

# Assessing the Reliability of Statistical Software: Part II

B. D. McCULLOUGH

Part I outlined a methodology for assessing the reliability of three areas: estimation, random number generation, and calculation of statistical distributions. The present article applies this methodology to SAS, SPSS, and S-Plus, with attention to implementation details. Weaknesses are identified in all the random number generators, the S-Plus correlation procedure, and in the one-way ANOVA and nonlinear least squares routines of SAS and SPSS.

**KEY WORDS:** Accuracy; Benchmarks; Random number generation; Software testing; StRD.

## 1. INTRODUCTION

Part I (McCullough 1998) proposed a methodology for intermediate-level assessment of the reliability of statistical software on three fronts: (1) estimation, using the NIST StRD (Rogers, et al. 1998) to evaluate the accuracy of univariate summary statistics, one-way ANOVA, linear regression, and nonlinear least squares; (2) random number generation, using Marsaglia's (1996) DIEHARD Battery of Randomness Tests to determine whether random numbers are uncorrelated; and (3) statistical distributions, using either ELV (Knüsel 1989) or DCDFLIB (Brown 1998) to verify the accuracy of statistical distributions (used to calculate  $p$  values or critical values). Here in Part II the methodology is applied to three popular statistical packages: SAS 6.12, SPSS 7.5, and S-Plus 4.0, running on a 133 MHz Pentium under Windows 95. Before this can be done, several important details must be addressed. First among these is how to measure the accuracy of a computed statistic.

As discussed in Part I, the base-10 log relative error (LRE) is used to measure the accuracy of estimated coefficients. If  $x$  is a value computed by a statistical package, and  $c$  is the correct value (NIST refers to these as "certified values"), then the number of correct digits in  $x$  is given by the log relative error (LRE)  $\log_{10}(|x - c|/|c|)$ , denoted by  $\lambda$  with an appropriate subscript. If  $c = 0$ , then the log absolute error is used. Whether the LRE can be accurately calculated depends upon whether the package permits users to access or control displayed results.

Some programs allow the user to access the results of computations; for example, the slope estimates in a linear regression. The user can then read in the certified values and calculate the LRE. S-Plus does this easily, allowing direct access. SAS requires coefficients to be stored and then accessed. Other programs do not allow the user to access such results, but do allow the user to control the number of displayed digits. In this case, after displaying the maximum number of digits, the output can be saved to an ASCII file, edited, and then read in to another program along with the certified values to calculate the LRE. Some SPSS procedures are like this. Yet other programs do not allow the user either to access results or control the number of displayed digits. Other SPSS procedures are like this. This can have either a beneficial or deleterious effect on perceived precision. If the certified value has no more digits than are displayed, the implicit rounding can raise the LRE: for example, the certified value is .004 exactly and the program calculates .00386 but displays only three decimals after rounding up. Alternatively, if the certified value contains more digits than are displayed, this can artificially deflate the LRE: for example, the certified value is .3456789 but the program only displays three places, .346.

Occasionally, the LRE can inform the user of what is not written in the user manual. For example, in Part I it was noted that Package W calculates .786 for the sample standard deviation ( $s$ ) of the dataset Michelson, while the StRD certified value is .790, for an LRE of  $\lambda_s = 2.3$ . It can be deduced that Package W divides by  $n$  instead of  $n - 1$ . [Thanks to J. Doornik for pointing this out.] The documentation gives no formula for  $s$ , but does mention the usual degrees of freedom correction for the standard errors of regression coefficients. Most users might not expect the maximum likelihood estimator of  $s$ . Therefore, the application of the benchmark has revealed useful information about the package.

A procedure can return satisfactory LREs for almost all the test problems in a suite, and yet be judged inadequate. As an example, consider the linear regression suite. In the presence of collinear data, some algorithms are likely to give erroneous results. Therefore, before displaying results, the program should test the data to ensure they are not too collinear—if they are too collinear, the program should not report results but instead return a "near singular data matrix" or other similar error message. So important is this pre-testing that Press, Teukolsky, Vetterling, and Flannery (1992, p. 23) wrote "much of the sophistication of complicated 'linear equation-solving packages' is devoted to the detection" of near-singularity.

For most any linear equation solver, it is possible to reverse-engineer a problem that causes the solver to fail. Such problems do not constitute useful, general-purpose benchmarks. The StRD problems were carefully chosen to

B. D. McCullough is Senior Economist, Federal Communications Commission, 445 12th Street SW, Room 2C134, Washington, DC 20554 (Email: [bmccullo@fcc.gov](mailto:bmccullo@fcc.gov)). Thanks to the software vendors for their support and to their employees for assistance, especially T. Hesterberg (MathSoft), D. Nichols (SPSS, Inc.), and M. Stokes (SAS Institute). Thanks also to an associate editor, the referees, seminar participants at the Washington Statistical Society, and various persons who wish to remain anonymous. Special thanks are due to the editor, whose numerous comments and suggestions greatly improved the exposition. The views expressed herein are those of the author and do not necessarily reflect those of the Commission.

avoid this possibility. For the StRD problems, a reliable solver will either return several accurate digits or an error message: it will not produce a “solution” with zero accurate digits. The StRD linear regression test problem “Filip” is a tenth degree polynomial. This nearly singular problem can severely stress many linear solvers. Suppose a package returns several accurate digits for all linear regression problems but Filip, for which it returns zero accurate digits instead of an error message. Then that package’s linear solver can be considered unreliable on the grounds that it either does not check for near-singularity, or does a very poor job of checking.

There is an analogous situation for nonlinear least squares. It is too much to expect that a particular nonlinear solver can always find a solution, but it is not too much to expect that a solver can recognize when it has not found a solution. Accordingly, a reliable nonlinear solver will return either accurate digits or an error message indicating that a solution cannot be found. Nonlinear estimation is, of course, much more complicated than linear estimation. Therefore, the remainder of this section considers details of nonlinear estimation which are relevant to using the StRD.

The solution to a nonlinear least squares estimation problem obtained from using default options rarely is as good as that obtained from using some other settings. For every package there does seem to be some “preferred combination” of options which consistently yields better results than default options. Simply trying all possible combinations results in an unmanageably large set of possibilities, so some more efficient method is necessary. All 27 of the nonlinear test problems come with two sets of starting values: Start I is “far” from the solution, and Start II is “close” to the solution. Start II is used only if no solution—even one with zero accurate digits—can be obtained from Start I. To keep the number of combinations manageable, all the benchmarks were run in the following order, using the LRE of the coefficients ( $\lambda_\beta$ ) as the metric and only Start I values:

1. With everything else at default, the convergence criterion was varied, and the “best” criterion noted. For example, better performance might be observed with “convergence on the residual sum of squares” (RSS) than “convergence on parameters” (PARAM).
2. Having determined the best criterion, a good convergence tolerance was sought. If the default tolerance is 1.E-6, it was changed to the minimum tolerance—for example, 1.E-12. If a difference was observed, the tolerance was gradually increased to find the largest acceptable tolerance which is less than default.
3. Having determined criterion and tolerance, the method of solution was varied—for example, default Gauss–Newton was changed to the optional Levenberg–Marquardt.
4. Having determined criterion, tolerance, and method, the better form of derivative was chosen—that is, numerical or analytic and, if analytic, whether analytic first- and second-derivatives or just analytic first-derivatives.
5. Using results from the previous steps, obvious improvements were sought. For example, if the criterion was

determined using numerical derivatives, but analytic are much better, Step 4 was repeated to examine whether the use of analytic derivatives reverses the conclusion as to criterion.

Obviously, finding a preferred combination can involve several dozen nonlinear regressions. Developers frequently can provide useful guidance in choosing a preferred combination, and alleviate the burden of running so many nonlinear regressions. Once the preferred combination is determined, it is used on all 27 problems with Start I. If the preferred combination yields no solution, then other methods should be tried before resorting to Start II. There is also a Start III: use the certified values as the starting values; in some packages this can produce anomalous behavior. Note that some of the nonlinear problems use  $\pi$ ; if the package does not offer this constant, it should be defined to 16 digits.

The choice of derivative calculation merits attention as well, since derivatives are central to nonlinear estimation. It is well-known that exact analytic derivatives are more accurate than their numerical approximations (Bard 1974, p. 117; Kennedy and Gentle 1980, p. 474). The general rule is that analytic first and numerical second derivatives are as good as analytic first and second derivatives, though there are exceptions such as GARCH estimation (Fiorentini, Calzolari, and Panattoni 1996). See Donaldson and Schnabel (1987) for further comparison of analytic and numerical derivatives in the context of nonlinear estimation.

Packages vary in the way they accommodate derivatives. Some packages such as SAS have automatic analytic differentiation. Many packages offer only numerical derivatives. Other packages, such as SPSS, use numerical derivatives by default, but can accommodate analytic derivatives if the user calculates and codes them. S-Plus is none of the above. It uses numerical derivatives by default, and allows analytic derivatives to be supplied, but does not require the user to calculate or code them. S-Plus has a built-in facility for calculating and coding analytic derivatives: the “deriv” command is used for analytic first derivatives, and Smith’s (1992) “deriv3” extension produces analytic second derivatives.

Requiring users to supply analytic derivatives is yet a further impediment to reliable nonlinear estimation, because the not-insubstantial effort involved frequently leads users to eschew analytic derivatives completely, and instead rely on default numerical differentiation (Dennis 1984, p. 1766). Such an approach is not conducive to accuracy, as noted by Gill, Murray, and Wright (1981, p. 285), “[I]f computing first derivatives is merely inconvenient, the user should be aware of the increased complexity and decreased reliability that result when exact first derivatives are not available to an algorithm.” This decreased reliability can easily be verified by using a package to run the StRD nonlinear problems twice: once with numerical derivatives and again with analytic derivatives.

User-supplied analytic derivatives are twice ripe for error, since the user not only must calculate the derivatives, but also code them. Whether calculating or coding is more onerous depends upon the equation and number of param-

Univariate Statistics				Nonlinear Regression				
test	$\lambda_\mu$	$\lambda_s$	$\lambda_\rho$	test	start	$\lambda_\beta$	$\lambda_\sigma$	$\lambda_r$
PiDigits (l)	15	15	15	Misra1a (l)	I	9.2 (9.3)	8.9	10.5
Lottery (l)	15	15	14.9	Chwirut2 (l)	I	7.6 (6.2)	8.0	11
Lew (l)	15	15	14.8	Chwirut1 (l)	I	8.6 (5.9)	8.9	11
Mavro (l)	15	13.1	13.8	Lanczos3 (l)	I	6.7 (6.6)	6.7	10.6
Michelso (a)	15	13.8	13.4	Gauss1 (l)	I	8.7 (6.9)	8.5	11
NumAcc1 (a)	15	15	NA†	Gauss2 (l)	I	8.4 (8.4)	8.1	10.6
NumAcc2 (a)	14.0	14.2	15	DanWood (l)	I	10.1 (8.0)	10.1	11
NumAcc3 (a)	15	9.5	11.9	Misra1b (l)	I	10.1 (10.1)	9.9	11
NumAcc4 (h)	14.0	8.3	10.7	Kirby2 (a)	I	7.5 (6.4)	7.8	11
Analysis of Variance				Hahn1 (a)	I	7.8 (6.0)	8.6	10.6
test	$\lambda_F$			Nelson (a)	I	7.1 (5.5)	7.1	10.9
SiRstv (l)	8.3			MGH17 (a)	II	8.8 (ns)	8.3	11
SmnLsg01 (l)	13.3			Lanczos1 (a)	I	10.7 (10.6)	3.2	3.0
SmnLsg02 (l)	11.4			Lanczos2 (a)	I	10.3 (10.3)	9.0	10.2
SmnLsg03 (l)	11.8			Gauss3 (a)	I	9.2 (9.2)	8.6	11
AtmWtAg (a)	0			Misra1c (a)	I	10.5 (10.5)	10.5	11
SmnLsg04 (a)	0			Misra1d (a)	I	8.7 (11)	8.4	11
SmnLsg05 (a)	0			Roszman1 (a)	I	8.6 (5.5)	9.1	11
SmnLsg06 (a)	0			ENSO (a)	I	7.1 (2.8)	8.3	11
SmnLsg07 (h)	0			MGH09 (h)	II	6.5 (ns)	6.6	11
SmnLsg08 (h)	0			Thurber (h)	I	6.4 (4.4)	5.8	9.9
SmnLsg09 (h)	0			BoxBOD (h)	II	7.1 (ns)	7.1	10.4
Linear Regression				Rat42 (h)	I	8.3 (7.2)	8.0	11
test	$\lambda_\beta$	$\lambda_\sigma$	$\lambda_r$	MGH10 (h)	I	0 (0)	0	0
Norris (l)	12.3	11.9	11.6	Eckerle4 (h)	II	8.3 (ns)	8.3	10.7
Pontius (l)	11.4	9.2	8.9	Rat43 (h)	I	0 (0)	0	0
NoInt1 (a)	14.7	15	11.6	Bennett5 (h)	I	0 (0)	0	1.5
NoInt2 (a)	15	14.9	15	NONLINEAR OPTIONS: method=Gauss-Newton (default) criterion=PARAM tolerance=1E-6 derivatives=analytic (default) 'ns' = no solution default $\lambda_\beta$ in parentheses				
Filip (h)	ns							
Longley (h)	8.6	10.3	10.8					
Wampler1 (h)	8.3	15	15					
Wampler2 (h)	10.0	15	15					
Wampler3 (h)	7.0	10.9	10.8					
Wampler4 (h)	7.0	11.5	14.8					
Wampler5 (h)	7.0	11.5	15					

†three values insufficient to calculate  $\rho$ .

Figure 1. StRD Results for SAS V6.12.

eters. Several benchmarks have seven or eight parameters, and ENSO has nine, for which gradient and Hessian combined produce 54 supplementary equations. Symbolic computation can drastically relieve the first burden. A package such as *Mathematica v3.01* (Wolfram 1996), which was used for this project, can completely eliminate both burdens by displaying the 54 ENSO equations in FORTRAN-style text complete with proper command syntax for immediate cut-and-paste into the statistical package's program editor. Not only does this method offer speed, it eliminates the inevitable transcription errors which occur when cod-

ing a displayed equation to FORTRAN-style text. The first and second derivatives for the StRD nonlinear problems are available at the TAS ftp site (<ftp://www.amstat.org/tas>).

The remainder of this article is organized as follows. Section 2 applies the StRD to the three packages and compares the results. The comparison raises some interesting numerical issues, which are explored in detail. Section 3 discusses some technical details for applying the DIEHARD tests, and presents the DIEHARD results for the three packages. Mention is made of the fact that none of the developers provides users with necessary information concerning its random number generator (RNG). Section 4 uses ELV to assess the accuracy of selected statistical distributions, and

references to more complete assessments are given. Section 5 presents the conclusions.

## 2. StRD

The datasets and the certified values provided by NIST are in double-precision. Users of packages that offer single-precision storage with an option for double-precision storage should be sure to invoke the double-precision option, to ensure that the data are correctly read by the program. [Thanks to P. Lachenbruch for pointing this out.] Such users should also be aware that single-precision storage can have an adverse effect on accuracy, even when the input data are single-precision. This can be seen with a problem for which the dataset is single-precision, such as the linear regression problem in Longley (1967). Solve the problem once with single-precision storage, and again with double-precision storage.

### 2.1 SAS

*Univariate Statistics*—The mean and standard deviation were calculated using the “MEANS” command. The first-order autocorrelation coefficient was calculated using the “IDENTIFY” option of the “ARIMA” command from the ETS package. The results, shown in Figure 1, are quite accurate, though why it should fail to calculate the first-order autocorrelation coefficient for NumAcc1, which has three observations, is not clear.

*Analysis of Variance*—The “ANOVA” command generated the statistics. Observe that performance degrades precipitously after the lower difficulty problems. A nearly identical degradation occurs using “GLM” to solve these problems, but not with “ORTHOREG”, which appears to be the preferred method.

*Linear Regression*—“REG” calculated the statistics. It indicated near-singularity for FILIPPEL, and so the “ORTHOREG” command was then used to obtain an accurate solution. All the results are quite accurate. Note that the expectation of ten digits of accuracy for a linear problem applies only to the lower level of difficulty problems.

*Nonlinear Regression*—“PROC NLIN” was used, which employs automatic analytic differentiation. Default method of solution is Gauss–Newton. Other methods are Marquardt, Newton, DUD (*regula falsi*), and steepest descent. Available convergence criteria are RSS and PARAM, the former being default. Trying various combinations of the options resulted in the following preferred combination: convergence criteria PARAM with tolerance 1E-6, analytic derivatives (default), and the Gauss–Newton method (default). Compared to default estimation, this preferred combination 8 times gave the same answer (including zero digits of accuracy three times), twice was less accurate by an average 2.4 digits, and 13 times was more accurate by an average 1.8 digits.

SAS does not provide initial estimates for parameters. In no case, including using other solution methods, could convergence be obtained for MGH17, MGH09, BoxBOD and

Eckerle4 from Start I (it was sometimes necessary to check the log file to determine that the output file really should not have reported results). Zero digits of accuracy were computed for MGH10, Rat43, and Bennett5 from Start I. Additional nonlinear procedures are “MODEL” in SAS/ETS and “NLP” in SAS/OR. Both MODEL and NLP will solve BoxBOD from Start I, while NLP will also solve MGH17 from Start I. As these commands are not part of the base system, their results are not presented. Solving from Start III produced no anomalies.

StRD results are presented in Figure 1. To see that the preferred combination does produce better solutions, the LRE of the coefficient produced by default estimation is given in parentheses next to the LRE of the coefficient produced by the preferred combination. For each dataset the table shows: the minima of the LREs for the coefficients ( $\lambda_\beta$ ) and the standard errors of the coefficients ( $\lambda_\sigma$ ); the LRE for the residual sum of squares ( $\lambda_r$ ); and the difficulty of the dataset (lower, average, higher) indicated by a parenthetical “l”, “a”, or “h”. If the procedure fails to produce a solution, this is indicated by “ns”.

### 2.2 SPSS

*Univariate Statistics*—The command “descriptives X” with the option “/statistics=mean,” “stddev” produced the mean and standard deviation for variable X. These results, shown in Figure 2, are quite accurate. The “acf” command produced the first-order autocorrelation coefficient, which could not be calculated for NumAcc1. Since “acf” displays only three decimal places, this understates accuracy for the first five tests and overstates accuracy for the last three tests.

*Analysis of Variance*—For two variables, X (treatment) and Y (response), where X takes on integer values between min and max, inclusive, the following command was issued: “oneway Y by X(min,max)”. SPSS handles the lower level of difficulty, but performance degrades precipitously thereafter. For the dataset AtmWtAg it does not report an F-statistic; the system missing-value is returned.

One-way analysis of variance tests can be conducted four other ways in SPSS: “means y by x /stat=anova”; “anova y by x(min,max)”; “manova Y by X(min,max)”; and by “glm y by x”. The results for “means” in SmnLsg04 includes a negative between-group sum of squares. The command “manova” is more accurate than the others.

*Linear Regression*—The command “regression /variables=Y,X1,X2 /criteria=tolerance(1.0E-12) /dependent=Y /method=enter” was used. SPSS will refuse to include all the independent variables if the tolerance is exceeded. This occurred for several of the problems, so the tolerance was set at 1E-12 (default is .0001). Even this was insufficient to produce a solution for Filip, a tenth-order polynomial. Overall, the results are quite accurate.

*Nonlinear Regression*—The nonlinear estimation command is “NLR”. The default (and only) method for this command is Levenberg–Marquardt, and starting values must be provided by the user. There are three convergence criteria: RSS, PARAM, and RCON (correlation between

Univariate Statistics				Nonlinear Regression				
test	$\lambda_\mu$	$\lambda_s$	$\lambda_\rho$	test	start	$\lambda_\beta$	$\lambda_\sigma$	$\lambda_r$
PiDigits (l)	14.7	15	0	Misra1a (l)	I	6.1 (6.1)	6.8	5.0
Lottery (l)	15	15	3.4	Chwirut2 (l)	I	7.5 (6.2)	6.5	8.9
Lew (l)	15	13.2	3.0	Chwirut1 (l)	I	7.1 (5.8)	6.1	9.5
Mavro (l)	15	12.1	4.9	Lanczos3 (l)	I	6.9 (6.8)	6.9	6.7
Michelso (a)	15	12.4	3.4	Gauss1 (l)	I	7.4 (6.8)	6.0	8.6
NumAcc1 (a)	15	15	NA†	Gauss2 (l)	I	7.4 (7.4)	6.3	9.2
NumAcc2 (a)	15	15	15	DanWood (l)	I	9.5 (8.0)	8.1	7.0
NumAcc3 (a)	15	9.5	15	Misra1b (l)	I	6.7 (6.7)	6.3	4.2
NumAcc4 (h)	15	8.3	15	Kirby2 (a)	I	7.7 (5.0)	6.7	7.2
Analysis of Variance				Hahn1 (a)	I	5.4 (5.4)	4.4	6.0
test	$\lambda_F$			Nelson (a)	I	6.5 (5.8)	7.1	6.1
SiRstv (l)	9.6			MGH17 (a)	I	7.6 (ns)	5.9	7.3
SmnLsg01 (l)	15			Lanczos1 (a)	I	9.6 (9.5)	3.3	3.0
SmnLsg02 (l)	15			Lanczos2 (a)	I	8.7 (6.2)	6.4	7.1
SmnLsg03 (l)	12.7			Gauss3 (a)	I	7.6 (7.5)	5.5	8.5
AtmWtAg (a)	miss‡			Misra1c (a)	I	5.9 (5.9)	5.9	4.1
SmnLsg04 (a)	0			Misra1d (a)	I	6.1 (6.1)	5.8	4.9
SmnLsg05 (a)	0			Rozzman1 (a)	I	6.6 (5.5)	5.6	7.3
SmnLsg06 (a)	0			ENSO (a)	I	0 (0)	0	0
SmnLsg07 (h)	0			MGH09 (h)	I	7.6 (4.2)	7.6	7.9
SmnLsg08 (h)	0			Thurber (h)	I	8.2 (4.6)	7.2	9.8
SmnLsg09 (h)	0			BoxBOD (h)	I*	6.9 (0)	7.0	8.6
Linear Regression				Rat42 (h)	I	6.8 (6.8)	5.2	6.4
test	$\lambda_\beta$	$\lambda_\sigma$	$\lambda_r$	MGH10 (h)	I	7.1 (0)	6.3	7.3
Norris (l)	12.3	10.2	9.9	Eckerle4 (h)	I	9.9 (7.2)	8.0	6.8
Pontius (l)	12.5	8.9	8.6	Rat43 (h)	I	8.8 (5.1)	8.6	9.7
NoInt1 (a)	14.7	12.5	12.8	Bennett5 (h)	I	9.9 (ns)	10.1	7.1
NoInt2 (a)	15	14.3	13.0	NONLINEAR OPTIONS: method=Levenberg-Marquardt (default) criterion=PARAM tolerance=1E-12 derivatives=analytic gradient supplied 'ns' = no solution default $\lambda_\beta$ in parentheses				
Filip (h)	ns							
Longley (h)	12.1	13.3	13.2					
Wampler1 (h)	6.6	6.6	15					
Wampler2 (h)	9.7	9.7	15					
Wampler3 (h)	7.4	10.6	10.8					
Wampler4 (h)	7.4	10.8	14.2					
Wampler5 (h)	5.8	10.8	15					

†three values insufficient to calculate  $\rho$ . ‡Reported system missing-value.

\* - using **CNLR** command.

Figure 2. StRD Results for SPSS v7.5.

residuals and derivatives). By default they operate simultaneously each with tolerance 1E-8; if any of the criteria is reached, the procedure terminates. Each can be disabled, though, so it is possible to use just one criterion; this was done. The preferred combination is: PARAM convergence criterion, tolerance 1E-12, analytic gradient. Compared to default estimation, both provided zero digits of accuracy for the same problem (ENSO), but default provided zero digits of accuracy for two more (BoxBOD and MGH10), and

produced no solution for two (MGH17 and Bennett5). Of the remaining 22 problems, 7 times they tied and 15 times the preferred combination was more accurate by an average 1.6 digits. SPSS also offer the "CNLR" command for nonlinear regression subject to constraints. Using this command without supplying constraints offers another method for solving unconstrained nonlinear least squares problems.

A solution for all problems was produced from Start I, though in the case of ENSO this solution had zero digits of accuracy. For BoxBOD the "CNLR" command was

Univariate Statistics				Nonlinear Regression				
test	$\lambda_\mu$	$\lambda_s$	$\lambda_\rho$	test	start	$\lambda_\beta$	$\lambda_\sigma$	$\lambda_r$
PiDigits (l)	15	15	6.8	Misrala (l)	I	9.3 (6.8)	9.0	10.5
Lottery (l)	15	15	7.4	Chwirut2 (l)	I	7.6 (4.9)	8.0	11
Lew (l)	15	15	7.0	Chwirut1 (l)	I	7.3 (4.7)	7.7	11
Mavro (l)	15	13.1	7.1	Lanczos3 (l)	I	6.6 (4.5)	6.6	10.6
Michelson (a)	15	13.8	7.3	Gauss1 (l)	I	8.7 (5.1)	8.5	11
NumAcc1 (a)	15	15	15	Gauss2 (l)	I	8.4 (4.7)	8.1	10.6
NumAcc2 (a)	14.0	15	7.1	DanWood (l)	I	8.0 (5.9)	8.1	11
NumAcc3 (a)	15	9.5	7.1	Misralb (l)	I	9.3 (6.8)	9.0	11
NumAcc4 (h)	14.0	8.3	7.3	Kirby2 (a)	I	7.4 (4.2)	7.8	11
Analysis of Variance				Hahn1 (a)	I	7.6 (4.4)	8.3	10.0
test	$\lambda_F$			Nelson (a)	I	7.6 (0)	7.7	10.9
SiRstv (l)	13.3			MGH17 (a)	I*	7.9 (ns)	7.5	11
SmnLsg01 (l)	14.5			Lanczos1 (a)	I	10.6 (10.6)	3.3	3.0
SmnLsg02 (l)	14.3			Lanczos2 (a)	I	10.3 (5.5)	10.0	9.8
SmnLsg03 (l)	12.9			Gauss3 (a)	I	9.2 (6.4)	8.6	11
AtmWtAg (a)	9.7			Misralc (a)	I	8.1 (8.0)	7.8	11
SmnLsg04 (a)	10.4			Misrald (a)	I	9.4 (6.8)	9.1	11
SmnLsg05 (a)	10.2			Roszman1 (a)	I	7.0 (3.9)	7.5	12.2
SmnLsg06 (a)	10.2			ENSO (a)	I	5.6 (2.5)	6.8	11
SmnLsg07 (h)	4.6			MGH09 (h)	I*	6.7 (ns)	7.0	11
SmnLsg08 (h)	2.7			Thurber (h)	I	6.9 (3.7)	6.2	9.9
SmnLsg09 (h)	0			BoxBOD (h)	I	7.8 (7.7)	8.0	10.4
Linear Regression				Rat42 (h)	I	7.6 (ns)	6.9	11
test	$\lambda_\beta$	$\lambda_\sigma$	$\lambda_r$	MGH10 (h)	II	10.3 (ns)	10.3	11
Norris (l)	12.5	14.1	13.8	Eckerle4 (h)	I*	9.2 (ns)	9.4	10.7
Pontius (l)	12.7	13.2	12.9	Rat43 (h)	I*	8.2 (ns)	8.4	11
NoInt1 (a)	14.7	14.4	14.0	Bennett5 (h)	I	10.3 (4.8)	10.1	11
NoInt2 (a)	15	15	14.9	NONLINEAR OPTIONS: method=Gauss-Newton (default) criterion=RSS (default) tolerance=1E-6 derivatives=analytic gradient supplied 'ns' = no solution default $\lambda_\beta$ in parentheses				
Filip (h)	7.1	7.0	7.8					
Longley (h)	13.0	14.2	14.1					
Wampler1 (h)	9.8	15	15					
Wampler2 (h)	13.5	15	15					
Wampler3 (h)	9.2	13.5	15					
Wampler4 (h)	7.5	13.6	15					
Wampler5 (h)	5.5	13.5	15					

\* solved using **nlregb** command

Figure 3. StRD Results for S-Plus v4.0.

used because no solution could be obtained with “NLR”, no matter what options were invoked. Solving from Start III, produced no anomalies. StRD results are presented in Figure 2.

### 2.3 S-Plus

**Univariate Statistics**—The “mean” function is used to obtain the mean. To obtain the standard deviation, the “var” function is used and its square root is taken. These results, shown in Figure 3, are quite accurate. The “acf” command produced the first-order autocorrelation coefficient. It

yielded about seven digits of accuracy which, for a linear procedure, is less than can be reasonably expected.

**Analysis of Variance**—The one-way analysis of variance was conducted using the “aov” command. Performance clearly degrades as the level of difficulty increases. This is to be expected, and will be addressed at length in the next subsection.

**Linear Regression**—The “lm” command was used to run the linear regressions. At default the program refused to compute a solution for Filip because it encountered a singularity. However, the “lm” command has an option for

Table 1. S-Plus LREs for SmnLsg Problems

<i>n</i>	<i>d</i>		
	1	10	13
189	SmnLsg01 (14.5)	SmnLsg02 (14.3)	SmnLsg03 (12.9)
1809	SmnLsg04 (10.4)	SmnLsg05 (10.2)	SmnLsg06 (10.2)
18009	SmnLsg07 (4.6)	SmnLsg08 (2.7)	SmnLsg09 (0)

NOTE: *d*: number of constant leading digits; *n*: number of observations

setting the tolerance. When tolerance was set at 1E-10, an answer was produced. The results are quite accurate.

**Nonlinear Regression**—The “nls” command was used to run the nonlinear regressions. The default (and only) estimation method is Gauss–Newton. The default (and only) convergence criterion is residual sum of squares. S-Plus does not offer default starting values. The preferred combination is: tolerance 1E-6, RSS convergence (default), Gauss–Newton method (default), and analytic gradient. Comparing the preferred combination to default estimation, both failed to provide a solution to five of the problems. Twice default failed where the preferred combination succeeded and once they were tied. In the remaining 19 cases, the preferred combination was more accurate by an average 2.8 digits.

S-Plus also offers the “nlregb” command for nonlinear regression subject to constraints. Using this command without supplying constraints is another way to solve unconstrained nonlinear least squares problems. This command solved from Start I the problems which “nls” with analytic gradient could not. To obtain standard errors from this command is a programming exercise. A much easier way, which is used here, is to take the solution from this command and feed it to the “nls” command to obtain standard errors.

All problems but MGH10 are solved from Start I, though for four problems it was necessary to resort to “nlregb”. In no case was zero digits of accuracy produced. Solving from Start III produced no anomalies. StRD results are presented in Figure 3.

## 2.4 Comparisons

For the univariate summary statistics, all packages accurately compute the mean and standard deviation. The first-order correlation coefficient is another matter. While SAS does well, the accuracy of SPSS cannot be determined because the user can neither access this statistic nor control the number of displayed digits. The S-Plus routine appears to be weak, delivering fewer accurate digits than can be expected for such a routine. Both SAS and SPSS inexplicably refuse to compute for NumAcc1, which has three observations.

ANOVA results vary greatly. SAS and SPSS cannot pass the average difficulty problems. S-Plus appears to do poorly on the higher difficulty problems; however, its results are approximately the same as can be obtained by using an accurate linear regression routine with appropriate dummy

variables. This suggests that the S-Plus results do not indicate a bad software but, rather, that these problems can exhaust the capabilities of 32-bit double-precision computation. By comparison, the SAS and SPSS routines clearly do not implement effective algorithms. Further analysis of the S-Plus results is instructive, and will be considered anon.

As far as linear regression is concerned, the Longley (1967) lesson has been well-learned: all packages demonstrate reliability on all of the datasets. The same cannot be said for nonlinear regression, which presents some interesting results. From a general perspective, two things are clear. First, analytic derivatives are preferable to numerical derivatives. Second, default estimation is not reliable, and for each package there exists a preferred combination of options. S-Plus never returns a solution with zero digits of accuracy, and solves from Start II only one time; the default numerical derivatives must be replaced with analytic first-derivatives to achieve this accuracy. SPSS always solves from Start I, but produces one solution with zero accurate digits; the default numerical derivatives must be replaced with user-supplied analytic first derivatives to achieve this accuracy. SAS three times produces zero accurate digits and four times solves from Start II, and has automatic analytic differentiation as the default.

Returning to the S-Plus performance on the ANOVA datasets, consider the dataset SmnLsg01, which has 189 observations on a treatment variable, *X*, which assumes integer values, and a response variable, *Y1*, which assumes values of unity plus a single decimal, for example, 1.2, 1.5, etc.; the mean of *Y1* is 1.4 exactly. The data of SmnLsg02 and SmnLsg03 are of the same magnitude, but with sample sizes of 1,809 and 18,009, respectively. The dataset SmnLsg04 has the same treatment variable as SmnLsg01, and the response variable *Y4* is the same as *Y1* except nine zeroes have been inserted between the 1 and the decimal, so its mean is 1000000000.4 exactly. Similarly, *Y7* from SmnLsg07 has twelve zeroes inserted, so its mean is 1000000000000.4 exactly.

Algebraically, SmnLsg01, SmnLsg04, and SmnLsg07 are the same problem, in the sense that they give rise to identical ANOVA tables (degrees of freedom, sums of squares, etc.). Numerically, though, they differ on a finite precision computer, as was seen. The same is true for SmnLsg02, SmnLsg05, SmnLsg08 and SmnLsg03, SmnLsg06, SmnLsg09. The larger the number of constant leading digits, the more difficult it becomes to calculate the sum of squares, because subtracting the treatment means from the overall mean produces cancellation error.

Constant leading digits are not the only problem—the number of observations can also reduce accuracy through the effect of cumulated rounding error. Some readers may be surprised to see that in the present example, the number of observations has a more deleterious effect than the number of leading digits. This is shown clearly in Table 1, which displays the dataset names according to the number of digits to the left of the decimal (*d*) and the sample size (*n*), together with the S-Plus LREs in parentheses.

The effect of increasing the number of leading digits (look across any row) is not nearly as serious as increas-

ing the number of observations (look down any column). The degradation is not serious going from 189 to 1,809 observations, but is grave going from 1,809 to 18,009 observations. Note that the S-Plus results are approximately the same as can be obtained by solving these ANOVA problems using an accurate linear regression routine with appropriate dummy variables, so it is clear that the problem is not with the software—the data have exhausted the capabilities of the 32-bit double-precision. For these problems, packages which can use symbolic calculation can achieve remarkable accuracy. *Mathematica v3.01*, for example, can solve all the StRD ANOVA problems to 15 digits of accuracy using Hunka's (1997) add-on package.

This is not to suggest that these problems are beyond accurate numerical computation using 32-bit double-precision. Again using linear regression, the usual iterative refinement technique for solving linear systems (Higham 1997, sec. 11) can produce 4.2 accurate digits for SmmLsg09, though it may be necessary to center the response variable if the total sum of squares is computed separately. The ANOVA tests underscore the importance of the researcher being aware of the limits of finite precision calculation and the algorithm being used, and having some idea of whether the problem at hand exceeds those limits. Clearly the S-Plus algorithm is not appropriate for large datasets, to say nothing of the SAS and SPSS algorithms. Yet, there are many disciplines where analyzing tens or even hundreds of thousands of observations is not uncommon. Published articles analyzing such large datasets rarely describe any procedures undertaken to determine whether the size of the dataset has exhausted the computer's finite precision (Kennedy and Gentle 1980, secs. 3.8 and 8.1.3).

### 3. RANDOM NUMBER GENERATOR

The DIEHARD Battery of Randomness Tests was not written with user-friendliness in mind, and the documentation is a bit sketchy, so some mention of how to use the program is appropriate. The input to the DIEHARD program is a file of some three million random 32-bit integers,

though signed 32-bit integers may be more practical for some users. [Note: Signed 32-bit integers are mentioned here because QBASIC, available through MS-DOS, will effect the necessary hexadecimal conversion for 32-bit signed integers, but not for 32-bit integers per se. Users with access to FORTRAN or some other package with more robust hexadecimal conversion can use 32-bit integers.] Letting RAN be a call to a uniform (0,1) generator, each integer is calculated as  $\text{INT}(-2147483649 + 4294967297 \cdot \text{RAN})$ , where  $\text{INT}(x)$  truncates the decimal part of  $x$ . The statistical package writes these to an ASCII file, which then must be converted to a specifically structured hexadecimal file (formatting details are in the DIEHARD documentation). Because some statistical packages offer neither hexadecimal conversion nor sufficient control over output formatting, it may be necessary to use a programming language to convert the integers to a hex file. An auxiliary program then converts the hex file to a binary file, which is the input to the main DIEHARD program that actually computes the tests. Of crucial importance is that the statistical package write the integers to the ASCII file in the order that they are generated, and that this order be preserved in the conversion to hexadecimal.

All the packages reviewed herein have reproducible RNGs. Whether any of the packages discussed herein meets the remaining desiderata of an RNG (Ripley 1990) cannot be determined from the user manuals. SAS cites Fishman and Moore (1982) for its RNG, but offers no further details. SPSS and S-Plus hardcopy manuals provide no details of their RNGs. None of the vendors provides the algorithm, its period, or the statistical tests it has passed. This is a serious omission. The S-Plus on-line documentation refers to a "modified" Marsaglia Super-Duper RNG, but does not describe the modifications. This may be misleading, since it is well-known that the Super-Duper passes all the DIEHARD tests for randomness. This bears out Knuth's (1997, p. 26) admonition against modifying RNGs. Test results appear in Table 2. Passing a test merits a lowercase "p" while failing a test earns an uppercase "F".

Table 2. Results of Marsaglia's DIEHARD Tests

<i>Test</i>	<i>SAS</i>	<i>SPSS</i>	<i>S-Plus</i>
Birthday Spacings Test	p	p	p
Overlapping 5-Permutation Test	p	p	p
Binary Rank for $31 \times 31$ Matrices	p	p	p
Binary Rank for $32 \times 32$ Matrices	p	p	p
Binary Rank for $6 \times 8$ Matrices	p	p	p
Bitstream Test ( <i>p</i> values)	p	p	p
OPSO Test	p	p	p
OQSO Test	p	p	F
DNA Test	p	p	F
Count the Ones Test (stream of bytes)	F	F	F
Count the Ones Test (specific byte)	p	p	F
Parking Lot Test	p	p	p
Minimum Distance Test	p	p	p
3-D Spheres Test	p	p	p
Squeeze Test	p	p	p
Overlapping Sums Test	p	p	p
Runs Test	p	p	p
Craps Test	p	p	p



Table 3. SAS  $F(x, 201, 10001)$  Distribution

$x$	exact value	SAS	relerr	$x$	exact value	SAS	relerr
1	.51287	exact	0	.50	3.7174E-10	4.3115E-10	.1598
.9	.15967	.15962	.0003	.40	1.4682E-15	2.2760E-15	.5502
.8	.017628	.017643	.0009	.38	6.2755E-17	1.1102E-16	.7691
.7	.00045779	.00046236	.0100	.37	1.1716E-17	0	1.0
.6	1.6421E-6	1.7170E-6	.0456	.30	9.2465E-24	0	1.0

All three vendors offer RNGs with periods approximately  $2^{31}$ . Since DIEHARD presumes a period of  $2^{32}$ , SAS and SPSS do about as well as can be expected—though S-Plus clearly does not. As discussed in Part I, none of the three RNGs is suitable for intensive use due to deficient period length. When software developers begin implementing RNGs suitable for intensive use, more sophisticated tests will be necessary. Marsgalia is planning an upgrade to DIEHARD, and L'Ecuyer's (L'Ecuyer and Hellekalek in press) TESTU01 program is in the testing stage, with release tentatively scheduled for some time in the coming year.

#### 4. STATISTICAL DISTRIBUTIONS

We are all familiar with the normal, Student's  $t$ , and other such distributions found in the appendix of a statistics text. It may come as a surprise to find that similar percentiles generated by a computer program are not always more accurate than these tables. It is important, therefore, that the quality of statistical distributions be assessed. Serious errors have been found in some packages, (Knüsel 1995, 1998; McCullough 1999). Note that none of the vendors describes the algorithms used to compute these quantities, nor provides the limits within which they work reliably. This is a serious omission.

Only a few examples of testing distributions will be given, because thorough investigations of both SAS v6.08 (Knüsel 1997a) and S-Plus v3.3 (Knüsel 1997b) have been done. Both of these studies are must-reading for the users of the respective packages. In general, both packages are fast and accurate, though each package has some inconsistencies which would be of interest to its users. A similar study of SPSS would be must-reading for its users.

Neither SAS nor S-Plus made significant changes to its statistical distributions in subsequent versions. The examples presented are taken directly from Knüsel's studies, and were verified for the current versions of the software. The exact value ( $c$ ) computed by ELV, the estimated value ( $x$ ) computed by the statistical packages, and the relative error ( $= |x - c|/c$ ) are presented. If  $\text{relerr} > 1$  it is set to unity.

#### 4.1 SAS

*Normal distribution*—"PROBNORM( $x$ )" computes the probability that a standard normal variate is less than  $x$ . The lower tail probability *seems to be correct* (relative error less than  $1\text{E-}6$ ) for very small probabilities ( $1\text{E-}100$  and even smaller).

*Chi-square distribution*—"PROBCHI( $x, k$ )" computes the probability that a chi-square distributed variate with  $k$  degrees of freedom is less than  $x$ . The lower-tail probability with  $k$  degrees of freedom *seems to be correct* (relative error less than  $1\text{E-}6$ ) for probabilities as small as  $1\text{E-}100$  and for  $k$  as large as  $1\text{E+}08$ . Real values are admitted for  $k$ .

*F-distribution*—"PROBF( $x, m, n$ )" computes the probability that an F-distributed variate with  $m$  and  $n$  degrees of freedom is less than  $x$ . The lower-tail probability with degrees of freedom  $m$  and  $n$  degrees can be incorrect for extreme values of the degrees of freedom, as shown in Table 3. Results for other distributions can be found in Knüsel (1997a).

#### 4.2 SPSS

*Normal distribution*—"CDF.NORM( $x$ )" computes the probability that a standard normal variate is less than  $x$ . The lower-tail probability *seems to be correct* (relative error less than  $1\text{E-}5$ ) for probabilities as small as  $1\text{E-}8$ .

*F-distribution*—"CDF.F( $x, m, n$ )" computes the probability that an F-distributed variate with  $m$  and  $n$  degrees of freedom is less than  $x$ . The lower-tail probability with degrees of freedom  $m$  and  $n$  degrees can be incorrect for extreme values of the degrees of freedom, as shown in Table 4.

#### 4.3 S-Plus

*Normal distribution*—"dnorm( $x$ )" computes the probability that a standard normal variate is less than  $x$ . The lower-tail probability *seems to be correct* (relative error less than  $1\text{E-}6$ ) for probabilities as small as  $1\text{E-}8$ .

Table 4. SPSS  $F(x, 201, 10001)$  Distribution

$x$	exact value	SPSS	relerr	$x$	exact value	SPSS	relerr
1.0	.51287	exact	.0	.50	3.7174E-10	4.9672E-10	.3362
.9	.15967	exact	.0	.40	1.4682E-15	8.2157E-15	1.0
.8	.017628	exact	.0	.38	6.2755E-17	6.6613E-16	1.0
.7	.00045779	.00045785	.0001	.37	1.1716E-17	2.2204E-16	1.0
.6	1.6421E-6	1.6791E-6	.0225	.30	9.2465E-24	.0	1.0

Table 5. S-Plus Student's-t Distribution

<i>x</i>	<i>n</i>	exact value	S-Plus	relerr
−9.7	100	2.251546E-16	2.220446E-16	.0138
−9.8	100	1.358964E-16	1.110223E-16	.1830
−9.9	100	8.202263E-17	5.551115E-17	.3232
−10.0	100	4.950844E-17	5.551115E-17	.1212
−10.1	100	2.988587E-17	5.551115E-17	.8574
−10.2	100	1.804321E-17	0.0	1

*Student's-t distribution*—"dt(x,k)" computes the probability that a *t*-distributed variate with *k* degrees of freedom is less than *x*. It has some problems in the extreme tails, as shown in Table 5.

*Chi-square distribution*—"dchisq(x,k)" computes the probability that a chi-square distributed variate with *k* degrees of freedom is less than *x*. It exhibits chaotic behavior for extreme values of the parameters, as shown in Table 6. Results for other distributions can be found in Knüsel (1997b).

Table 6. S-Plus Chi-Square Distribution

<i>x</i>	<i>n</i>	exact value	S-Plus	relerr
2E13	2E13	.5000000	.5233686	.0467
2E14	2E14	.5000000	.5902894	.1806
2E15	2E15	.5000000	.6478144	.2956
4E15	4E15	.5000000	−741.3565	1
6E15	6E15	.5000000	.7966655	.5933
8E15	8E15	.5000000	.8239072	.6478
1E16	1E16	.5000000	1.0	1

## 5. CONCLUSIONS

SAS, SPSS and S-Plus have been assessed using the methodology outlined in Part I (McCullough 1998). Flaws have been uncovered in all three areas: estimation, random number generation, and statistical distributions. Generally, all packages perform well on univariate summary statistics and linear regression test suites. For one-way analysis of variance, SAS and SPSS have inadequate routines, and the S-Plus routine exhibits a predictable degradation due to the limits of finite precision computation. SAS can correctly solve only 20 of the 27 nonlinear problems from Start I. SPSS correctly solves 26 of the 27 from Start I only when user-supplied analytic first derivatives are employed, and returns zero accurate digits once. S-Plus correctly solves all 27 problems, all but one from Start I. All three packages present both the random number generator (RNG) and statistical distributions as black boxes—without supplying critical information as to algorithm employed, and so on. All have RNGs that are inadequate due to deficient period length, and the S-Plus RNG fails too many of the DIEHARD tests. Previous work by Knüsel indicates that statistical distributions of SAS and S-Plus are satisfactory, but the adequacy of SPSS statistical distributions has yet to be demonstrated. As this article goes to press, all

three packages have released subsequent versions of their software, and may have remedied some of the deficiencies noted herein.

The reliability of statistical software cannot be taken for granted; neither can the reliability of econometric software (McCullough in press) or statistical functions in spreadsheet software (McCullough and Wilson in press). The tests applied herein only begin to inform users as to the reliability of their software. That the general linear and nonlinear routines can be shown to work well does not necessarily imply that specialized linear and nonlinear routines also work well. These, too, need to be benchmarked; several examples are given in McCullough and Vinod (in press). Perhaps a more widespread use of benchmarks in software reviewing will induce reviewers to develop new benchmarks for inclusion in their reviews, or afford more journals the incentive to publish new benchmarks.

An increase in the number of benchmarks and other means of assessing the reliability of statistical software and their use in software reviewing can only improve the quality of statistical software. This, in turn, will remediate the problem mentioned in the introduction of Part I: different packages giving different answers to the same problem.

[Received August 1998. Revised February 1999.]

## REFERENCES

- Bard, Y. (1974), *Nonlinear Parameter Estimation*, New York: Academic Press.
- Brown, B. W. (1998), "DCDFLIB v1.1" (Double precision Cumulative Distribution Function LIBrary), at <ftp://odin.mdacc.tmc.edu/pub/source>.
- Dennis, J. E., Jr. (1984), "A User's Guide to Nonlinear Optimization Algorithms," *Proceedings of the IEEE*, 72, 1765-1776.
- Donaldson, J., and Schnabel, R. (1987), "Computational Experience with Confidence Regions and Confidence Intervals for Nonlinear Least Squares," *Technometrics*, 29, 67-82.
- Fiorentini, G., Calzolari, G., and Panattoni, L. (1996), "Analytic Derivatives and the Computation of GARCH Estimates," *Journal of Applied Econometrics*, 11, 399-417.
- Fishman, G. S., and Moore, L. R. (1982), "A Statistical Evaluation of Multiplicative Congruential Generators with Modulus ( $2^{31} - 1$ )," *Journal of the American Statistical Association*, 77, 129-136.
- Gill, P. E., Murray, W., and Wright, M. H. (1981), *Practical Optimization*, London: Academic Press.
- Higham, N. J. (1997), *Accuracy and Stability of Numerical Algorithms*, Philadelphia, PA: SIAM.
- Hunka, S. (1997), *ANOVA.NB*, [www.mathsource.com](http://www.mathsource.com).
- Kennedy, W. J., and Gentle, J. E. (1980), *Statistical Computing*, New York: Marcel-Dekker.
- Knüsel, L. (1989), "Computergestützte Berechnung Statistischer Verteilungen," Oldenburg, München-Wien (An English version of the program is at [www.stat.uni-muenchen.de/~knuesel/elv](http://www.stat.uni-muenchen.de/~knuesel/elv)).
- (1995), "On the Accuracy of Statistical Distributions in GAUSS," *Computational Statistics and Data Analysis*, 20, 699-702.
- (1997a), "On the Accuracy of Statistical Distributions in S-Plus," unpublished manuscript, Department of Statistics, University of Munich, at [www.stat.uni-muenchen.de/~knuesel](http://www.stat.uni-muenchen.de/~knuesel).
- (1997b), "Note on the Accuracy of Some Statistical Distributions in SAS," manuscript, Department of Statistics, University of Munich, at [www.stat.uni-muenchen.de/~knuesel](http://www.stat.uni-muenchen.de/~knuesel).
- (1998), "On the Accuracy of Statistical Distributions in Microsoft Excel 97," *Computational Statistics and Data Analysis*, 26, 375-377.
- Knuth, D. E. (1997), *The Art of Computer Programming* (vol. 2, 3e), Reading, MA: Addison-Wesley.
- L'Ecuyer, P., and Hellekalek, P. (in press), "Random Number Generators:

- Selection Criteria and Testing,” in *Lecture Notes in Statistics No. 138: Random and Quasi-Random Point Sets*, New York: Springer.
- Longley, J. W. (1967), “An Appraisal of Computer Programs for the Electronic Computer from the Point of View of the User,” *Journal of the American Statistical Association*, 62, 819–841.
- Marsaglia, G. (1996), “DIEHARD: A Battery of Tests of Randomness,” at [stat.fsu.edu/pub/~diehard](http://stat.fsu.edu/pub/~diehard).
- McCullough, B. D. (1998), “Assessing the Reliability of Statistical Software: Part I,” *The American Statistician*, 52, 358–366.
- (in press), “Econometric Software Reliability: EViews, LIMDEP, SHAZAM and TSP,” *Journal of Applied Econometrics*.
- McCullough, B. D., and Vinod, H. D. (in press), “The Numerical Reliability of Econometric Software,” *Journal of Economic Literature*.
- McCullough, B. D., and Wilson, B. (in press), “On the Accuracy of Statistical Procedures in Microsoft Excel,” *Computational Statistics and Data Analysis*.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992), *Numerical Recipes in FORTRAN, 2e*, New York: Cambridge University Press.
- Ripley, B. D. (1990), “Thoughts on Pseudorandom Number Generators,” *Journal of Computational and Applied Mathematics*, 31, 153–163.
- Rogers, J., Filliben, J., Gill, L., Guthrie, W., Lagergren, E., and Vangel, M. (1998), “StRD: Statistical Reference Datasets for Assessing the Numerical Accuracy of Statistical Software,” NIST TN# 1396, Bethesda, MD: National Institute of Standards and Technology.
- Smith, D. (1992), “deriv3—an option to generate a Hessian function [for S-PLUS],” STATLIB at <http://www.stat.cmu.edu/S/>.
- Wolfram, S. (1996), *Mathematica 3.0 User's Guide*, New York: Cambridge University Press.