2} = 1.5708.$$

This result is useful, but it does not tell which is the better estimator in small samples, nor does it suggest how the estimators would behave in some other distribution. It is known that the mean is affected by outlying observations whereas the median is not. The effect is averaged out in large samples, but the small sample behavior might be very different. To investigate the issue, we constructed the following experiment: We sampled 500 observations from the t distribution with d degrees of freedom by sampling $d + 1$ values from the standard normal distribution and then computing

$$t_{ir} = \frac{z_{ir,d+1}}{\sqrt{\frac{1}{d}\sum_{l=1}^{d} z_{ir,l}^2}}, \quad i = 1, \ldots, 500, \quad r = 1, \ldots, 100.$$

The t distribution with a low value of d was chosen because it has very thick tails and because large, outlying values have high probability. For each value of d, we generated $R = 100$ replications. For each of the 100 replications, we obtained the mean and median. Because both are unbiased, we compared the mean squared errors around the true expectations using

$$M_d = \frac{(1/R)\sum_{r=1}^{R}(\text{median}_r - 0)^2}{(1/R)\sum_{r=1}^{R}(\bar{x}_r - 0)^2}.$$

We obtained ratios of 0.6761, 1.2779, and 1.3765 for $d = 3, 6$, and 10, respectively. (You might want to repeat this experiment with different degrees of freedom.) These results agree with what intuition would suggest. As the degrees of freedom parameter increases, which brings the distribution closer to the normal distribution, the sample mean becomes more efficient, the ratio should approach its limiting value of 1.5708 as d increases. What might be surprising is the apparent overwhelming advantage of the median when the distribution is very nonnormal even in a sample as large as 500.

The preceding is a very small, straightforward application of the technique. In a typical study, there are many more parameters to be varied and more dimensions upon which the results are to be studied. One of the practical problems in this setting is how

to organize the results. There is a tendency in Monte Carlo work to proliferate tables indiscriminately. It is incumbent on the analyst to collect the results in a fashion that is useful to the reader. For example, this requires some judgment on how finely one should vary the parameters of interest. One useful possibility that will often mimic the thought process of the reader is to collect the results of bivariate tables in carefully designed contour plots.

There are any number of situations in which Monte Carlo simulation offers the only method of learning about finite sample properties of estimators. Still, there are a number of problems with Monte Carlo studies. To achieve any level of generality, the number of parameters that must be varied and hence the amount of information that must be distilled can become enormous. Second, they are limited by the design of the experiments, so the results they produce are rarely generalizable. For our example, we may have learned something about the t distribution. But the results that would apply in other distributions remain to be described. And, unfortunately, real data will rarely conform to any specific distribution, so no matter how many other distributions we analyze, our results would still only be suggestive. In more general terms, this problem of **specificity** [Hendry (1984)] limits most Monte Carlo studies to quite narrow ranges of applicability. There are very few that have proved general enough to have provided a widely cited result.[3]

17.4.1 A MONTE CARLO STUDY: BEHAVIOR OF A TEST STATISTIC

Monte Carlo methods are often used to study the behavior of test statistics when their true properties are uncertain. This is often the case with Lagrange Multiplier statistics. For example, Baltagi (2005) reports on the development of several new test statistics for panel data models such as a test for serial correlation. Examining the behavior of a test statistic is fairly straightforward. We are interested in two characteristics: the **true size of the test**—that is, the probability that it rejects the null hypothesis when that hypothesis is actually true (the probability of a type 1 error) and the **power of the test**—that is the probability that it will correctly reject a false null hypothesis (one minus the probability of a type 2 error). As we will see, the power of a test is a function of the alternative against which the null is tested.

To illustrate a Monte Carlo study of a test statistic, we consider how a familiar procedure behaves when the model assumptions are incorrect. Consider the linear regression model

$$y_i = \alpha + \beta x_i + \gamma z_i + \varepsilon_i, \quad \varepsilon_i \,|\, (x_i, z_i) \sim \mathrm{N}[0, \sigma^2].$$

The Lagrange multiplier statistic for testing the null hypothesis that γ equals zero for this model is

$$\mathrm{LM} = \mathbf{e}_0'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{e}_0/(\mathbf{e}_0'\mathbf{e}_0/n)$$

where $\mathbf{X} = (\mathbf{1}, \mathbf{x}, \mathbf{z})$ and $\mathbf{e}_0$ is the vector of least squares residuals obtained from the regression of $\mathbf{y}$ on the constant and $\mathbf{x}$ (and not $\mathbf{z}$). (See Section 16.6.3.) Under the assumptions of the model above, the large sample distribution of the LM statistic is chi squared with one degree of freedom. Thus, our testing procedure is to compute LM, then reject the null hypothesis $\gamma = 0$ if LM is greater than the critical value. We

[3] Two that have withstood the test of time are Griliches and Rao (1969) and Kmenta and Gilbert (1968).

TABLE ~~17.1~~ Size and Power Functions for LM Test

Model	Gamma 0.0	0.1 −0.1	0.2 −0.2	0.3 −0.3	0.4 −0.4	0.5 −0.5	0.6 −0.6	0.7 −0.7	0.8 −0.8	0.9 −0.9	1.0 −1.0
Normal	0.059	0.090	0.235	0.464	0.691	0.859	0.957	0.989	0.998	1.000	1.000
		0.103	0.236	0.451	0.686	0.863	0.961	0.989	0.999	1.000	1.000
$t(5)$	0.052	0.083	0.169	0.320	0.508	0.680	0.816	0.911	0.956	0.976	0.994
		0.080	0.177	0.312	0.500	0.677	0.822	0.921	0.953	0.984	0.993
Het.	0.071	0.098	0.249	0.457	0.666	0.835	0.944	0.984	0.995	0.998	1.000
		0.107	0.239	0.442	0.651	0.832	0.940	0.985	0.996	1.000	1.000

will use a nominal size of 0.05, so the critical value is 3.84. The theory for the statistic is well developed when the specification of the model is correct. [See, for example, Godfrey (1988).] We are interested in two specification errors. First, how does the statistic behave if the normality assumption is not met? Because the LM statistic is based on the likelihood function, if some distribution other than the normal governs ε_i, then the LM statistic would not be based on the OLS estimator. We will examine the behavior of the statistic under the true specification that ε_i comes from a t distribution with 5 degrees of freedom. Second, how does the statistic behave if the homoscedasticity assumption is not met? The statistic is entirely wrong if the disturbances are heteroscedastic. We will examine the case in which the conditional variance is $\text{Var}[\varepsilon_i \mid (x_i, z_i)] = \sigma^2[\exp(0.2x_i)]^2$.

The design of the experiment is as follows: We will base the analysis on a sample of 50 observations. We draw 50 observations on x_i and z_i from independent N[0, 1] populations at the outset of each cycle. For each of 1,000 replications, we draw a sample of 50 ε_is according to the assumed specification. The LM statistic is computed and the proportion of the computed statistics that exceed 3.84 is recorded. The experiment is repeated for $\gamma = 0$ to ascertain the true size of the test and for values of γ including $-1, \ldots, -0.2, -0.1, 0, 0.1, 0.2, \ldots, 1.0$ to assess the power of the test. The cycle of tests is repeated for the two scenarios, the $t(5)$ distribution and the model with heteroscedasticity.

Table 17.1 lists the results of the experiment. The first row shows the expected results for the LM statistic under the model assumptions for which is is appropriate. The size of the test appears to be in line with the theoretical results. Comparing the first and third rows, it appears that the presence of heteroscedasticity seems not to degrade the power of the statistic. But the different distributional assumption does. Figure ~~17.1~~ 15.2 plots the values in the table, and displays the characteristic form of the power function for a test statistic.

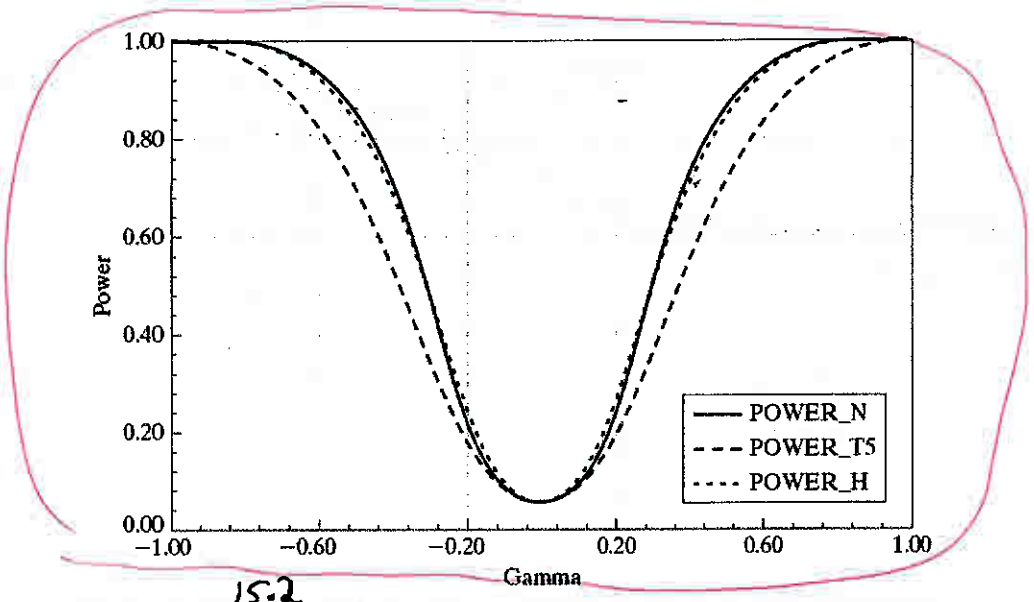
~~17.4.2~~ A MONTE CARLO STUDY: THE INCIDENTAL PARAMETERS PROBLEM

Section ~~16.9.6.c~~ examines the maximum likelihood estimator of a panel data model with fixed effects,

$$f(y_{it} \mid \mathbf{x}_{it}) = g(y_{it}, \mathbf{x}_{it}'\boldsymbol{\beta} + \alpha_i, \theta)$$

where the individual effects may be correlated with $\mathbf{x}_{it}$. The extra parameter vector θ represents M other parameters that might appear in the model, such as the disturbance

15-18



15.2

FIGURE 17.4 Power Functions.

variance, σ_ε^2, in a linear regression model with normally distributed disturbance. The development there considers the mechanical problem of maximizing the log-likelihood

$$\ln L = \sum_{i=1}^{n} \sum_{t=1}^{T_i} \ln g(y_{it}, \mathbf{x}_{it}'\boldsymbol{\beta} + \alpha_i, \theta)$$

with respect to the $n + K + M$ parameters $(\alpha_1, \ldots, \alpha_n, \boldsymbol{\beta}, \theta)$. A statistical problem with this estimator that was suggested there is a phenomenon labeled the **incidental parameters problem** [see Neyman and Scott (1948), Lancaster (2000)]. With the exception of a very small number of specific models (such as the Poisson regression model in Section 25.3.2), the "brute force," unconditional maximum likelihood estimator of the parameters in this model is inconsistent. The result is straightforward to visualize with respect to the individual effects. Suppose that $\boldsymbol{\beta}$ and θ were actually known. Then, each α_i would be estimated with T_i observations. Because T_i is assumed to be fixed (and small), there is no asymptotic result to provide consistency for the MLE of α_i. But, $\boldsymbol{\beta}$ and θ are estimated with $\Sigma_i T_i = N$ observations, so their large sample behavior is less transparent. One known result concerns the logit model for binary choice (see Section 23.2–23.5). Kalbfleisch and Sprott (1970), Andersen (1973), Hsiao (1996), and Abrevaya (1997) have established that in the binary logit model, if $T_i = 2$, then plim $\hat{\boldsymbol{\beta}}_{\text{MLE}} = 2\boldsymbol{\beta}$. Two other cases are known with certainty. In the linear regression model with fixed effects and normally distributed disturbances, the slope estimator, $\mathbf{b}_{\text{LSDV}}$ is unbiased and consistent, however, the MLE of the variance, σ^2 converges to $(T-1)\sigma^2/T$. (The degrees of freedom correction will adjust for this, but the MLE does not correct for degrees of freedom.) Finally, in the Poisson regression model (Section 25.3.2), the unconditional MLE is consistent [see Cameron and Trivedi (1988)]. Almost nothing else is known with certainty—that is, as a firm theoretical result—about the behavior of the maximum