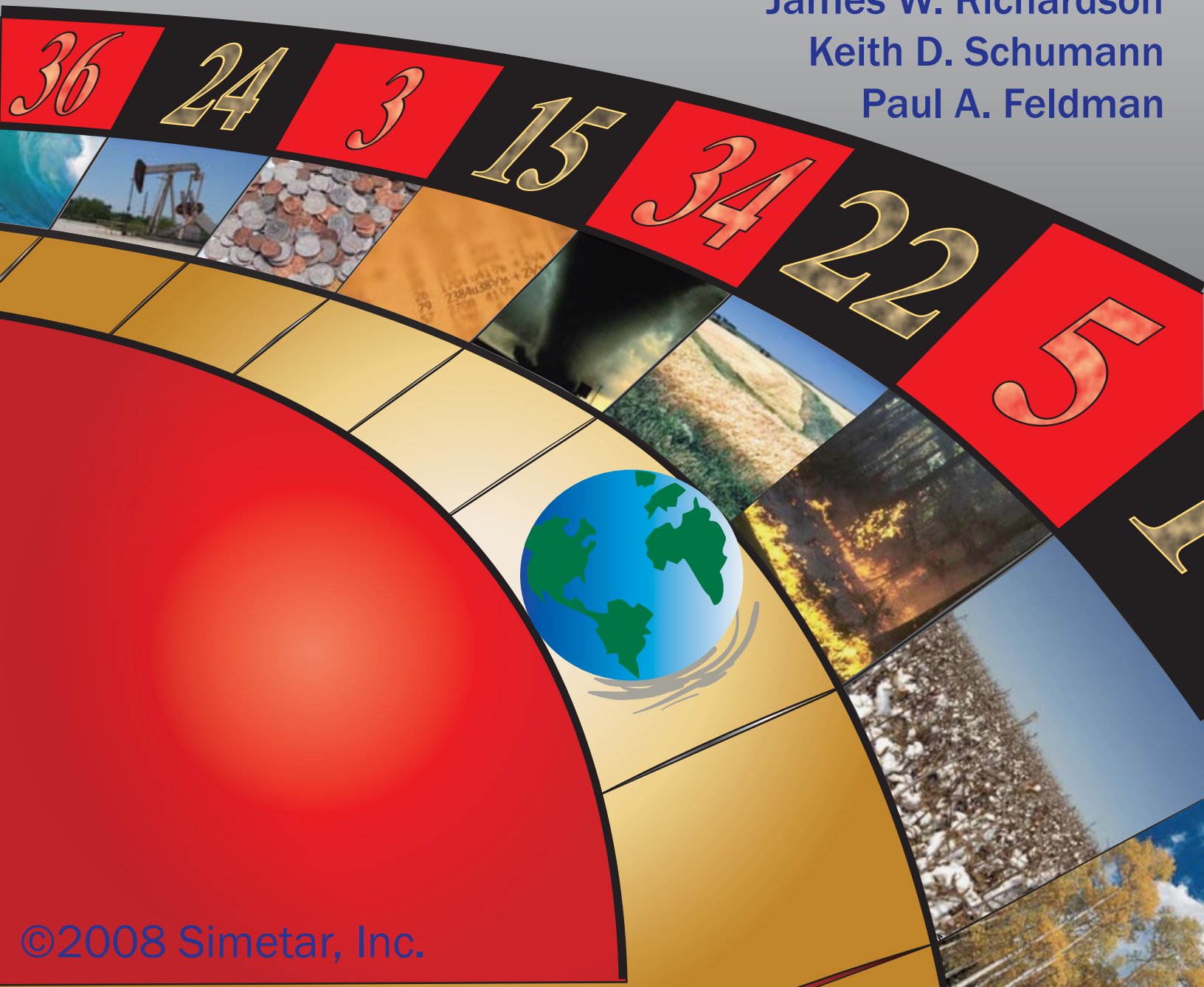# $imetar ©

Simulation & Econometrics to Analyze Risk

James W. Richardson
Keith D. Schumann
Paul A. Feldman

Simetar© 2008
Simulation & Econometrics To Analyze Risk

James W. Richardson

Keith Schumann

Paul Feldman

Table of Contents

Simetar© 2008
Simulation & Econometrics To Analyze Risk

1.0 What is Simetar?

Simetar© 2008 is a simulation language written for risk analysts to provide a transparent method for analyzing data, simulating the effects of risk, and presenting results in the user friendly environment of Microsoft® Excel[1]. Any Excel spreadsheet model can be made stochastic and simulated using Simetar functions. Simetar, an acronym for Simulation for Excel to Analyze Risk is an Excel add-in. Simetar requires little additional memory and operates efficiently on most PCs running Excel XP, Excel 2000, Excel 2003, and Excel 2007. Instructions for installing Simetar are provided in Section 1.1.

Simetar consists of Menu Driven and User Defined Functions for Excel. A common principle in Simetar, is that all functions are dynamic; so if changes are made to the original data most all parameters, hypothesis tests, regression models, and risk ranking strategies are automatically updated. This feature of having Excel dynamically recalculate parameters offers significant efficiencies during the development, validation, verification, and application of stochastic simulation models.

The more than 230 functions in Simetar can be categorized into six groups: (a) simulating random variables, (b) parameter estimation and statistical analyses, (c) graphical analysis, (d) ranking risky alternatives, (e) data manipulation and analysis, (f) multiple regression, and (g) probabilistic forecasting. Simetar can be used to perform all of the steps for developing, simulating, and applying a stochastic model in Excel, namely: estimate parameters for random variables, simulate stochastic variables, test the validity of the random variables, present the results graphically, and rank risky alternatives.

The next section describes the procedure for installing Simetar. After installing Simetar open the demonstration program to see learn how to apply the major functions in Simetar. More than 100 demonstration programs will be installed on your computer at **Start > Programs > Simetar > Demos**. Refer to these demonstration programs as you read the User's Manual to learn how the functions are applied in working simulation and forecasting models.

1.1 Installing Simetar

The first step in installation is to set the macro security level for Excel to low. (If you currently have Simetar installed be sure to uninstall Simetar and delete the C:\Program Files\Simetar folder.) After setting macro security to Low, close Excel and insert the Simetar CD in your computer's CD drive. (If you are installing from a file downloaded from the Simetar website, copy the file to your computer's hard drive and proceed with the installation.) From the Windows Explorer, double click on the Simetar.exe file name and the Setup Wizard will open to

---

[1] . Simetar© is copyrighted by the authors. Microsoft, Excel, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Figure 1. Click the Next box to proceed with the installation. The License Agreement is provided in the second screen of the Wizard (Figure 2). Read the License Agreement and click on the I Agree box to proceed with installation.
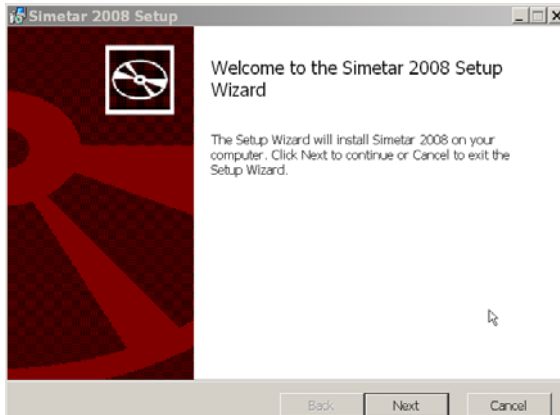
Figure 1. Install Simetar.
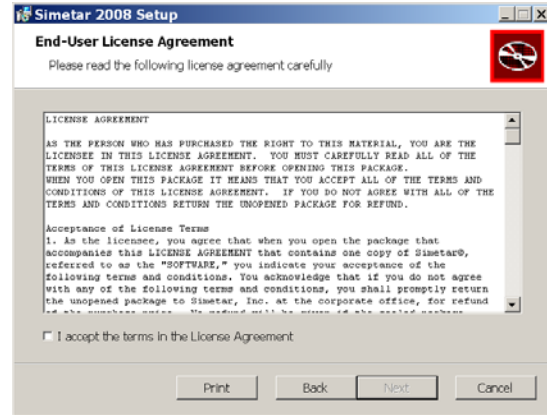
Figure 2. License Agreement.

Enter the License Code provided on your CD or with the Download Instructions (Figure 3). If you did not uninstall Simetar, a screen will appear that allows you uninstall using our unistaller. In the next screen select the "Typical" installation. Figure 4 is provided so you can change your mind as to the type of installation.
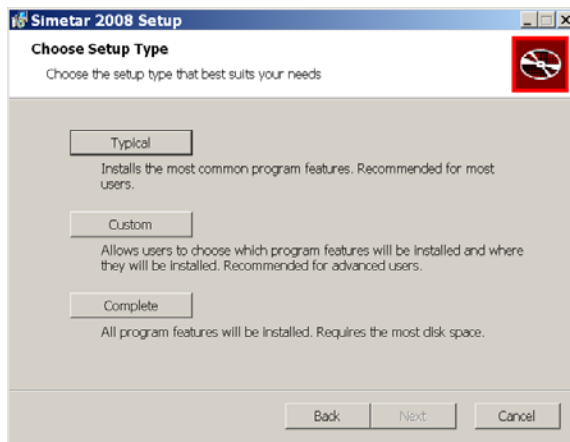
Figure 3. Type of Installation.
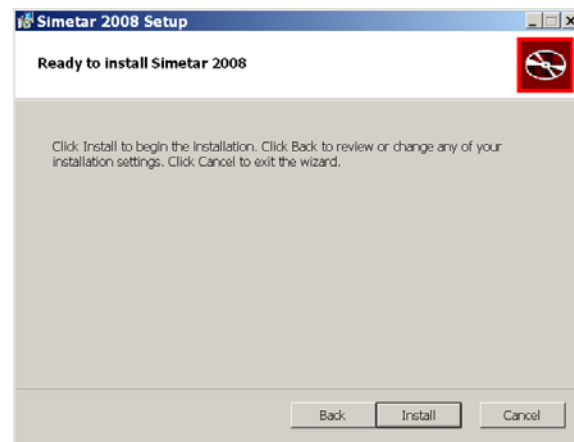
Figure 4. Final Chance to Change Your Installation Type.

Enter your license code in Figure 5, make sure all letters are caps and the dashes are included. The installation will take 2-3 minutes as the files are transferred and the appropriate files are updated so Simetar can operate in the Microsoft environment. The program will be stored in C:\Program Files\Simetar 2008. The last screen (Figure 6) indicates that Simetar has finished installing properly. Open Excel and you will see the Simetar toolbar in Excel 2003. For Excel 2007 you must click Add-Ins and then click on the word Simetar to see the Toolbar presented below. To test Simetar type the following command in cell A1 =NORM() press Enter and then press F9. You will see random draws of a standard normal random variable.

The installation procedure will place the word "Simetar" on the toolbar and add the Simetar icon

3

toolbar below:

Figure 5. Enter Your License Code.
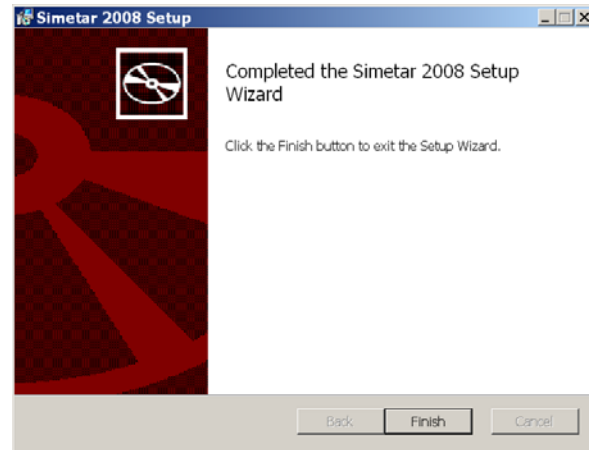
Figure 6. Final Installation Screen.

2.0 Simulating Random Variables

Simulating a stochastic model in Excel is accomplished by generating random values for each of the random variables, letting Excel update the model's equations, and saving the results of key output variables (KOVs) for statistical analysis and presentation. Repeating this process a large number of times (iterations or trials) causes the model to be simulated for a wide range of possible combinations of the random variables. The resulting array of 100 or more simulated values for a KOV defines an empirical probability distribution for each of the output variables. Probability distributions for the output variables are analyzed to gain a better understanding of the risk for the system being modeled. An example of simulation with Simetar is provided in example program Simulation Demo.xls.

2.1 Probability Distributions in Simetar

Simetar includes functions for generating pseudo-random numbers from more than 50 probability distributions plus six distributions included in Excel. An alphabetical list of probability distributions simulated by Simetar is provided in page 4. A detailed description of each Simetar function for simulating random numbers is provided in Section 3. See the Probability Distributions Demo.xls workbook for examples of how the functions are used in Excel. Access the Simetar demonstration programs from the Start menu:

Start > Programs > Simetar > Demos

Distribution	Function Name and Parameters for each Probability Distribution in Simetar
Bernoulli	=BERNOULLIDIST(ProbabilityofTrueOutcome)
Binomial	=BINOMINV(n,Prob,[USD])
Bootstrap	=BOOTSTRAPPER(ListofPossibleOutcomes,RecalculationOff)
Cauchy	=CAUCHY(Median,Sigma,[USD])
Cosine	=COSINV(Center,Radius,[USD],MaxIterations,Precision)
Correlated SND	=CSND(RangeCorrelationMatrix, [ISNDs])
Correlated USD	=CUSD(RangeCorrelationMatrix,[ISNDs],[MatrixRow],[RankCorr])
Discrete Empirical	=DEMPIRICAL(Values,[USD],[Probabilities of Values])
Double Exponential	=DEXPONINV(Mu,Sigma,[USD])
Dirichlet	=DIRICHINV(Alphas,[USD],[MatrixRow])
Empirical	=EMP(Values,Probabilities,[USD],[NormTails])
Empirical	=EMPIRICAL(Values,Probabilities,[USD],[NormTails])
Exponential	=EXPONINV(Beta,[USD])
Extreme value	=EXTVALINV(Mu,Sigma,[USD])
Geometric	=GEOMINV(Prob,[USD])
GRK	=GRK(MinValue,MidPoint,MaxValue,[USD])
GRKS	=GRKS(MinValue,MidPoint,MaxValue,[USD],[LowerSD],[UpperSD])
Hotelling T Sq	=HOTELLTINV(P,Degrees_Freedom,[UniformRandonNumber])
Hypergeometric	=HYPERGEOMINV(n,N1,S1,[USD])
Inverse Gaussian	=INVGAUS(Mu,Sigma,[USD],[MaxIterations],[Precision])
Kernal Density	=KDEINV(DataRange,BandWidth,KernelEstimator,[USD],[MaxIter],[Prec])
Logistic	=LOGISTICINV(Mu,Sigma,[USD])
Log-Log	=LOGLOGINV(Mu,Sigma,[USD])
Log-Logistic	=LOGLOGISTICINV(Alpha,Beta,[UUSD])
Modified 2 Piece Normal	=MTPNORM(MinValue,MidPoint,MaxValue,[USD],[LowSD],[UpSD])
Multinomial	=MULTINOMINV(NumTrials,Probs,[USDs])
Multivariate Empirical	=MVEMPIRICAL(RandomValuesDataMatrix,[SND],[MatrixRow])
Multivariate Log Normal	=MVLOGNORM(MeanVector,CovMatrix,[SNDs],[MatrixRow],[Moments])
Multivariate Normal	=MVNORM(MeansVector,CovarianceMatrix,[SNDs],[MatrixRow])
Multivariate Students t	=MVTINV(Student t,CovarianceMatrix,[DegreesFreefom],[SNDs],[MatrixRow])
Negative Binomial	=NEGBINOMINV(k,Prob,[USD])
Normal	=NORM(Mean,StandardDeviation,[USD])
Pareto	=PARETO(Alpha,Beta,[UniformRandonNumber])
PERT	=PERTINV(A,B,C,[USD])
Power Normal	=PNORM(Mean,StandardDeviation,P,[USD])
Poisson	=POISSONINV(Lambda,[USD])
Random Sorting	=RANDSORT(InputRangeLocation,[RecalculationOff],[DataHorizontal])
RandomWalk	=RANDWALK(Mean,StandDev,USD,Distribution,InitialVal,Coefficient)
Semicircle	=SEMICIRCDIST(X,Center,Radius,[Cumulative or Density])
Truncated Empirical	=TEMPIRICAL(RandomValues,Probabilities,MinVal,MaxVal,[USD])
Truncated Gamma	=TGAMMAINV(Alpha,Beta,AbsoluteMin,AbsoluteMax,[USD])
Truncated Normal	=TNORM(Mean,StanDev,[Min],[Max],[USD],[StackTails])
Truncated Wiebull	=TWEIBINV(Alpha,Beta,[Min],[Max],[USD])
2 Piece Normal	=TPNORM(Mean,StandardDeviation1,StandardDevviation2,[USD])
Triangle	=TRIANGLE(A,B,C,[USD])
Uniform	=UNIFORM(LowerValue,UpperValue,[USD])
Uncorrelated SNDs	=USND(CorrelationMatrixRange,CorrelatedNormalDeviatesRange)
Uncorrelated USDs	=UUSD(CorrelationMatrixRange,CorrelatedUniformDeviatesRange)
Weibull	=WEIBINV(Alpha,Beta,[USD])
Wilk's Lambda	=WILKSLINV(P,FirstDegrees of Freedom,SecondDegrees of Freedom)
Wishart	=WISHINV(CovarianceMatrix,Degrees of Freedom)

Native Excel probability distributions can be simulated in Simetar

Beta	=BETAINV(Uniform(),Alpha,Beta,Minimum,Maximum)
Chi-Squared	=CHIINV(Uniform(),Degrees of Freedom)
Gamma	=GAMMAINV(Uniform(),Alpha,Beta)
Log Normal	=LOGINV(Uniform(),Mean,StandardDeviation)
Students t	=TINV(Uniform(),Degrres of Freedom)
F	=FINV(Uniform(),Degrees of Freedom1, Degrees of Freedom2)

Simetar allows the user to specify the type of sampling procedure and the random number generator to use in generating random values. Three different random number generators are available: Mercene Twister, the Multiplicative Random Number Generator, or Excel's native generator. Two different random number sampling procedures are available : Latin hypercube and Monte Carlo. These random number generators are pseudo random and thus are suitable for conducting scenario and sensitivity analyses. The user can select the random number generator and the sampling method by selecting the General Settings Options ☒ icon and choosing the desired options in the Default User Settings menu (Figure 7).
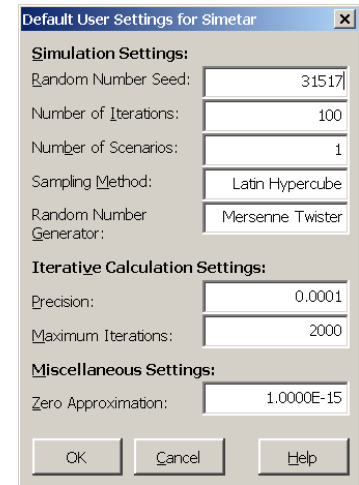
Figure 7. Setup Menu for User's Settings.

2.2 Simulation Engine in Simetar

The dialog box in Figure 8 for simulating a stochastic Excel simulation model is accessed by the ☒ icon on the Simetar toolbar. Options specified in the dialog box are saved by selecting the *Save* or *SIMULATE* buttons.

The user must specify one or more *Output Variables* (KOVs) for the statistical analysis of simulated results. The summary statistics and each simulated value (in iteration order) for each KOV are saved in the *SimData* worksheet. An output KOV can be any cell in the spreadsheet. KOVs can be cells that contain random variables, intermediate calculations, and final answers.
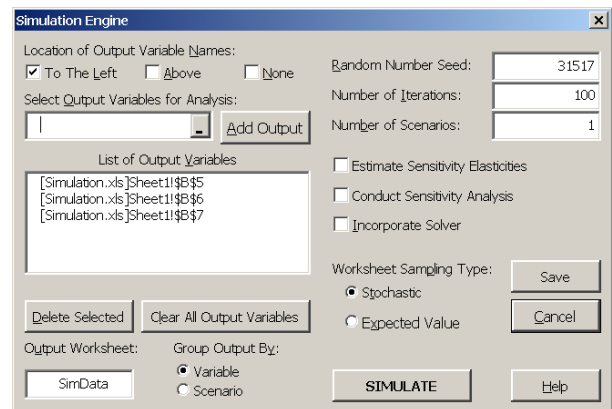
Figure 8. Simulation Dialog Box for Simetar.

Add variables to the *List of Output Variables* box by clicking in the *Select Output Variables for Analysis* window, highlighting the spreadsheet cell or cells to include, and clicking the *Add Output* box. Indicate where the variable's label is located, as in the cell *To The Left*, in the cell *Above*, or *None*. Several hundred output variables can be handled by Simetar. The sample menu in Figure 8 shows that the variables in B5, B6, and B7 are the output variables and their labels are *To The Left*. To delete an output variable or several variables, highlight the variables in the dialog box and click the *Delete Selected* button. Clicking on the *Clear All Output Variables* button will delete all of the output variables listed in the dialog box. Clicking on an output variable in the *List of Output Variables* box causes Excel to highlight the particular variable in the workbook. Simetar updates the location of KOVs in the Output Variable table if the spreadsheet is modified by adding rows or columns. Information in the Simulation Engine must be re-entered each time the workbook is opened.

After specifying the output variable(s) click the *SIMULATE* button and Simetar will simulate the workbook and save the simulated values for the output variables in the SimData worksheet or in the worksheet specified by the user. The statistics for each output variable are provided in rows 3-7 of SimData and the simulated values for each variable, by iteration, start in row 9 (Figure 9). After the 100 or more simulated values there are 10 rows of pre-programmed equations to

calculate the probability of the output variable being less than a specified target. Type in a target value in a row labeled "x_i - value" and the probability of the KOV being less than or equal to the value will appear in the next row labeled "Prob (X<=x_i)." For example, there is a 38.0 percent chance that receipts will be less than $1,300 (see column D of Figure 9).

	A	B	C	D	E
1	Simetar Simulation Results for 100 Iterations				© 2005.
2	Variable	Sheet1!B4	Sheet1!B5	Sheet1!B6	
3	Mean	10.00694	149.9765	1508.8	
4	StDev	3.027181	29.02188	561.451	
5	CV	30.25081	19.35096	37.21177	
6	Min	2.434353	100.4193	360.1179	
7	Max	18.65196	199.1456	2871.398	
8	Iteration	Price	Production	Receipts	
9	1	10.08475	198.7316	2004.159	
10	2	8.584123	151.1181	1297.217	
11	3	10.6525	139.8259	1489.495	
12	.	.	.	.	
13	99	4.635541	196.4296	910.5577	
14	100	8.88908	195.6557	1739.199	

Figure 9. Example of Stochastic Results in the SimData Worksheet.

The simulated variables in the SimData worksheet always appear in the order they were added to the *List of Output Variables* (Figures 8 and 9). The rows of simulated values for the output variables correspond to the actual iterations as they were simulated, i.e., the iteration order is maintained across output variables in SimData. The simulated values of each iteration for all output variables are provided so the user can analyze the results using Simetar functions, (for hypothesis tests, charts for presenting simulation results, and ranking risky alternatives.

2.3 Specifying Options in the Simulation Engine

2.3.1 Variable Names. The user must specify the name for a KOV before it is added to the *List of Output Variables* box (Figure 8). The variable name will appear with the stochastic results in the SimData worksheet (Figure 9). There are three options for specifying the variable names. The first option is to use the text in the cell to the left of the KOV. The second option is to use the text in the cell above the KOV and the third option is to not specify

	A	B	C	D
4	Labels	KOVs		
5	Price	8.46	=NORM(10,3)	
6	Production	101.50	=UNIFORM(100,200)	
7	Receipts	858.79	=B5*B6	

Figure 10. Labels for Key Output Variables in Cells to the Left.

a name for the KOV. The variable name can be a concatenation of the text in the cells to the left and above the KOV cell (Figure 10). The user must specify the location of the label before adding the variable to the List of Output Variables table.

2.3.2 Random Number Seed. The user may specify the *Random Number Seed*, in place of the default seed, 31517, to insure the same starting point for the pseudo random number generator from one run to the next (Figure 8). The default seed can be changed permanently in the Default User Settings menu (Figure 7).

2.3.3 Number of Iterations. The *Number of Iterations* to simulate the spreadsheet model can be set by the user (Figure 8). The default number of iterations can be changed in the Default Users Settings menu (Figure 7).

2.3.4 Output Worksheet. Output results for a simulation are stored in the SimData worksheet of the current (or a new) workbook using the specified *Output Location* (Figure 8).

2.3.5 Scenarios. The *Number of Scenarios* defaults to 1 in the menu box (Figure 8). If your model uses the =SCENARIO( ) function to simulate multiple scenarios, enter the number of scenarios. See Section 7.0 to learn more about the Scenario feature.

2.3.6 Conduct Sensitivity Elasticities Analysis. This option causes Simetar to simulate the spreadsheet model once for the base situation and once for each variable listed in the Sensitivity Variable Input window (Figure 11). The elasticity is defined as the percentage change of the KOV to a one percent change in an exogenous variable. The larger the elasticity, the greater the sensitivity of the KOV to the exogenous variable. See Section 9.0 to learn more about this option.
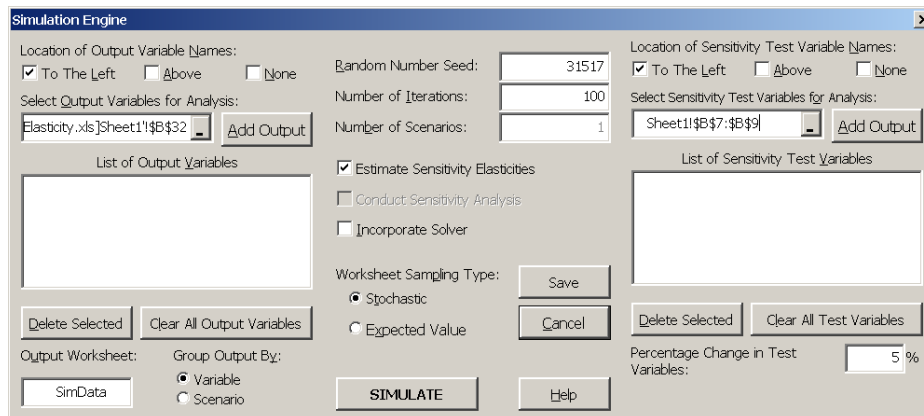
Figure 11. Simulation Menu Expanded to Estimate Sensitivity Elasticities for Variables in B6 – B8.

2.3.7 **Conduct Sensitivity Analysis.** Any Excel spreadsheet model can be simulated using the sensitivity analysis option. Numerous KOVs can be tested for percentage changes in one exogenous variable. Three percentage change levels, say, ±5%, ±10%, and ±20%, can be specified by the user. See Section 5.0 to learn more about this option. Simetar also performs sensitivities by selecting the *Conduct Sensitivity Analysis* option. (See Section 8.0 for details on simulating sensitivity analyses.)

2.3.8 Incorporate Solver. Simetar can stochastically simulate a simultaneous equation or linear programming model by selecting *Incorporate Solver*. (See Section 10.0 for details on simulating with an optimizer.)

2.3.9 Expected Value. Once stochastic variables have been incorporated into an Excel simulation model, all of the values (cells) update every time the sheet calculates or F9 is pressed. This feature in Excel is very useful for testing if stochastic variables are working correctly and if they have been linked to the proper equations in the model. However, it is also very useful to have the stochastic values fixed at their means for equation verification. Clicking the Expected Value icon sets all random variables to their means and un-clicking the icon causes Excel to calculate values for the stochastic variables. During simulation Simetar overrides the Expected Value button's setting and simulates stochastic values for all of the random variables.

2.4 User Defined Settings

The user may specify his/her preferred settings for the type of random number generator, sampling method, number of iterations, number of scenarios, random number seed, precision for MLE parameter estimation, and the maximum number of iterations for MLE and other iterative solution functions. The user defined settings are specified in the dialog box associated with the

icon (Figure 7).

3.0 Probability Distributions Simulated in Simetar

Simetar is capable of simulating univariate and multivariate random numbers from more than 50 probability distributions. Each probability distribution is described in detail in this section. Univariate probability distributions are treated first followed by multivariate probability distributions. Examples of how to simulate univariate the probability distributions are provided in Probability Distributions Demo.xls. Section numbers in the text are used to organize and identify the distributions in the demonstration workbook.

3.1. Uniform Probability Distribution

Uniformly distributed random numbers are the basis for all random numbers and are simulated by Simetar using the =UNIFORM() function. The function can be programmed three different ways:

> = UNIFORM (Min, Max, [CUSD or USD])
> = UNIFORM (B8, B9)
> = UNIFORM ( )

where: Min is the minimum value for the distribution or a cell reference,
Max is the maximum value for the distribution or a cell reference, and
CUSD is an optional input value reserved for a correlated USD (or uniform standard deviate) required for correlating non-normal distributions. See Section 3.9.2 for simulating CUSDs.

The =UNIFORM( ) function defaults to a uniform standard deviate (USD) distributed between 0 and 1 if it is programmed as =UNIFORM( ). This form of the function is an essential input in the other Simetar random number generators, particularly for simulating the native Excel probability distribution functions. Three examples of the UNIFORM function are provided below and in 3.1.1 of Probability Distribution Demo.xls.

	A	B	C	D	E	F	G
8	3.1 Generate Random Numbers Using the Uniform Distribution						
9	Minimum		10.0				
10	Maximum		20.0				
11	Unifrom Standard Deviate U(0,1)		0.924		Random No.	Formula	
12	Specific Formula using actual numbers			Uniform (10,20)	14.906	=UNIFORM(10,20)	
13	General Formula using cell references			Uniform	16.371	=UNIFORM(C9,C10)	
14	General Formula using cell references and USD			Uniform w/ USD	19.241	=UNIFORM(C9,C10,C11)	

3.2 Normal Related Probability Distributions

3.2.1 Normal. A normally distributed random number is simulated using the =NORM( ) function. The =NORM( ) function defaults to a standard normal deviate (SND) generator when no parameters are provided, as =NORM( ). A SND is a normally distributed random variable with a mean of zero and standard deviation of one. The function is programmed using one of the following forms of the command:

> = NORM (Mean, Std Dev, [USD])

= NORM (B35, B36, D13)
= NORM (B35, B36)
= NORM ( )

where: Mean is the mean of the distribution (or a cell reference, as B35),
Std Dev is the standard deviation of the distribution (or a cell reference as B36), and
USD is an optional uniform standard deviate. When a USD is not provided, Simetar generates its own uniform standard deviate. This optional variable is included so Simetar can simulate multivariate normal distributions.

	A	B	C	D	E	F	G
17	3.2.1 Generate Random Numbers Using the Normal Distribution						
18	Mean		10.000				
19	Std. Deviation		3.000				
20	Uniform Standard Deivate (USD)		0.057		Random No.	Formula	
21	Specific Formula using actual numbers			Normal		10.641	=norm(10,3)
22	General Formula using cell references			Normal		12.377	=norm(C18,C19)
23	General Formula specifying the USD			Normal		5.271	=norm(C18,C19,C20)

3.2.2 Truncated Normal. A truncated normal distribution uses the =TNORM() function. The function is programmed as follows:

= TNORM (Mean, Std Dev, [Min], [Max], [USD])
= TNORM (B47, B48, B49, B50, D13)

where: Mean is the mean for the distribution entered as a number or stored in a cell as B47,
Std Dev is the standard deviation for the distribution as B48,
Min is the absolute minimum value as B49 and is optional,
Max is the absolute maximum value as B50 and is optional, and
USD is the optional uniform standard deviate generated by =UNIFORM( ).

To simulate a truncated normal with a truncated minimum, use the function as:

= TNORM (Mean, Std Dev, Min,   , [USD])

To simulate a truncated normal distribution with a truncated maximum, use the function as:

= TNORM (Mean, Std Dev, , Max, [USD])

	A	B	C	D	E	F	G
37	Truncated Normal -- Truncate the Minimum only						
38	Mean		10.000				
39	Std Deviation		3.000				
40	Minimum		5.000		Random No.	Formula	
41	General Formula using cell references			Tnorm Min		8.197	=Tnorm(C38,C39,C40)
42							
43	Truncated Normal -- Truncate the Maximum only						
44	Mean		10.000				
45	Std Deviation		3.000				
46	Maximum		12.500		Random No.	Formula	
47	General Formula using cell references			Tnorm Max		11.615	=Tnorm(C44,C45,,C46)

3.2.3 Two-Piece Normal. The two-piece normal distribution combines half of the densities for two normal distributions with the same mean and possibly different standard deviations. The distribution is simulated as:

=TPNORM(Mean, SD Lower, SD Upper, [USD]

where: Mean is the middle value for the distribution,
SD Lower is the standard deviation for distribution less than the Mean,
SD Upper is the standard deviation for distribution greater than the Mean, and
USD is an optional uniform standard deviate.

3.2.4 Modified Two-Piece Normal. The two piece normal distribution is fully defined by specifying the minimum, middle point, the maximum and the standard deviations for the two sides. The =MTPNORM( ) is specified as:

=MTPNORM(Min, Mid, Max, [USD], [Lower SD], [Upper SD])

where: Min is the minimum value for the random variable on the number scale (default -1),
Mid is the middle value for the random variable (default 0),
Max is the maximum value for the random variable on the number scale (default 1),
USD is an optional uniform standard deviate,
Lower SD is the number of standard deviations in the lower tail (default of 2 means the minimum value is two standard deviations below the middle value), and
Upper SD is the number of standard deviations in the upper tail (default of 2 means the maximum value is two standard deviations above the middle value).

3.2.5 Student's-t (Excel's). The student's t-distribution is native to Excel but can be simulated using Simetar by providing a USD generated by =UNIFORM(). The probability distribution is simulated as:

=TINV (USD, Degrees of Freedom)

where: USD is a uniform standard deviate generated by =UNIFORM( ), and
Degrees of Freedom is self explanatory.

	A	B	C	D	E	F
68	3.2.5 Generate Random Numbers Using the Student's t (Excel) Distribution					
69	Uniform Standard Deviate		0.213			
70	df Degrees of Freedom		30.000		Random Nos.	Formulas
71	Student's t-distribution with cell references			Student's t	1.272	=TINV(C69,C70)

3.2.6 F (Excel's). The F distribution, an Excel function, is simulated as:

=FINV (USD, Degrees of Freedom1, Degrees of Freedom 2)

where: USD is a uniform standard deviate generated by =UNIFORM( ), and
Degrees of Freedom 1 and 2 are self explanatory.

3.2.7 Chi-Squared (Excel's). The chi-squared distribution, an Excel function, is simulated as:

=CHIINV (USD, Mean)

where: USD is a uniform standard deviate generated by =UNIFORM ( ), and
Mean is the average for the distribution.

3.2.8 Log Normal (Excel's). The log normal distribution, an Excel function, is used to simulate quantities like a normal distribution. The distribution is simulated as:

=LOGINV (USD, Mean, Std Dev)

where: USD is a uniform standard deviate generated by =UNIFORM ( ),
Mean is the average, and
Std Dev is the standard deviation for the distribution.

The simulated values from =LOGINV( ) are in "natural log" form so take the anti-log of the stochastic values using the Excel function =LN( ).

3.2.9 Power Normal. The power normal distribution is simulated in Simetar using the =PNORM( ) function as:

=PNORM(Mean, Sigma, Exp P, [USD])

where: Mean is a real number and indicates the central tendency parameter for the distribution,
Sigma is a number greater than zero that represents the variance for the distribution,
Exp P is a value greater than zero, the exponent parameter for the distribution, and
USD is an optional uniform standard deviate.

3.2.10 Inverse Gaussian. The inverse Gaussian distribution is simulated using an iterative solution procedure. The =INVGAUS( ) function is programmed as:

=INVGAUS (Mu, Sigma, [USD], [Max Iter], [Precision])

where: Mu is a positive real number representing the first parameter of the distribution,
Sigma is a number greater than zero that indicates the shape parameter for the distribution,
USD is an optional uniform standard deviate,
Max Iter is an optional maximum iterations used to find the stochastic value, and
Precision is an optional term to specify the precision of the answer.

3.3 Continuous Probability Distributions

3.3.1 Gamma (Excel's). The gamma distribution, an Excel function, can be used to simulate the length of time to complete a task. The distribution is specified as:

=GAMMAINV (USD, Alpha, Beta)

where: USD is a uniform standard deviate generated by =UNIFORM ( ),
Alpha is the first parameter for the gamma distribution, and
Beta is the second parameter for the gamma distribution.

3.3.2 Truncated Gamma. The gamma distribution can be truncated at the lower or upper end
with the =TGAMMAINV( ) function. The function is used as:

=TGAMMAINV (Alpha, Beta, [Min], [Max], [USD])

where: Alpha is the first parameter for the gamma distribution and must be greater than zero,
Beta is the second parameter for the gamma distribution and must be greater than zero,
Min is the optional value for the absolute minimum (0 < min < max),
Max is the optional value for the absolute maximum (min < max < ∞), and
USD is an optional uniform standard deviate.

3.3.3 Exponential. The exponential distribution can be used to simulate times between
independent events that occur at a constant rate, such as arrivals at a service center. The
distribution is simulated as:

= EXPONINV (Beta, [USD])

where: Beta is the only parameter for the exponential distribution, and
USD is an optional uniform standard deviate.

3.3.4 Double Exponential. The double exponential distribution can be used to simulate times
between independent events that occur at a constant rate, such as arrivals at a service center. The
distribution is simulated as:

= DEXPONINV (Beta, [USD])

where: Beta is the only parameter for the double exponential distribution, and
USD is an optional uniform standard deviate generated by =UNIFORM( ).

3.3.5 Weibull. The Weibull distribution is often used to simulate reliability or lifetimes for
machinery. The distribution is simulated as:

= WEIBINV (Alpha, Beta, [USD])

where: Alpha is the first parameter for the Weibull distribution and must be greater than zero,
Beta, the second parameter for the Weibull distribution, must be greater than zero, and
USD is an optional uniform standard deviate generated by =UNIFORM( ).

3.3.6 Truncated Weibull. The Weibull distribution can be simulated with a finite minimum
and/or maximum as:

=TWEIBINV (Alpha, Beta, [Min], [Max], [USD])

where: Alpha is the first parameter of the Weibull distribution and must be greater than zero,

Beta is the second parameter of the Weibull distribution and must be greater than zero,
Min is the absolute minimum ($0 <$ min $<$ max),
Max is the absolute maximum (min $<$ max $< \infty$), and
USD is an optional uniform standard deviate.

3.3.7 Cauchy. The Cauchy distribution is a symmetrical distribution about its parameter theta (θ). If median and sigma parameters are not provided the function defaults to a =CAUCHY(0,1) random variable. The distribution can be simulated in Simetar as:

=CAUCHY ([Median], [Sigma], [USD])

where: Median is an optional value for the mid point of the distribution,
Sigma is an optional term to indicate the shape of the distribution, and
USD is an optional uniform standard deviate.

3.3.8 Logistic. A logistic distribution can be simulated using the =LOGISTICINV() function as:

=LOGISTICINV(Mu, Sigma, [USD])

where: Mu is the first parameter for the logistic distribution and it must be a real value,
Sigma is the second parameter for the distribution and must be greater than zero, and
USD is an optional uniform standard deviate.

3.3.9 Log-Log. The log-log distribution is simulated as:

=LOGLOGINV (Mu, Sigma, [USD])

where: Mu is any real value indicating the position of the distribution on the number scale,
Sigma is a value greater than zero indicating the scale parameter, and
USD is an optional uniform standard deviate.

3.3.10 Log-Logistic. The log-logistic distribution is simulated as:

=LOGLOGISTICINV (Alpha, Beta, [USD])

where: Alpha is a value greater than zero which represents the shape parameter,
Beta is the scale parameter and must be greater than zero, and
USD is an optional uniform standard deviate.

3.3.11 Extreme Value. An extreme value distribution can be simulated as:

=EXTVALINV (Mu, Sigma, [USD])

where: Mu is the real value indicating the location parameter for the extreme value distribution.
Sigma any value greater than zero indicating the scale parameter of the distribution, and
USD is an optional uniform standard deviate generated by =UNIFORM( ).

3.3.12 Pareto. A Pareto distribution can be simulated using the =PARETO() function as:

=PARETO(Alpha, Beta, [USD])

where: Alpha is the first parameter for a Pareto distribution and it must be greater than zero,
Beta is the second parameter for the distribution and it must be greater than zero, and
USD is an optional uniform standard deviate.

3.4 Finite-Range Continuous Probability Distributions

3.4.1 Triangle. The triangle distribution is defined by the minimum, mode, and maximum. The distribution can be simulated as:

=TRIANGLE (Min, Mode, Max, [USD])
=TRIANGLE (A95, A96, A97)

where: Min is the minimum for the distribution,
Mode is the mode for the distribution,
Max is the maximum for the distribution, and
USD is an optional uniform standard deviate.

	A	B	C	D	E	F	G
205	3.4.1 Generate Random Numbers Using the Traingle Distribution						
206	Minimum		12.000				
207	Mode		18.000				
208	Maximum		27.000				
209	Uniform Standard Deviate		0.925		Random Nos.	Formulas	
210	Triangle Distribution			Triangle	23.134	=triangle(C206,C207,C208)	
211	Genearalized triangle			Triangle	23.823	=triangle(C206,C207,C208,C209)	

3.4.2 Beta (Excel's). The beta distribution, an Excel function, can be used to simulate the proportion of defective items in a shipment or time to complete a task. The distribution is simulated as:

=BETAINV (USD, Alpha, Beta, [Min], [Max])
=BETAINV (UNIFORM ( ), Alpha, Beta)

where: USD is a uniform standard deviate generated by =UNIFORM ( ),
Alpha is the first parameter for the distribution,
Beta is the second parameter for a beta,
Min is an optional value for truncating the minimum of the distribution, and
Max is an optional value for truncating the maximum of the distribution.

3.4.3 PERT Distribution. A PERT distribution can be simulated by Simetar using the =PERTINV() function as:

=PERTINV(Min, Middle, Max, [USD])

where: Min is a lower bound parameter,
Middle is a middle parameter with min < middle < max,

Max is an upper bound parameter, and
USD is an optional uniform standard deviate.

3.4.4 Cosine. The cosine distribution is simulated by Simetar using an iterative solution procedure. The =COSINV( ) function is programmed as:

=COSINV(Center, Radius, [USD], [Max Iter], [Precision])

where: Center is a real number that represents the first parameter for a cosine distribution,
Radius is a positive value that represents the second parameter,
USD is an optional uniform standard deviate,
Max Iter is the maximum number of iterations used to find the stochastic value, and
Precision is an optional term to specify the precision of the answer.

3.4.5 Semicircle. The semicircle distribution is simulated as:

=SEMICIRCINV(Center, Radius, [USD], Max Iter, Precision)

where: Center is a real number that indicates the first parameter of the distribution,
Radius is the second parameter for the distribution and must be greater than zero,
USD is an optional uniform standard deviate,
Max Iter is the maximum number of iterations to find the value (max > 0), and
Precision is a positive value to specify how precise the optimum answer should be. If an optimum answer is not found within the precision level in the maximum number of iterations, #VALUE1 error is returned.

3.5 Analogs to Finite Range Probability Distributions

3.5.1 GRK. The GRK distribution is an empirical substitute for the triangle distribution and is similar to a two piece normal distribution. The GRK distribution simulates values less than the minimum about two percent of the time. Values greater than the maximum are observed about two percent of the time. A GRK distribution can be simulated as:

= GRK (Min, Middle, Max, [USD])
= GRK (A95, A96, A97)

where: Min is the value for the minimum,
Middle is the value for the mid point of the distribution,
Max is the value (or cell) for the maximum, and
USD is an optional uniform standard deviate.

3.5.2 GRKS. The GRKS distribution is a continuous probability distribution for sampling from a minimum data population. Given a minimum, middle value and a maximum to describe the population the =GRKS( ) function is a continuous distribution substitute for the triangle distribution. The LSD and USD parameters indicate the number of standard deviations below and above the middle value that the distribution can extend. An LSD of 2 implies the minimum is at approximately the 2nd percentile and a LSD of 3 implies sampling with a minimum at approximately the 0.5 percentile. Program =GRKS( ) as follows:

=GRKS (Min, Middle, Max, [USD], [LSD], [USD])
=GRKS (C250, C251, C252, C253, C259, C260)

where: Min is the value for the minimum,
        Middle is the value for the mid point of the distribution,
        Max is the value (or cell) for the maximum,
        USD is an optional uniform standard deviate,
        LSD optional number of standard deviations below the middle, as D97, and
        USD optional upper number of standard deviations above the maximum, as D98.

	A	B	C	D	E	F	G
248	3.5.1 Generate Random Numbers Using the GRK Distributions						
249	Parameters for the GRK Distribution						
250	Minimum		3.000				
251	Mode		5.000				
252	Maximum		16.000				
253	Uniform Standard Deviate		0.853		Random Nos.	Formulas	
254	Specific Formula using actual numbers			GRK	3.197	=grk(3,5,16,UNIFORM())	
255	General Formula using cell references			GRK	8.200	=grk(C250,C251,C252)	
256	General Formula with a USD specified			GRK	10.857	=grk(C250,C251,C252,C253)	
257	3.5.2 Generate Random Numbers Using the GRKS Distributions						
258	General formula without bounds			GRKS	4.564	=grks(3,5,16,UNIFORM())	
259	Lower no. of standard deviations		3.000	GRKS Min	4.748	=grks(3,5,16,UNIFORM(),C259)	
260	Upper no. of standard deviations		4.000	GRKS Max	7.889	=grks(C250,C251,C252,C253, ,C260)	
261				GRKS Min&Max	7.889	=grks(C250,C251,C252,C253,C259,C260)	

3.6 Discrete Probability Distributions

3.6.1 Bernoulli. A Bernoulli distribution can be used to simulate the occurrence of an event, such as a machine failure during a given time period. Simulate a Bernoulli distribution as:

        = BERNOULLI (P)
        = BERNOULLI (A10)

where: P is the probability $(0 < P < 1)$, of the variable or condition being true (or 1).

	A	B	C	D	E	F
265	3.6.1 Generate Random Numbers Using the Bernoulli Distribution					
266	Parameter for the Bernoulli Distribution					
267	Probability of a success or a 1		0.450		Random Nos.	Formulas
268	Specific Formula using actual numbers			Bernoulli	0.000	=bernoulli(0.45)
269	General Formula using cell references			Bernoulli	1.000	=bernoulli(C267)

3.6.2 Binomial. The binomial distribution is a discrete distribution for simulating the number of successes in N independent Bernoulli trials each having a probability P of success. Other applications are to simulate the number of units demanded in a given time period. Simulate the binomial distribution as:

        =BINOMINV (N, P, [USD])

where: N is the number of trials,
        P is the probability of a positive success, and
        USD is an optional uniform standard deviate.

3.6.3 Negative Binomial. The negative binomial distribution simulates the number of failures before the Nth success in a sequence of independent Bernoulli trials each having a probability P of success. Simulate the negative binomial distribution as:

=NEGBINOMINV (N, P, [USD])

where: N is a positive integer representing the number of failures before the next success,
P is the probability of success, and
USD is an optional uniform standard deviate.

3.6.4 Multinomial. The multinomial probability distribution returns either an array of values or a scalar, depending upon how it is used. If the probabilities (Probs) are entered as an array the function returns an array, but if Probs is a scalar it returns a scalar. An example of the multinomial distribution in Step 3.6.4 of Probability Distributions Demo.xls demonstrates how the function can be used both ways.

=MULTINOMINV(No. Trials, Probs, [USD])

where: No. Trials is the sample size (integer greater than zero) used in the distribution,
Probs is a vector of cell probabilities associated with each cell's random variable.
Individual values are between zero and one and must sum to one. If a single value is entered for Probs the function returns a binomial random variable.
USD is an optional univariate standard deviation.

3.6.5 Poisson. The Poisson distribution simulates the number of events that occur in an interval of time, such as arrivals at a service point. The distribution can also be used to simulate random quantities demanded during an interval of time. Simulate the Poisson distribution as:

=POISSONINV (L, [USD])

where: L, the only parameter for a Poisson, must be positive and is generally an integer, and
USD is an optional uniform standard deviate.

3.6.6 Geometric. The geometric distribution simulates the number of failures before the first success in a sequence of independent Bernoulli trials each with a P probability. Also this distribution can simulate the number of items demanded in a given period. The geometric distribution is simulated as:

=GEOMINV (P, [USD])

where: P is the probability of each independent Bernoulli trial, and
USD is an optional uniform standard deviate.

3.6.7 Hypergeometric. The Hypergeometric distribution is used to simulate the number of units that are acceptable in a sample of size K taken from a population of size N when it is known that M of the units in the population are acceptable. This is a sample without replacement problem made famous by the urn with N balls, m of which are green, (N-M are red), and a sample of K

	A	B	C	D	E	F	G
321	3.7.1 Generate Random Numbers Using the Empirical Distribution						
322	Parameters for an Empirical Distribution						
323							
324	P(x)	Price					
325	0	1.5030					
326	0.038462	1.5031					
327	0.115385	1.9440					
328	0.192308	2.0112					
329	0.269231	2.0743					
330	0.346154	2.2647					
331	0.423077	2.2847		Uniform Std Deviate			
332	0.500000	2.3649		0.834			
333	0.576923	2.3749					
334	0.653846	2.4551			Random Nos.	Formulas	
335	0.730769	2.5052		Four Forms of the Empirical Distribution using a Common USD			
336	0.807692	2.5453		Empirical 1	2.604	=empirical(B325:B339,A325:A339,D332)	
337	0.884615	2.7156		Empirical 2	2.604	=emp(B325:B339,A325:A339,D332)	
338	0.961538	3.2467		Empirical 3	2.604	=emp(B325:B339,,D332)	
339	1	3.2468		EMP Norm Tail	2.604	=emp(B325:B339,A325:A339,D332,1)	

balls are drawn. The Hypergeometric function returns the number of red balls in the sample of K. Simulate the Hypergeometric distribution as:

=HYPERGEOMINV (N, M, K, [USD])

where: N is the population size,
        M is the number of units in the population with the desired characteristic,
        K is the sample size, and
        USD is an optional uniform standard deviate

3.7 Sample Based Probability Distributions

3.7.1 Empirical. An empirical distribution can be simulated by Simetar using the =EMPIRICAL( ) or the =EMP() function. The function assumes a continuous distribution so it interpolates between the specified points on the distribution (S_i) using the cumulative distribution probabilities ($F(S_i)$). The most direct form of the function is =EMPIRICAL(S_i) or =EMP(S_i) which causes Simetar to calculate the $F(S_i)$ and USD values for the distribution. The function is programmed as follows:

= EMPIRICAL(S_i, $F(S_i)$, [USD], [Normal Tails])
= EMP(B75:B89, A75:A89, D13)

where: S_i represents an array of N sorted random values including the min and max,
        $F(S_i)$ cumulative probabilities for the S_i values, including the end points of zero and one,
        USD is an optional uniform standard deviate generated by =UNIFORM(), and
        Normal tails is an optional term to extend the tails of the distribution beyond the end of the data (enter a 1) or to truncate the distribution with the default value of 0.

Note: i = 1 to n for the S_i and $F(S_i)$ parameters denotes that these are ranges and not individual values.

3.7.2 Truncated Empirical. A truncated empirical distribution is the same as an empirical distribution but with a defined minimum and maximum. The distribution is simulated as:

=TEMPIRICAL((S_i, $F(S_i)$), Min, Max, [USD])

where: S_i represents an array of N sorted random values including the min and max,
$F(S_i)$ cumulative probabilities for the S_i values, including the end points of zero and one,
USD is an optional uniform standard deviate generated by =UNIFORM(),
Min is the minimum for the distribution, and
Max is the maximum for the distribution.

3.7.3 Discrete Empirical. When it is not appropriate to interpolate between the S_i points on the empirical distribution, then the data are said to be distributed discrete empirical. This distribution is applicable if the data can only take on set values. Each value is assumed to have an equal chance of being selected. The function is programmed in Simetar as follows:

=DEMPIRICAL (S_i, [USD])
=DEMPIRICAL (B75:B89, D13)

where: S_i represents an array of n random values; the values do not have to be sorted, and
USD is an optional uniform standard deviate.

	A	B	C	D	E	F	G
347	3.7.3 Generate Random Numbers Using the Discrete Empirical Distribution						
348	Sorted Values for a Discrete Random Variable						
349	1						
350	5	USD	0.3009				
351	8				Random Nos.	Formulas	
352	9			Discrete Emp	9.000	=DEMPIRICAL(A349:A353)	
353	11			Discrete Emp	5.000	=DEMPIRICAL(A349:A353,C350)	

3.7.4 Kernel Density Estimated Random Variable. The =KDEINV( ) function uses Parzen type kernel density estimators to evaluate a smoothed value that represents a point on a cumulative distribution function (CDF). Eleven alternative kernel density estimators can be used to smooth an empirical distribution and simulate random values in Simetar. A graphical representation of the kernel density smoothed function can be developed using the smoothing option in the CDF chart tool (see Section 6.2 for the CDF chart function). The kernel density estimated random variable function is simulated as:

=KDEINV(Data Range,[BW], [KE], [USD], [Max Iter], [Precision])

where: Data range is the location for data series for the empirical distribution to simulate,
BW is an optional bandwidth to use in estimating the influence of each data point on the CDF estimation,
KE is an optional term to specify the kernel estimation type used to estimate the CDF. The KE types are: Gaussian (0 or 1), Uniform (2), Casinus (3), Triangle (4), Triweight (5), Epanechnikow (6), Quartic (7), Cauchy (8), Double Exponential (9), Histogram (10), and Parzen (11),
USD is an optional uniform standard deviate,
Max Iter is the maximum number of iterations to use to find the result, and

Precision is an optional term to specify how precise the final solution should be. If an optimal answer is not found #VALUE! will appear in the cell.

	A	B	C	D	E	F	G
353	3.7.4 Generate Random Numbers Using the Kernel Density Estimator Distribution						
354	Unsorted Data Range						
355	1.500	Bandwidth	0.1163	=BANDWIDTH(A355:A367)			
356	1.940	Kernel Estimator	0	"0" indicated Gaussian			
357	2.540	USD	0.0148				
358	2.360	Max Iterations	1000				
359	2.280	Precision	0.00001				
360	2.370						
361	2.070				Random Nos.	Formulas	
362	2.500		Kernel Density E	1.398670281	=KDEINV(A355:A367,C355,C356,C357,C358,C359)		
363	2.260						
364	3.240						
365	2.710						
366	2.450						
367	2.507						

3.7.5 Discrete Uniform. A discrete uniform random variable can take on only certain values, each with an equal probability. For example, a fair die can take on one of six values (1, 2, 3, 4, 5, 6) with an equal probability. To simulate a discrete uniform random variable use the =RANDSORT( ) function. For example, if the random values to define a distribution are 1, 2, 3, 4, 5, 6, and are stored in cells A1:A6, simulate a random value, by typing the following command in a cell:

=RANDSORT(A1:A6)

3.7.6 Random Sorting. The array form of the =RANDSORT( ) function can be used to simulate (sample) random draws of a list of names or objects or numbers without replacement. For example, if five names Jim, Joe, Sam, John, and Bill are to be randomly sorted (shuffled), enter the names in an array and use =RANDSORT( ) as an array function. Assume the five names are in A1:A5 and the random sample is to appear in B1:B5; type the following command in B1 after highlighting array B1:B5:

=RANDSORT(A1:A5)

Press *Control Shift Enter*, rather than Enter after typing the function, because this is the array form of the function. Press F9 to "resort" the data for a second iteration or sample.

	A	B	C	D	E	F
382	3.7.6 Random Sorting a List of Names Using the Array Function form of RANDSORT					
383	List of Names to be Sampled				Random Nos.	Formula
384	Ace		Random Shuffle of Objects		Jack	=randsort(A384:A387)
385	King		Using the Array Function form:		Ace	
386	Queen				King	
387	Jack				Queen	
388			Draw a single name at random		Jack	=randsort(A384:A387)

3.7.7 Bootstrapping (Random Sampling with Replacement). Bootstrap sampling techniques are used for advanced simulation problems and assume that past deviates or errors can be re-sampled an infinite number of times. This method of sampling can be accomplished using the =BOOTSTRAPPER( ) function which samples from a known distribution with replacement. An example of the function is provided below and in Step 3.7.7 in Probability Distributions Demo.xls.

=BOOTSTRAPPER (Array of Random Values, [Preserve Rows])
=BOOTSTRAPPER (A27:A31, 1)

where: Array of random values is the location for the array of random values to be sampled during simulation, and

Preserve Rows the optional term to retain the order of the values in rows, if the array of random variables is a matrix.

	A	B	C	D	E	F	G	H	I	J
392	3.7.7 Generate Random Numbers Using the Bootstrap Simulation									
393	Random Values for X				Random Nos.	Formulas				
394	1			Bootstrap		1	=BOOTSTRAPPER($A$394:$A$399)			
395	2			Bootstrap		5	=BOOTSTRAPPER($A$394:$A$399)			
396	3			Bootstrap		5	=BOOTSTRAPPER($A$394:$A$399)			
397	4			Bootstrap		6	=BOOTSTRAPPER($A$394:$A$399)			
398	5			Bootstrap		6	=BOOTSTRAPPER($A$394:$A$399)			
399	6			Bootstrap		4	=BOOTSTRAPPER($A$394:$A$399)			
400										
401	Bootstrap Simulation with a Matrix of Observations: Preserve the values across the row.									
402					Random Bootstrap Nos. Maintaining the Rows					
403	X1	X2	X3		X1	X2	X3	Formula		
404	1.0	23.4	4.5		5.0	55.5	1.5	=BOOTSTRAPPER($A$404:$C$408,1)		
405	4.0	56.7	3.3		4.0	56.7	3.3	=BOOTSTRAPPER($A$404:$C$408,1)		
406	8.0	12.9	6.9		1.0	23.4	4.5	=BOOTSTRAPPER($A$404:$C$408,1)		
407	9.0	6.4	2.3		5.0	55.5	1.5	=BOOTSTRAPPER($A$404:$C$408,1)		
408	5.0	55.5	1.5		4.0	56.7	3.3	=BOOTSTRAPPER($A$404:$C$408,1)		

3.8 Time Series Probability Distributions

3.8.1 Random Walk. The =RANDWALK( ) function generates a random variable that is characteristic of a random walk. A random walk distribution for x_t is characterized as $X_t = X_{t-1} + \tilde{e}_t$ where $\tilde{e}_t$ is normally distributed. Simulating a variable for N iterations will result in a sample of length N. The function is used as:

=RANDWALK (Mean, Std Dev, [USD], [Distribution], [Initial Value], [Coefficient])

where: Mean is the expected value for the random variable,
Std Dev is the standard deviation for the variable and is greater than zero,
USD is an optional uniform standard deviate,
Distribution is an optional code for the distribution for generating random changes as: normal (0 or 1), uniform (2), cosine (3), Cauchy (8), double exponential (9), logistic (12), extreme value (13), exponential (14), and log normal (15),
Initial Value is an optional initial value to start the random sequence; the default is zero,
Coefficient is an optional value on the lag variable as α in $X_t = \alpha X_{t-1} + \tilde{e}_t$.

	A	B	C	D	E	F	G	H
412	3.8 Generate Random Numbers Using the Random Walk							
413	Mean		10					
414	Standard Deviation		3					
415	USD		0.099617958					
416	Distribution	Gaussian	0					
417	Initial Value		4.5					
418	Coefficient		0		Random Nos.	Formulas		
419	Random Walk sequence			RandomWalk	6.148805478	=randwalk(C413,C414,C415,C416,C417,C418)		

3.9 Multivariate Distributions

3.9.1 Correlated Standard Normal Deviates. Correlated standard normal deviates (CSND's) are generated in Simetar using the =CSND( ) function. An array of correlated standard normal deviates can be used to simulate multivariate normal (MVN) probability distributions in a two step procedure. An array of CSNDs is simulated as:

        = CSND (Correlation Matrix Range, [Optional Range of Independent SNDs])
        = CSND (B152:G157)
        = CSND (B152:G157, B161:B166)

where: Correlation Matrix Range specifies the location of a non-singular NxN correlation matrix. Calculate the correlation matrix using the function described in Section 12.3. Optional Range of Independent SND's (ISND's) is an Nx1 array of SND's generated using =NORM( ) in N cells.

As an array function =CSND( ) <u>must</u> be used as follows: highlight the output location for N cells and type the command =CSND (correlation matrix location, optional range of ISNDs) and press the *Control Shift Enter* keys.

	A	B	C	D	E	F	
422	3.9.1 Generate Correlated Standard Normal Deviates						
423	Correlation Matrix for 3 Random Variables						
424		Price 1	Price 2	Price 3			
425	Price 1	1	0.868849998	0.980957963			
426	Price 2	0	1	0.815781325			
427	Price 3	0	0	1			
428	CSND is an array function so highlight a 3x1 array for the three variables						
429					Random Nos.	Formulas	
430				Price 1	1.37071	=csnd(B425:D427)	
431				Price 2	1.05759		
432				Price 3	1.22886		

3.9.2 Correlated Uniform Standard Deviates. Correlated uniform standard deviates (CUSDs) are used to simulate multivariate non-normal (e.g., empirical) probability distributions in a two step process. An array of CUSDs is simulated as:

        =CUSD (Correlation Matrix Range, [Optional Range of Independent SNDs])
        =CUSD (B152:G157)
        =CUSD (B152:G157, B161:B166)

where: Correlation Matrix Range specifies location of a non-singular NxN correlation matrix. Calculate the correlation matrix using the function described in Section 12.3. Optional Range of Independent SNDs is an Nx1 array of SNDs generated using =NORM( ) in N cells.

As an array function =CUSD( ) <u>must</u> be used as follows: highlight the output location for N cells and type the command =CSND (correlation matrix location, optional range of ISNDs), and press the *Control Shift Enter* keys.

	A	B	C	D	E	F	
435	3.9.2 Generate Correlated Uniform Standard Deviates						
436	Correlation Matrix for 3 Random Variables						
437		Price 1	Price 2	Price 3			
438	Price 1	1	0.868849998	0.980957963			
439	Price 2	0	1	0.815781325			
440	Price 3	0	0	1			
441	CUSD is an array function so highlight a 3x1 array for the three variables						
442					Random Nos.	Formulas	
443				Price 1	0.53728	=cusd(B438:D440)	
444				Price 2	0.79301		
445				Price 3	0.46306		

3.9.3 Multivariate Normal (MVN) Distribution in One Step

Simetar provides a one step function for simulating a MVN distribution. The =MVNORM( ) function uses an Nx1 array of means and an NxN covariance matrix to generate correlated random values that are distributed multivariate normal. The array function is entered as follows:

=MVNORM (Means Vector, Covariance Matrix, [Array of ISNDs])

where: Means Vector is an Nx1 array of the averages to use for simulating MVN values, and Covariance Matrix is an NxN covariance matrix for the N random variables. Array of ISNDs is an optional Nx1 array of independent standard normal deviates generated with n cells of =NORM( ).

To use the array function, first highlight the number of cells equal to the number of means at the output location, second type =MVNORM (location for the array of means, location of covariance matrix), and press the *Control Shift Enter* keys.

	A	B	C	D	E	F	G
448	3.9.3 Multivariate Normal Distribution in One Step						
449	Covariance Matrix						
450		Price 1	Price 2	Price 3	Prod 1	Prod 2	Prod 3
451	Price 1	100.000	173.770	147.144	-8.980	-43.833	-190.314
452	Price 2		400.000	244.734	-9.523	-183.525	-246.381
453	Price 3			225.000	-12.938	-63.369	-304.910
454	Prod 1				9.000	39.829	63.377
455	Prod 2					625.000	425.693
456	Prod 3						1296.000
457				Array Function for			
458		Assumed Means		Correlated MVN Values		Formulas for random numbers	
459	Price 1	100.0		98.333		=mvnorm(B459:B464,B451:G456)	
460	Price 2	200.0		185.444		=mvnorm(B459:B464,B451:G456)	
461	Price 3	250.0		246.706		=mvnorm(B459:B464,B451:G456)	
462	Prod 1	25.0		22.737		=mvnorm(B459:B464,B451:G456)	
463	Prod 2	190.0		211.980		=mvnorm(B459:B464,B451:G456)	
464	Prod 3	260.0		252.160		=mvnorm(B459:B464,B451:G456)	

3.9.4 Multivariate Normal Distribution in Two Steps.
A general formula for simulating a multivariate normal distribution is accomplished by first generating a vector of CSNDs and then using the CSNDs in the formula for a normal distribution. In step 1 an Nx1 array of CSNDs is generated using =CSND( ), see Section 3.9.1. The example provided here is for a three variable model so N equals 3. Assume the non-singular covariance matrix is in A1:C3, the three means

are in cells B7:B9, and the three standard deviations are in cells C7:C9.

Step 1:	In A4:A6	= CSND (A1:C3)
Step 2:	In A7	= B7 + C7 * A4
	In A8	= B8 + C8 * A5
	In A9	= B9 + C9 * A6

These three Excel statements can be repeated for N variables. The three random variables will be appropriately correlated within each period but will be independent across periods.

An alternative two step procedure for simulating a multivariate normal distribution uses a vector of CUSDs. In step 1 an Nx1 array of CUSDs is generated using =CUSD( ), see Section 3.9.2. In step 2 use the =NORM( ) function to simulate the random values. The example provided here is for a three variable model so N equals 3. Assume the non-singular correlation matrix is in A1:C3, the three means are in cells B7:B9, and the three standard deviations are in cells C7:C9.

Step 1:	In A4:A6	= CUSD (A1:C3)
Step 2:	In A7	= NORM(B7, C7, A4)
	In A8	= NORM(B8, C8, A5)
	In A9	= NORM(B9, C9, A6)

These three Simetar statements can be repeated for N variables. The three random variables will be appropriately correlated within each period but will be independent across periods.

3.9.5 Multivariate Empirical (MVE) in One Step

Simetar provides a one step function for simulating a MVE distribution. The =MVEMPIRICAL( ) function uses as input the MxN matrix of the M observations for the N random variables. The result is an Nx1 array of MVE correlated random values for the N variables. Program the function as:

      =MVEMPIRICAL (Range for Random Variables ,,,, [Vector of Means], [Type])

where: Range for Random Variables is an MxN matrix of the M observed values for the N random variables,
Vector of Means is an array of forecasted means for the N random variables, and
Type is a option code for the type data transformation used to generate the forecasted means for the MVE: (0) for actual data, (1) for percent deviations from mean, (2) for percent deviations from trend, and (3) is for differences from the mean.

The =MVEMPIRICAL( ) function is an array function so highlight an Nx1 array at the output location and type the function, followed by pressing the *Control Shift Enter* keys. An example of using the one step =MVEMPIRICAL( ) function is provided below for a MVE distribution with 6 (N) variables and 13 (M) observations.

3.9.6 Multivariate Empirical (MVE) Distribution in Two Steps

Multivariate empirical distributions can be simulated in two steps using the =EMP( ) function and an array of correlated uniform standard deviates or CUSDs generated using the =CUSD( ) function described in Section 3.9.2. An example of the two step MVE is provided for a three variable model, assuming the correlation matrix is in H1:J3, the forecasted means are in cells A1:A3, the fractional deviations (S_i) from the mean are in cells C1:E12, and the three variables the probabilities for the deviates $(F(S_i))$ are in B1:B12.

Step 1:   In A14:A16   = CUSD (H1:J3)

Step 2:   In C14   = A1 + A1 * EMP(C1:C12, B1:B12, A14))
           In C15   = A2 + A2 * EMP (D1:D12, B1:B12, A15))
           In C16   = A3 + A3 * EMP (E1:E12, B1:B12, A16))

The values in cells C14:C16 are appropriately correlated based on the correlation matrix in cells H1:J3 and are distributed empirical about the respective forecasted means in cells A1:A3. The formulas in cells A14:A16 and C14:C16 can be repeated for as many periods (years) as the model simulates. Simulated MVE values will be correlated within each period but will be independent across periods.

	A	B	C	D	E	F	G
496	3.9.5 Multivariate Empirical Distribution in One Step						
497	Historical Data for the Random Variables						
498		Var 1	Var 2	Var 3	Var 4	Var 5	Var 6
499	1	1.500	2.420	1.370	119.375	34.428	67.744
500	2	1.940	2.570	1.700	119.844	37.674	69.396
501	3	2.540	3.720	2.270	84.613	34.070	63.779
502	4	2.360	3.720	2.100	116.165	32.748	55.428
503	5	2.280	2.610	2.120	118.503	39.498	63.077
504	6	2.370	3.000	2.251	108.581	34.333	59.256
505	7	2.070	3.240	1.873	131.480	39.305	72.616
506	8	2.500	3.260	2.310	100.705	38.213	59.912
507	9	2.260	3.450	2.128	138.608	37.575	72.805
508	10	3.240	4.550	3.190	113.453	35.812	55.614
509	11	2.710	4.300	2.340	127.051	36.314	67.471
510	12	2.450	3.380	2.200	127.043	39.740	69.546
511	13	2.007	2.605	1.815	133.300	43.265	66.469
512							
513	Variables	One Step MVE	Formulas used for the random numbers			Vector of Means	
514	0.115384616	89.186	=MVEMP(B499:G511,,,,F514:F519,1)			100.000	
515	0.192307696	154.426				200.000	
516	0.269230783	246.795				250.000	
517	0.346153855	27.924				25.000	
518	0.423076928	218.821				190.000	
519	0.5	285.841				260.000	

3.9.7 Multivariate Mixed Distribution

Simetar can simulate a multivariate mixed distribution (MVM) which has correlated variables that are distributed differently. For example a MVM could include variables that are distributed uniform, empirical, normal, and beta. To simulate a MVM, use the =CUSD( ) function to simulate an Nx1 vector of correlated uniform standard deviates, one CUSD for each variable. Use each of the CUSDs in the appropriate Simetar function to simulate the random variables. Using an example of a four variable MVM with the variables distributed uniform, empirical, normal and beta, respectively, use the following functions:

Step 1:   =CUSD(Correlation Matrix Range)

Step 2: =UNIFORM(Min, Max, $CUSD_1$)
=EMP(S_i, $F(S_i)$, $CUSD_2$)
=NORM(Mean, Std, Dev, $CUSD_3$)
=BETAINV($CUSD_4$, Alpha, Beta, [Min], [Max])

where: $CUSD_i$ values refer to the ith correlated uniform standard deviate simulated in the Nx1 CUSD array.

The simulated random variables will be appropriately correlated based on the correlation matrix.

3.9.8 Multivariate Log Normal. A log normally distributed series of random variables can be simulated multivariate using the =MVLOGNORM( ) array function. The function is used as:

=MVLOGNORM (Mean Vector, Covariance, [Array of ISNDs], [Matrix Row], [Moments])
=MVLOGNORM (A1:A4, B1:E4, F1:F4, 1, TRUE)

where: Means Vector is the location of the Nx1 vector of means. If the Moments switch is true, each mean must be greater than zero. If the Moments is false, mean are reals.
Covariance is an NxN covariance matrix for the series,
Array of ISND is an optional nx1 array of independent standard normal deviates generated with N cells of =NORM( ).
Matrix Row is an optional term for the ith variable if the function is to return only the random value for the ith variable (Leaving this value blank makes the function return n values so treat it as an array function with *Control Shift Enter.*), and
Moments is an optional switch to use the function two ways: if the term is TRUE ('1') the function is for the moments of a log normal vector, and FALSE ('0') indicates the moments are for a transformed normal distribution.

3.9.9 Multivariate Student's t. A distribution of N variables can be simulated multivariate Student's t using the =MVTINV( ) array function as:

=MVTINV (Means Vector, Covariance Matrix, [Array of ISND], [Matrix Row])
=MVTINV (A1:A4, B1:E4, F1:F4, 1,)
=MVTINV (A1:A4, B1:E4, F1:F4)

where: Means Vector is the location of the Nx1 vector of means,
Covariance Matrix is the location of the NxN covariance matrix for the series,
Array of ISND is an optional Nx1 array of N cells with =NORM( ) SNDs, and
Matrix Row is the optional ith variable if only the random number for the ith series is to be simulated. Leaving this value blank makes the function return N values so treat it as an array function with *Control Shift Enter.*

**3.9.10 Hotelling T-Squared.** The Hotelling T^2 distribution is a multivariate analog to the univariate Student's t distribution. If $\underline{x}$ is a Px1 random vector distributed as multivariate normal with a zero mean vector and an identity covariance matrix and W is a PxP random matrix distributed as Wishart with an identity covariance matrix and m degrees of freedom. And $\underline{x}$ and W are independent, then the variable $T^2 = m \, \underline{x}^T W^{-1} \underline{x}$ is distributed as a Hotelling T^2 random variable. A special case is the Hotelling T^2 random variable with 1 and M degrees of freedom which is an F distribution with 1 and M degrees of freedom. The parameters for the Hotelling T^2

function, which produces a Hotelling T^2 random variable, are p and df = M. Simulate Hotelling T-Squared distribution as:

=HOTELLTINV(P, DF, [USD])

where: P is an integer indicating the dimension of the PxP covariance or identity matrix for a
Wishart distribution,
DF is the degrees of freedom or the number of observations in the MxP data matrix for a
Wishart distribution, and
USD is an optional uniform standard deviate.

3.9.11 Wishart. The Wishart distribution is a matrix generalization of the univariate chi square distribution. The Wishart array function produces a matrix of random values that are distributed Wishart. The distribution is derived from an MxP matrix X of normally distributed independent vectors with mean zero and covariance matrix C. The PxP matrix of X'X has a Wishart distribution and is simulated as:

=WISHINV(C, DF)

where: C is a PxP covariance matrix that is positive definite, and
DF is the degrees of freedom or the number of rows in an MxP data matrix of values
used to calculate C.

The Wishart function is an array function so highlight a PxP block of cells, type the function, and end by pressing the *Control Shift Enter* keys.

3.9.12 Wilks' Lambda. If two independent random matrices, X and Y, are distributed as Wishart, both with a PxP identity covariance matrix and N1 and N2 degrees of freedom, respectively, then the scalar |X|/|X+Y| has the Wilks' lambda distribution with P, N1, and N2 degrees of freedom. This distribution is found in several likelihood ratio tests in multivariate testing settings. Simulate Wilks' lambda distribution as:

=WILKSLINV(P, N1, N2)

where: P is an integer representing the dimension of the Wishart random matrix PxP
N1 is the integer value for the degrees of freedom in the random Wishart matrix X, and
N2 is the integer value for the degrees of freedom in the random Wishart matrix Y.

3.9.13 Dirichlet. A Dirichlet series of correlated random variables can be simulated using the Dirichlet array function as:

=DIRICHINV(Alpha Array, [Array of IUSD], [Matrix Row])

where: Alpha Array is the location of an Nx1 array of parameter values for the Dirichlet
distribution; each value is greater than zero,
Array of IUSD is the location of an optional nx1 array of independent uniform standard
deviates simulated =UNIFORM( ), and
Matrix Row is the ith variable of the random series if the function is to return only the

ith series. Leaving this value blank makes the function return n values so treat it as an array function with *Control Shift Enter.*

3.9.14 Uncorrelating Random Deviates (USD and SND). In advanced simulation applications it is useful to uncorrelate random values. Simetar provides a function to calculate the implicit independent deviates from a vector of CUSDs. The uncorrelated standard normal deviates function, =USND( ), converts a vector of CSNDs to a vector of independent SNDs. The function is programmed as:

=USND (Correlation Matrix, CSND Array)

where: Correlation Matrix is the cell reference location for the correlation matrix used to generate the CSNDs, and CSND Array is the cell reference location for the array of CSNDs to be converted to independent SNDs.

The uncorrelated uniform standard deviates function, =UUSD( ), converts correlated uniform standard deviates (CUSDs) to uncorrelated USDs. The function is programmed as follows:

=UUSD (Correlation Matrix, CUSD Array)
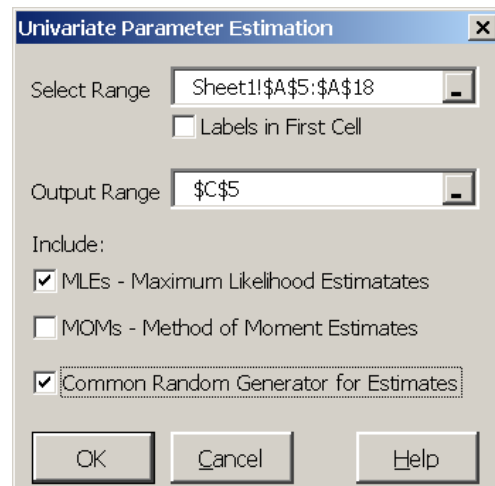
Figure 12. Univariate Parameter Estimator

3.10 Iteration Counter

For advanced simulation applications it is useful to use the iteration number to key a simulation model to perform certain calculations. For example a table lookup function can be used to draw values from a table where the rows correspond to the iterations for previously generated and tested random values. The iteration number function in Simetar is =ITERATION() and returns the iteration number from 1 to N, where N represents the number of iterations. As indicated in the example below, the function returns "1" until the workbook is simulated. Selecting the cell with =ITERATION( ) as a KOV for simulation will produce a series of values: 1, 2, 3, … , 500 for a stochastic simulation with 500 iterations.

=ITERATION ( )

	A	B	C	D	E
721	3.10 Iteration Counter in Simetar shows the iteration during simulation				
722	Count of present iteration			1	=ITERATION()

4.0 Parameter Estimation for Probability Distributions

4.1 Parametric Probability Distributions

A univariate parameter estimator in Simetar estimates the parameters for simulating a random variable for 16 parametric probability distributions. The univariate parameter estimator is activated by using the θ icon. The Simetar menu for the univariate parameter estimator requires the user to specify the historical data series for the random variable and the method for estimating the parameters: method of moments or maximum likelihood estimator (Figure 12). If

29

a variable is not consistent with a distribution, its parameter cells will be blank rather than contain a value.

Simetar also prepares the equations for simulating the random variable using the calculated distribution parameters in the Formulas column of the example above. The formulas in the Formulas column can be simulated to test how well the different assumed distributions simulate the random variable. The =CDFDEV( ) function can be used to calculate a test scalar to determine which distribution is best for simulating the random variable. See Section 5.7 for an explanation of =CDFDEV( ). An example of the parameter estimation is provided in Parameter Estimation Demo.xls. The =CDFDEV( ) scalar in the example above indicates that the Beta distribution fits the data better than the other distributions tested.

4.2 Empirical Probability Distributions

The parameters for an empirical probability distribution are estimated using a Simetar function activated by the ⎙ icon. The *Select Input Ranges* window indicates the data to be used for defining the probability distribution (Figure 13). Be sure to select the *Labels in First Cell* box when there is a name in the first cell (row or column) of the *Selected Input Ranges*. Four examples of using the Empirical Parameter estimation dialog box are provided in the Empirical Demo.xls workbook program.

Figure 13. Parameter Estimation for the Empirical Distribution Dialog Box.

The dialog box (Figure 13) allows the user to estimate the parameters for one empirical

	C	D	E	F	G	H	I	J
5	Univariate Parameter Estimation for Wheat P at 12/18/2005 1:06:30 PM						Compare Simulated	
6			MLEs			Fomulas	Results to History	
7	Distribution	Parameters	Parm. 1	Parm. 2	Distribution	MLE	CDFDEV	SCALER
8	Beta	α, β; A≤x≤B, α,β>0	0.739	1.004	Beta	3.206	1.962765	
9	Double Exponential	α, β; α≤x<∞, -∞<α<∞, β>0	3.260	0.513	Double Exp	3.239	7.267778	
10	Exponential	μ, σ; -∞<x<∞, -∞<μ<∞, σ>0	2.420	0.874	Exponential	2.991	1.955898	
11	Gamma	α, β; 0≤x<∞, α,β>0	27.204	0.121	Gamma	3.222	3.335559	
12	Logistic	μ, σ; 0≤x<∞, -∞<μ<∞, σ>0	3.278	0.371	Logistic	3.248	5.38898	
13	Log-Log	μ, σ; -∞<x<∞, -∞<μ<∞, σ>0	2.990	0.530	Log-Log	3.154	2.718552	
14	Log-Logistic	μ, σ; 0≤x<∞, -∞<μ<∞, σ>0	8.493	3.226	Log-Logistic	3.196	3.863648	
15	Lognormal	μ, σ; 0≤x<∞, -∞<μ<∞, σ>0	1.174	0.192	Lognormal	3.203	3.096948	
16	Normal	μ, σ; -∞<x<∞, -∞<μ<∞, σ>0	3.294	0.639	Normal	3.262	4.147795	
17	Pareto	α, β; α≤x<∞, α,β>0	2.420	3.449	Pareto	2.925	1.958128	
18	Uniform	a, b; a≤x≤b	2.420	4.550	Uniform	3.442	1.939766	
19	Weibull	α, β; 0≤x<∞, α,β>0	5.480	1055.587	Weibull	3.296	5.300051	
20	Binomial	n, p; x=0,1,2,...,n; 0≤p≤1	4.000	0.712	Binomial	3.000	13.20388	
21	Geometric	p ; x=1,2,...; 0≤p≤1	0.260		Geometric	3.000	7.140645	
22	Poisson	λ ; x=0,1,...; 0≤λ<∞	2.846		Poisson	3.000	13.20388	
23	Negative Binomial	s, p; x=1,2,...; 0≤p≤1			Negative Bir			
24						Common USD		
25						0.48		

distribution or for numerous distributions at once. The only restriction for using this function is that all of the data series must have the same number of observations. The dialog box allows estimation of the parameters four different ways:

> − Use actual data (no transformations) for the distribution,
> − Convert the actual data to differences (residuals) from the mean prior to estimating the parameters,
> − Convert the actual data to deviations (residuals divided by the mean) from the mean prior to estimating the parameters, and
> − Convert the actual data to deviations (residuals divided by the trend values) from a linear trend line prior to estimating the parameters.

The empirical distribution parameter estimation output includes the random data (residuals from trend or mean), summary statistics, correlation matrix if more than one variable is specified, and the sorted random values with cumulative distribution probabilities. The sorted deviations required by =EMP( ) to simulate an empirical distribution are demonstrated in the example output to the right. Once the Empirical distribution parameters are estimated, they can be simulated using the =EMP (S_i, $F(S_i)$) function (see Section 3.7.1).
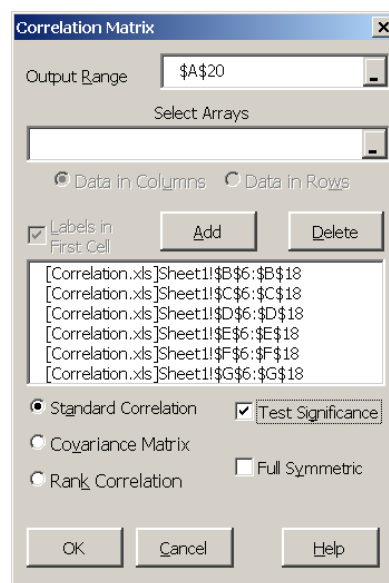
Figure 14. Correlation Matrix Dialog Box.

4.3 Multivariate Probability Distributions

Correlation and covariance matrices are both parameters for multivariate probability distributions. Correlation and covariance matrices can be calculated using the Correlation Matrix dialog box activated by the icon. This Simetar dialog box calculates the upper right triangle correlation matrix of size NxN when the user specifies N variables. The first step to using the dialog box is to specify the location for placing the upper left hand corner of the generated correlation matrix by indicating the *Output Range* in the menu (Figure 14). Next, specify whether the data to correlate are in columns or rows. The first cell of each column (or row) indicated in the *Selected Arrays* box should have a label so the output matrix is easier to read.

The Correlation Matrix dialog box calculates either the Pearson's (standard) correlation coefficient matrix or the rank correlation matrix. The default is the Pearson's correlation coefficient matrix. The rank correlation coefficient matrix is calculated when the *Rank Correlation* radio button is selected.

The statistical significance of each correlation coefficient can be tested by Simetar. Student's-t values for the correlation coefficients greater than the t-critical value indicate whether the correlation coefficient is statistically different from zero and are displayed in bold. See Complete Correlation Demo.xls for examples of using the correlation matrix dialog box.

A covariance matrix can be calculated using the Correlation Matrix dialog box (Figure 14). The upper triangle covariance matrix is calculated by selecting the *Covariance Matrix* radio button after specifying the arrays to include in the matrix. The *Full Symmetric* covariance matrix is

calculated by selecting this option in the dialog box and the *Covariance Matrix*. See the demonstration program Complete Correlation Demo.xls for examples of estimating covariance matrices.

	A	B	C	D	E	F	G
20	**Correlation Matrix**						
21		Corn P	Wheat P	Sorg P	Corn Y	Wheat Y	Sorg Y
22	Corn P	1	0.87	0.98	-0.30	-0.18	-0.53
23	Wheat P		1	0.82	-0.16	-0.37	-0.34
24	Sorg P			1	-0.29	-0.17	-0.56
25	Corn Y				1	0.53	0.59
26	Wheat Y					1	0.47
27	Sorg Y						1
28							
29	Correlation Coefficient t-values.						
30	Significanc	95%		t-critical	2.20		
31		Corn P	Wheat P	Sorg P	Corn Y	Wheat Y	Sorg Y
32	Corn P		**5.83**	**16.79**	1.04	0.60	2.06
33	Wheat P			**4.68**	0.54	1.31	1.21
34	Sorg P				1.00	0.58	**2.27**
35	Corn Y					2.08	**2.40**
36	Wheat Y						1.78
37	Sorg Y						
38	Bold values indicate statistical significance at the specified level.						
39							
40	**Covariance Matrix**						
41		Corn P	Wheat P	Sorg P	Corn Y	Wheat Y	Sorg Y
42	Corn P	0.160533	0.222644	0.160682	-1.68098	-0.20039	-1.20096
43	Wheat P		0.408639	0.213254	-1.42204	-0.66108	-1.24053
44	Sorg P			0.16711	-1.64783	-0.1966	-1.30985
45	Corn Y				194.9708	20.82524	46.47593
46	Wheat Y					7.887421	7.534503
47	Sorg Y						32.17119

4.4 GRKS Probability Distribution

Parameters for the GRKS probability distribution can be estimated using the dialog box in Figure 15. The GRKS distribution dialog box is accessed via the toolbar Simetar drop down menu – GRKS Distribution. The GRKS pdf is defined by three values: *Minimum, Middle Value, and Maximum*. Simetar places the parameters on the worksheet starting in the designated *Output Range*. The parameters are presented as values and their associated probabilities (see GRKS Distribution Demo.xls). Simetar also generates a chart of the distribution and that displays how the shape of the distribution changes as the minimum, middle, and maximum values change. Test this feature by changing the three parameters and observing their affects on the GRKS distribution figure. The GRKS pdf parameters can be simulated using the =GRKS( ) function in Section 3.5.2.

Figure 15. Parameter Estimation for the GRK Distribution.

5.0 Statistical Tests for Model Validation

Model validation must be done prior to application of a simulation model for decision making. Validation can utilize graphs, such as PDFs and CDFs, but statistical testing of the simulated distributions is required to determine whether the stochastic variables in the model are

	A	B	C	D	E	F	G	H
52	Step 3. Compare Two Distributions Using Student's-t and F Tests			Compare Historical Corn P with the Simulated Corn P				
53	Distribution Comparison of Corn P & Sim Corn P							
54	Confidence Level		95.0000%					
55		Test Value	Critical Value	P-Value				
56	2 Sample t Test	-0.02	2.49	0.982	Fail to Reject the Ho that the Means are Equal			
57	F Test	1.03	1.85	0.428	Fail to Reject the Ho that the Variances are Equal			

statistically from the same distribution as the historical data. To facilitate the validation process several hypothesis tests have been included in Simetar. The tests are organized using 5 tabs in the Hypothesis Testing for Data dialog box opened by the ⬛ icon (Figure 16). Examples of the validation tests described in this section are available in Hypothesis Tests Demo.xls.

5.1 Univariate Distribution Tests for Model Validation

The means and variances for two distributions (or series) can be compared by using the *Compare Two Series* tab for the Hypothesis Testing dialog box (Figure 16). The mean and variance tests are univariate as they only test the difference between two variables. This type of hypothesis testing is useful in validation for comparing the simulated distribution to the historical distribution. The null hypotheses are that the simulated mean equals the historical mean and the simulated variance equals the historical variance. As demonstrated in the example below, it is useful to statistically test if the simulated data have the same mean and variance as the historical data series.

The statistical tests are performed when the *Compare Two Series* tab in Figure 16 is selected and you specify the two distributions (data series) to compare. A two sample, Student-t test is used to allow comparison of means from distributions with an un-equal number of observations (see example below). See Step 4 in Hypothesis Tests Demo.xls for an example of comparing two distributions.
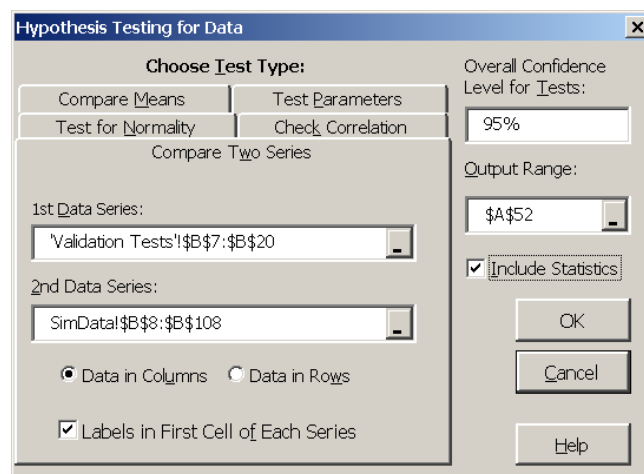
Figure 16. Univariate and Multivariate Distribution Tests.

5.2 Multivariate Distribution Tests for Model Validation

Means and variances for multivariate (MV) probability distributions can be statistically tested against the distribution's historical data in one step by selecting the *Compare Two Series* tab in the Hypothesis Testing for Data dialog box and specifying matrices as the input Figure 17.

The first MV test uses the two-sample Hotelling T^2 test which tests whether two data matrices (the historical data MxN and the simulation results PxN) statistically have equivalent mean vectors and covariance matrices. Assume historical data are arranged in an MxN matrix and the simulated data are in a PxN matrix, where P
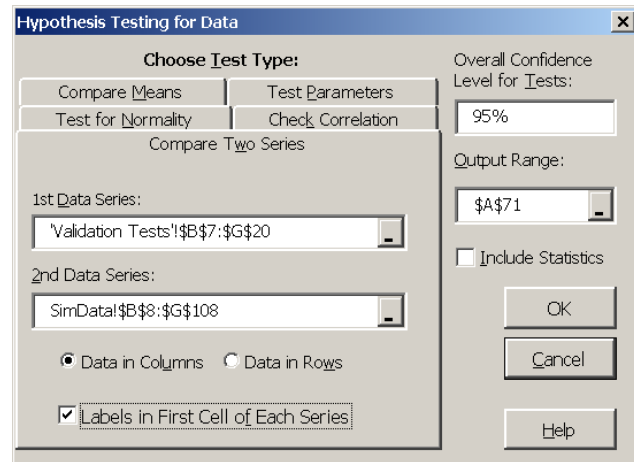
Figure 17. Multivariate Hypothesis Tests for Six Variables.

is the number of iterations, then the means can be tested with the Hotelling T^2 test procedure. The Hotelling T^2 test is analogous to a Student's-t test of two means in a two-sample univariate case.

The second MV test calculated for this statistical test, Box's M, tests the equality of the covariance matrices with dimensions MxM and PxN, respectively, using a large sample likelihood ratio testing procedure. The Box's M test of homogeneity of covariances is used to test whether the covariance matrices of two or more data series, with n columns each, are equal. The assumptions under this test are that the data matrices are MV normal and that the sample is large enough for the asymptotic, or central Chi-Squared, distribution under the null hypothesis to be used.

	A	B	C	D	E
69	**Step 4. Compare Simulated Results to the Input Distribution**				
70	**Distribution Comparison of Two Data Series**				
71	**Distribution Comparison of Corn P & Sim Corn P**				
72	Confidence Level		95.0000%		
73		Test Value	Critical Value	P-Value	
74	2 Sample Hotelling T^2 Test	0.00	13.73	1.000	*Fail to Reject the Ho that the Mean Vectors are Equal*
75	Box's M Test	1.99	32.67	1.000	*Fail to Reject the Ho that the Covariance Matrices are Equivalent**
76	Complete Homogeneity Test	2.12	40.11	1.000	*Fail to Reject the Ho that the Mean Vectors and Covariance Matrices are Equivalent, Respectively**
77					*Based on asymptotic distribution*

The third MV test is the Complete Homogeneity test. This statistical test simultaneously tests the mean vectors and the covariance matrices for two distributions. The historical data's mean vector and covariance matrix are tested against the simulated sample's mean vector and covariance matrix. If the test fails to reject that the means and covariance are statistically equal, then one can assume that the multivariate distribution in the historical series is being simulated appropriately. An example of this test is provided above and in Step 4 of Hypothesis Tests Demo.xls.

5.3 Test Correlation

Another multivariate distribution validation test in Simetar is a test to compare the correlation matrix implicit in the simulated output to the input (assumed) correlation matrix. This test is useful for validating multivariate probability distributions, particularly the non-normal multivariate distributions. Selecting the *Check Correlation* tab in the Hypothesis Testing for Data dialog box (Figure 18) calculates the Student's-t test statistics for comparing the corresponding correlation coefficients in two matrices. The dialog box requires information for the location of the simulated series (a PxN matrix) and the location of the NxN correlation matrix used to simulate the multivariate distribution (or the correlation matrix implicit in the historical data for the distribution).

The confidence level for the resulting Student's-t test defaults to a value greater than 0.95 but can be changed by the user after the test has been performed. An example of this test is provided in Step 6 of Hypothesis Tests Demo.xls. If a correlation coefficient for two simulated variables is statistically different from the respective historical correlation coefficient, the Student's t-test statistic will exceed the Critical Value and its respective statistic will be displayed as a bold value. If the test shows several bold values check the formulas used to simulate the multivariate distribution to insure the distribution is modeled correctly.
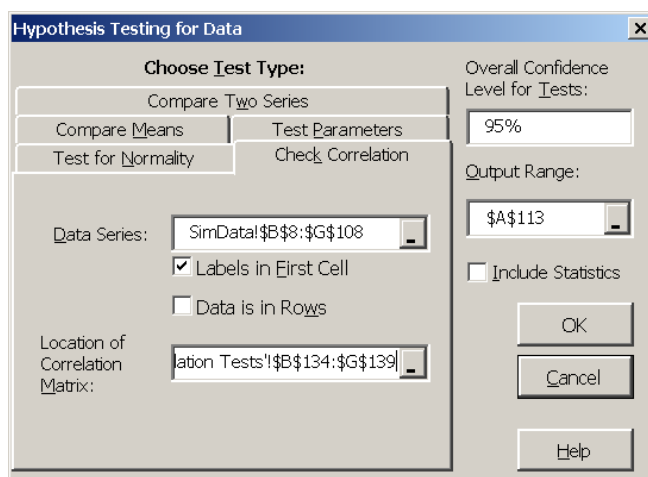
Figure 18. Test Correlation of MV Distribution Simulation Results.

	A	B	C	D	E	F
112	**Step 6. Compare Simulated Correlation Matrix to Input Correlation Matrix**					
113	**Test Correlation Coefficients**					
114	Confidence Level		99.6586%			
115	Critical Value		3.00			
116						
117		Sim Wheat P	Sim Sorg P	Sim Corn Y	Sim Wheat Y	Sim Sorg Y
118	Sim Corn P	0.31	0.03	0.39	0.79	0.21
119	Sim Wheat P		0.42	0.50	0.48	0.57
120	Sim Sorg P			0.38	0.94	0.28
121	Sim Corn Y				0.67	1.19
122	Sim Wheat Y					0.57

5.4 Test Mean and Standard Deviation

The mean and standard deviation for any data series (e.g., simulated data) can be compared to a specified mean and standard deviation using the *Test Parameters* tab in Figure 19. The Student's-t test is used to compare the user specified mean to the observed mean of any distribution (or series) as demonstrated in Figure 19. A Chi-Squared test is used to test a user specified standard deviation against the standard deviation for any distribution. The null hypothesis is that the statistic for the series equals the user's specified values. An example of testing the historical data for a variable against a specified mean and standard deviation is provided below and in Hypothesis Tests Demo.xls.
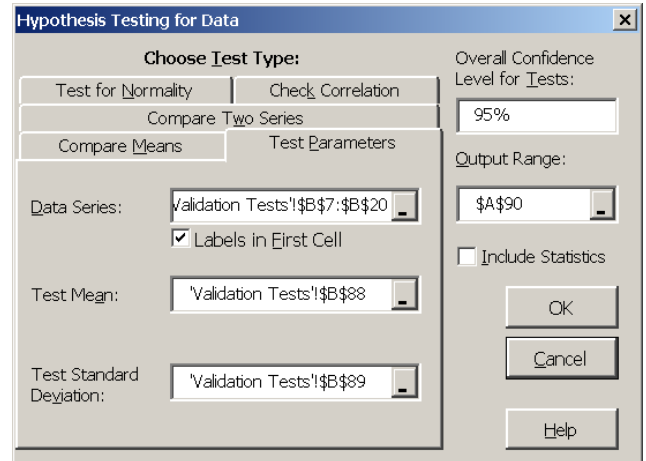
Figure 19. Test Mean and Standard Deviation for a Univariate Distribution.

	A	B	C	D	E	F	G	H	I	J	K
87	Step 5. Compare Simulated Results to an Assumed Mean and Standard Deviation										
88	Assumed Mean	2.325									
89	Assumed Std Dev	0.400									
90	Test of Hypothesis for Parameters for Corn P										
91	Confidence Level		95.0000%								
92		Given Value	Test Value	Critical Value	P-Value						
93	t-Test	2.325	0.00	2.56	1.00	Fail to Reject the Ho that the Mean is Equal to 2.325					
94	Chi-Square Test	0.400	13.01	LB: 4.40	0.74	Fail to Reject the Ho that the Standard Deviation is Equal to 0.4					
95				UB: 23.34							

5.5 Univariate Tests for Normality

Five different tests for normality can be performed by selecting the *Test for Normality* tab after the ▦ icon is selected (Figure 20). The normality tests are: Kolmogornov-Smirnoff, Chi-Squared, Cramer-von Mises, Anderson-Darling, and Shapiro-Wilks. The Chi-Squared test requires the number of bins (or intervals); 20 or more intervals appear to work for most data series. In addition to the normality tests this option calculates the skewness and kurtosis, relative to a normal distribution (not shown in the example below). See an example of these normality tests in Hypothesis Tests Demo.xls.
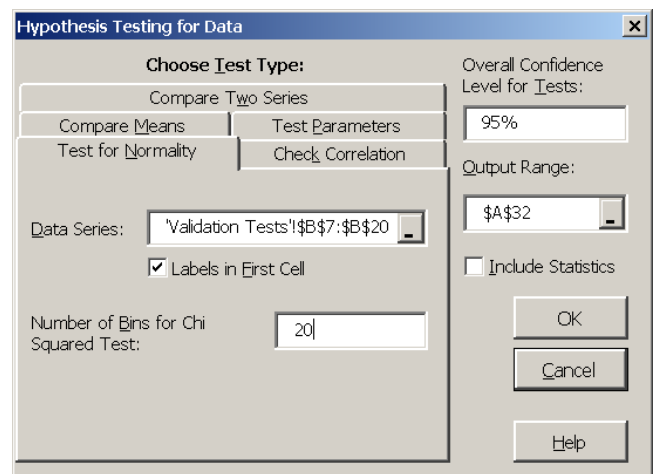
Figure 20. Univariate Normality Test.

	A	B	C	D	E	F	G	H
31	Step 2. Tests for Normality							
32	Test for Normality of Distribution for Corn P							
33	Confidence Level		95.0000%					
34	Procedure	Test Value	p-Value					
35	S-W	0.962956115	0.798551994	Fail to Reject the Ho that the Distribution is Normally Distributed*				
36	A-D	0.303060068	0.523780533	Fail to Reject the Ho that the Distribution is Normally Distributed*				
37	CvM	0.04591582	0.544799272	Fail to Reject the Ho that the Distribution is Normally Distributed*				
38	K-S	0.149172142	NA	Consult Critical Value Table				
39	Chi-Sqared	13.61538462	0.805633117	Fail to Reject the Ho that the Distribution is Normally Distributed*				
40				*Based on approximate p-values				

5.6 Multivariate Tests for Normality

A multivariate distribution test for normality can be performed on any data matrix of PxN. The MV normality test can be performed by specifying a PxN matrix in the *Data Series* box for the *Test for Normality* tab in the Hypothesis Testing dialog box (Figure 21). The MV normality tests are: skewness criterion, kurtosis criterion, and Chi-Squared quantile correlation. Simetar reports the test statistics, critical value, and p-value for the first two tests and the test statistic for the third test. The null hypothesis is that the data matrix is distributed MV normal. See the example output for this test below and in Hypothesis Tests Demo.xls.
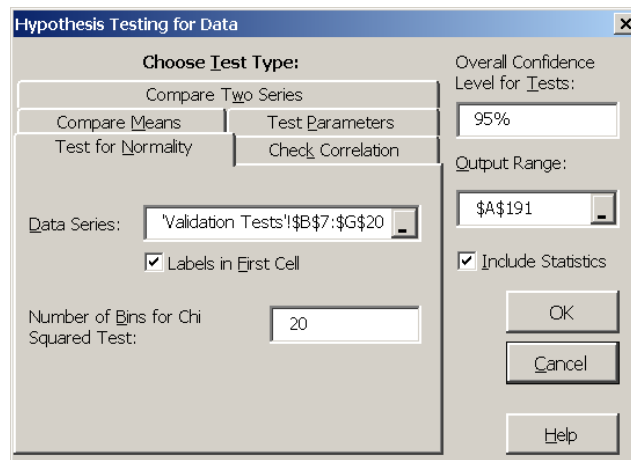
Figure 21. Multivariate Normality Tests Dialog Box.

	A	B	C	D	E	F	G	H	I	J
190	Step 9. Multivariate Test for Normality									
191	Test for Normality of a Multivariate Distribution									
192	Confidence Level		95.0000%							
193	Procedure	Test Value	Critical Value	p-Value						
194	Skewness Criterion	43.65956655	74.4683241	0.88502126	Fail to Reject the Ho that the Data Are Multivariate Normally Distributed*					
195	Kurtosis Criterion	-1.196897667	1.959963985	0.115673222	Fail to Reject the Ho that the Data Are Multivariate Normally Distributed*					
196	Chi-Squared Quantile Correlation	0.940571025			*Based on asymptotic p-values					

5.7 Compare Means (ANOVA)

The Hypothesis Testing for Data dialog box includes a means test (ANOVA) capability (Figure 22). Selecting the *Compare Means* tab in the Hypothesis Testing for Data dialog box produces a menu for specifying the two series to compare. For this test the user must specify the two distributions (or series) using the *Select Data Series* to compare window and the Add button to list the series in the window at the bottom. The confidence level defaults to 0.95 and must be specified before clicking the *OK* button. The results of the ANOVA test are the sum of squares, mean square error, F-statistic and its p-value. A sample ANOVA test is demonstrated below and is provided in Hypothesis Tests Demo.xls.
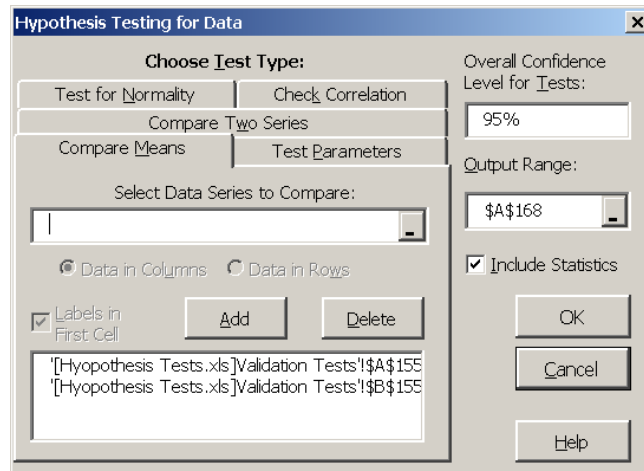
Figure 22. Compare Means Test Dialog Box

	A	B	C	D	E	F
167	Step 8. Compare Means for Two Series in other words, an ANOVA test					
168	Tests to Compare Multiple Means					
169	ANOVA Table					
170	Source of	Sum of	Degrees of	Mean		
171	Variation	Squares	Freedom	Square	F statistic	p-value
172	Treatments	6	1	6	19.51642633	0.0002175
173	Error	6.763533333	22	0.307433333		
174	Total	12.76353333	23			

5.8 Compare Two Cumulative Distribution Functions (CDFs)

A scalar measure to compare the difference between two cumulative distribution functions (CDFs) is calculated by the =CDFDEV( ) Simetar function. The function calculates the sum of the squared differences between two CDFs with an added penalty for differences in the tails. The scalar is calculated for two CDFs, F(x) and G(x) as:

$$CDFDEV = \sum_{i=1}^{N} (F(x_{(i)}) - G(x_{(i)}))^2 + w_i$$

where: w_i is a penalty function that applies more weight to deviations in the tails than values around the mean.

If the G(x) distribution is the same as the F(x) distribution, then the CDFDEV value equals zero. The CDFDEV measure is programmed to compare a historical series Nx1 to a simulated series Px1 as follows:

   =CDFDEV(Range for Historical Series, Range for Simulated Series)

where: Range for Historical Series is the location for the historical data, such as B1:B10, and Range for Simulated Series is the location for the simulated values, such as B9:B109.

The =CDFDEV( ) function is useful when testing the ability of different assumed probability distributions to simulate a random variable. In this case, the =CDFDEV( ) measure is calculated using the simulated values for each of the alternative probability distributions. The probability distribution associated with the lowest =CDFDEV( ) scalar is the "best" distribution for simulating the random variable. See Parameter Estimation Demo.xls for an example.

Compare Ability of Alternative Distributions to Fit a Series			
Distributions Tested	CDFDEV()	Formula	
Beta	0.0163	=cdfdev(Sheet1!$A$6:$A$18,SimData!B9:B108)	
Double Exponential	2.2610	=cdfdev(Sheet1!$A$6:$A$18,SimData!C9:C108)	
Exponential	1.1321	=cdfdev(Sheet1!$A$6:$A$18,SimData!D9:D108)	
Gamma	0.2233	=cdfdev(Sheet1!$A$6:$A$18,SimData!E9:E108)	
Logistic	1.2496	=cdfdev(Sheet1!$A$6:$A$18,SimData!F9:F108)	
Log-Log	0.3239	=cdfdev(Sheet1!$A$6:$A$18,SimData!G9:G108)	

6.0 Graphical Tools for Analyzing Simulation Results

Simetar provides nine graphics tools for displaying the results of stochastic simulations and for analysis of data. These graphics tools utilize the charting capabilities of Excel so all charts and graphs can be edited and enhanced using standard Excel charting tools. Simetar charts and graphs are developed using menus which allow the user to easily specify the data, titles, and labels for charts that are used frequently for simulation. An example of Simetar's charts is provided in Charts Demo.xls.

6.1 Line Graph

Any series of numbers can be graphed on an X-Y axis as a line graph using this option. The icon to access line graphs is ![LINE icon] . The Line Graph menu (Figure 23) requires that you specify the values for the X axis (such as, years) and the Y values (such as, prices) in the *X and Y-Range* boxes. Labels for these variables are optional and are entered in the *Y and X-Axis Label* boxes. The *Chart Title* is optional. You may include a label in the first cells (row or column) indicated for each Y variable, if you select the box for *Series Labels in First Cell*.

The chart can have more than one line by using the *Add Y's* button and indicating multiple Y series in the *Select Y-Axis Range*, one at a time or all at once if the variables are contiguous. Once the graph is drawn by Excel, it can be edited using Excel chart commands.
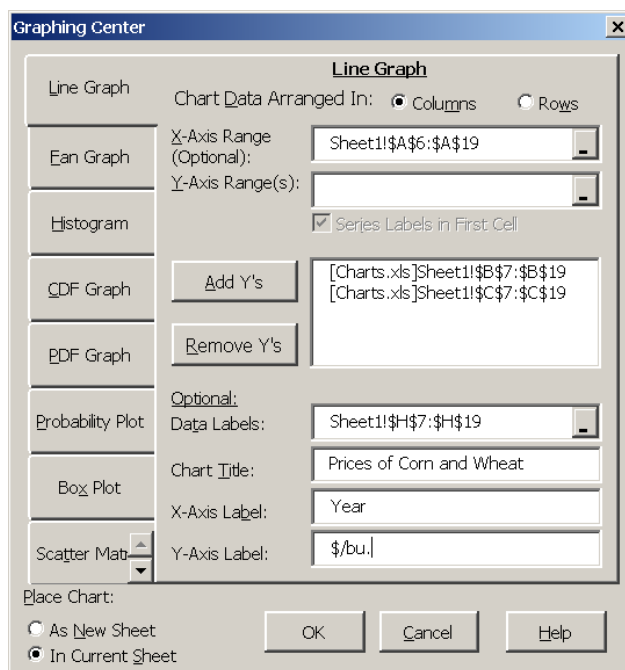
Figure 23. Dialog Box for Developing a Line Chart.

The Line Graph dialog box allows the user to label the points on line graphs. For example, a price/quantity chart can be developed with year labels on the individual data points to show years when structural changes took place. To use this option indicate the column or row of labels in the *Data Labels* box, being sure to have the same number of labels as there are rows (or columns) of data to graph. The result of the chart specified in Figure 18 is presented below and in Charts Demo.xls.
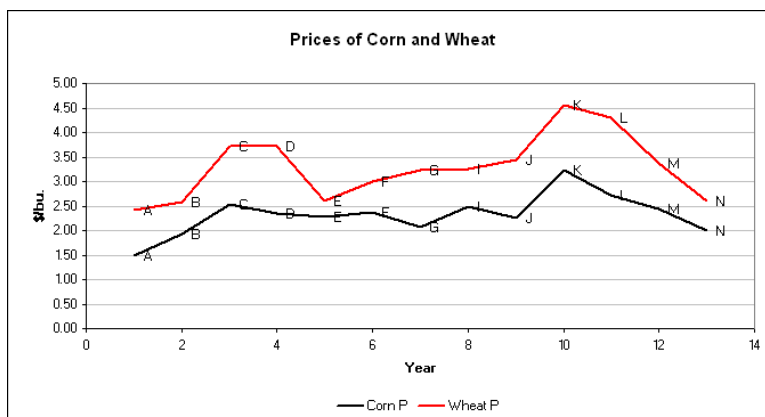
6.2 CDF Graph

Cumulative distribution function (CDF) charts of individual or multiple variables (simulated values) can be developed using Simetar. CDF graphs are initiated by selecting the [CDF icon] icon. Identify the variables to graph by highlighting the column(s), after first clicking in the *Select Range to Graph* box (Figure 24). Include names in the first cell of the variable range, so the chart will include names for the individual lines. (Be sure the variable names begin with a letter.) The chart can be placed on the current worksheet or in a new chart sheet. Use Excel's chart commands to format the scale for the X axis and to make changes to the title.

CDF graphs developed using Simetar are dynamic so when the values referenced for the chart change, the CDF graph is automatically updated by Excel. This feature is particularly useful for simulation. Each time the simulation results are updated in SimData, the CDF graphs will be updated.
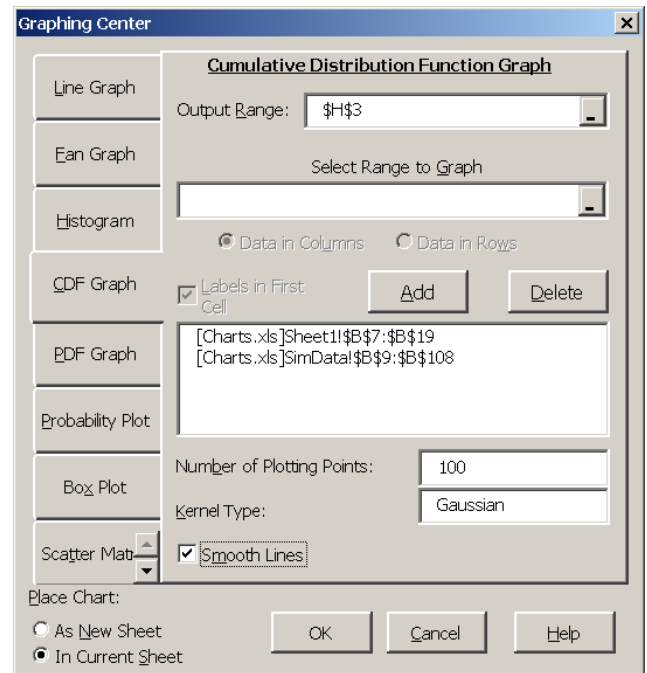
Figure 24. CDF Chart Dialog Box.

The *Smoothing* option in the CDF menu utilizes kernel density functions to smooth the observed values and develop smoothed CDF charts. In addition to the CDF charts, the output for this option includes a text box with a drop down menu to allow the user to select the kernel. The default kernel is the Gaussian, but ten more are provided. The kernel smoothed CDF for a historical series depicts the probability distribution Simetar would use if the series was simulated using =KDEINV( ), see Section 3.7.4.

The CDF graph option is useful for comparing simulated values of a random variable to the variable's historical data. This is possible in Simetar even though the two series have a different number of observations. See the example below and in Charts Demo.xls.
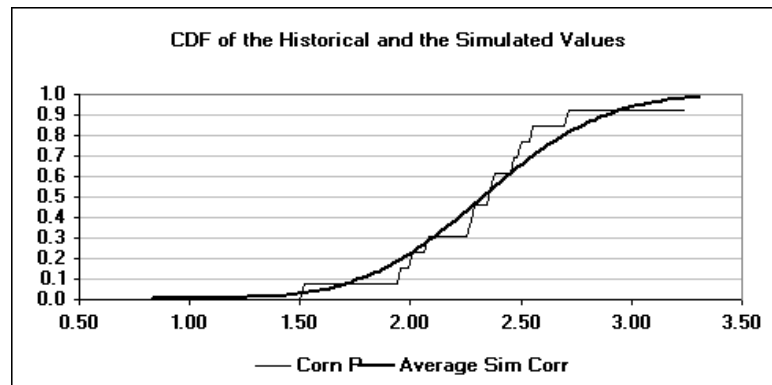
6.3 PDF Graph

Probability distribution function (PDF) graphs of individual or multiple variables can be estimated using the [PDF] icon. Identify the variables to include in the PDF graph by selecting the variables in the *Select Range to Graph* box and the *Add* button if the variables are not in continuous columns (or rows) (Figure 25). The PDF graph function uses kernel estimators to smooth the data rather than just using line segments to connect the dots. Eleven kernels are available to develop the PDF graphs: Gaussian, Cauchy, Cosinus, Double Exp., Epanechnikov, Histogram, Parzen, Quartic, Triangle, Triweight, and Uniform. Once the graph is drawn you can change the kernel by editing the output range in the worksheet.

If the data series have names in the first cell indicate this on the menu, otherwise unselect the *Labels in First Cell* option. Multiple PDFs can appear on the same axis so the simulated values and their historical values can both be graphed on the same axis. This feature is possible because the data series being graphed do not have to be the same lengths.

Figure 25. PDF Chart Dialog Box.

PDF graphs developed using Simetar are dynamic so when the values in the *Selected Range to Graph*, change the graph is instantly updated. This feature is useful when displaying simulation results using PDFs. The mean of the variables in a PDF is included in the chart. Confidence intervals at the alpha equal 5 percent level can be added by selecting the *Plot Quantiles*. The quantiles can be redrawn by changing the Alpha equal 0.9 to 0.10 in the seventh row of the PDF Graph output table. The title can be changed by editing the first line of the PDF Graph output. See the example below of a PDF chart developed for a simulated series in Charts Demo.xls.

6.4 Histograms

Histograms of individual variables (simulated output) can be developed using the Simetar menu. The histogram icon ![HIST] activates this option. Indicate the variable to graph by clicking the *Select Range to Graph* box in the dialog box (Figure 26) and highlighting the variable in the worksheet. Specify the *Number of Bins* (intervals) and select *OK*. The more bins the smoother the histogram. The maximum number of bins is the number of observations minus one. Experiment with the number of bins to find the number which best suits the data. An added feature of the histogram option in Simetar is to display data as a cumulative distribution with the bins growing in height from zero to one as the X value gets large.

Figure 26. Histogram Dialog Box.
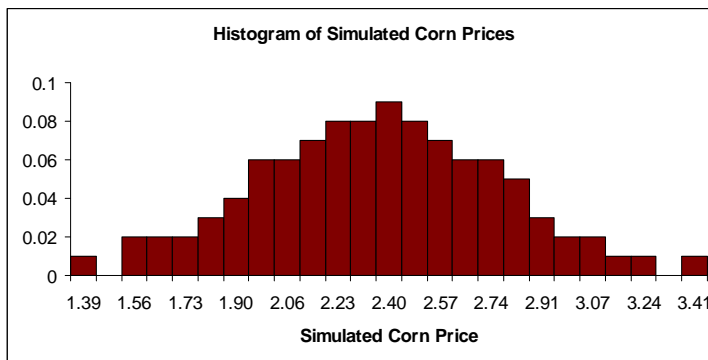
6.5 Fan Graph

A Fan Graph consists of multiple lines in the Y axis for multiple scenarios (or multiple years of one variable) graphed in the X axis. The variables graphed in the X axis can be successive years for a simulated output variable. Alternatively, the variables on the X axis can be the same simulated variable but for different scenarios. The purpose of a Fan Graph is to show the effect of risk on a variable over time or across scenarios.

A Fan Graph showing the simulated mean and percentiles or confidence interval lines about the mean can be developed using the ![FAN] icon in Simetar. The range of variables to be graphed on the X axis

Figure 27. Fan Graph Dialog Box.

must be specified in the *Select Ranges to Graph* box (Figure 27). The variables (scenarios or years) must be specified in the order they appear in the graph. For example, if the graph is for 10 years of a probabilistic forecast, specify the 10 variables across the, say, 500 iterations as the selected range to graph. If the variables are not contiguous, they can be specified one at a time using the *Add* box.

The Fan Graph dialog box (Figure 27) provides boxes to specify up to six percentile or confidence lines about the mean. The individual lines to add to the Fan Graph must be specified as fractions, such as 0.05 and 0.95 would result in a graph with 3 lines: the mean, the 5 percentile and the 95 percentile lines. Once the Fan Graph has been developed, you can dynamically change the graph by editing the percentile values in the output table. For example, if the 5% and 95% lines need to be changed to 1% and 99%, simply change the 0.05 to 0.01 and the 0.95 to 0.99 in the Fan Graph output table. Changing the percentile causes Excel to re-draw the graph. An example of a fan graph developed to show the relative risk between three distributions is provided below.
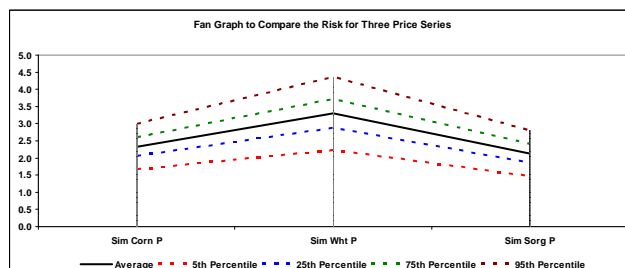
6.6 StopLight Chart

The StopLight chart compares the target probabilities for one or more risky alternatives and is activated by selecting the 🚦 icon. The user must specify two probability targets (*Lower Target* and an *Upper Target*) for the StopLight and the alternative scenarios to compare (Figure 28). The StopLight function calculates the probabilities of: (a) exceeding the upper target (green), (b) being less than the lower target (red), and (c) observing values between the targets (yellow). An example is provided below.
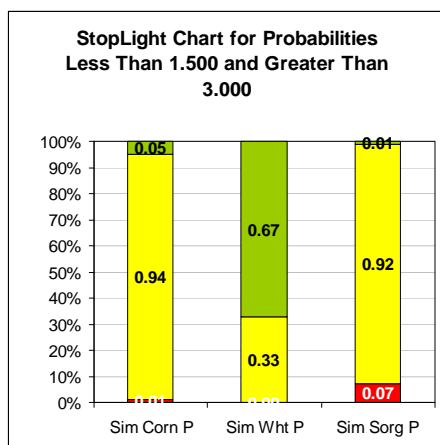
Figure 28. StopLight Dialog Box.

6.7 Probability Plots

Three types of probability plots can be generated by selecting the probability plot icon ![icon]. The probability plot function develops Normal Probability (or NP), Quantile–Quantile (or Q–Q) Plots and Probability–Probability (or P–P) Plots (Figure 29). See Charts Demo.xls for an example of all three types of probability plots.

The Normal Plot is a method for checking how close to normal a random variable is distributed. A Normal Plot compares the ordered data to the standard normal distribution's percentiles. If a variable is normally distributed the sorted data values will be entirely on a straight line with the only deviations from the line due to sampling error. A Quantile-Quantile (Q-Q) Plot can be used to compare two distributions. If the two random variables have the same distribution, their paired observations lie on a 45° line. If the two random variables are in the same family of distributions, their paired observations tend to be linear although they may not lie on the 45° line. A P-P Plot consists of a graph of the percentiles for the sorted values of two variables graphed on one axis. If the two random variables have the same distribution (shape) the observations for a P-P Plot will be on a 45° line.

Figure 29. Probability Plot Dialog Box.

6.8 Box Plots

Box plots of one or more variables can be prepared by selecting the ![icon] icon. The Box Plot dialog box (Figure 30) indicates the information required for this function. The Box Plot is a quartile summary of a random variable in graphical form that indicates whether a variable is skewed to the left or right. The names and values of the Box Plot are best defined in a chart:

where: IQR = [75th Quartile – 25th Quartile]

Fifty percent of the observed values fall within the box (25th to 75th quartile). If the distribution is skewed to the right then the bottom line segment is longer than the top line segment, and vice versa if

Figure 30. Box Plot Dialog Box.

the distribution is skewed left. Values that lie outside the extreme lines are likely to be outliers. The median and mean will show up as one line for symmetrical distributions.
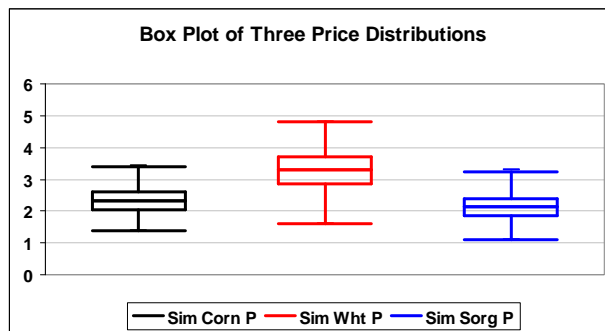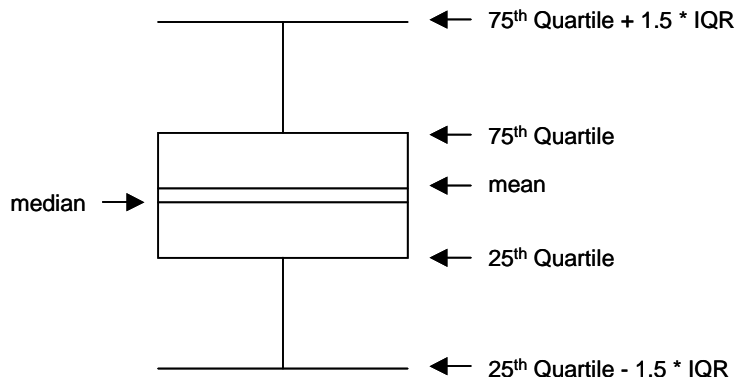
6.9 Scatter Matrix Graph

A scatter matrix of multiple univariate data series can be created using the scatter matrix icon [SCAT] (Figure 31). The scatter matrix is an array of individual graphs of several univariate data series. Each series is plotted against each of the other series, one at a time, like a correlation matrix (see the example below). The graphs show the linear relationships between individual series and can be useful in multiple regression to determine collinearity and for identifying linear relationships between variables for a multivariate probability distribution. See Charts Demo.xls for an example of a scatter matrix.
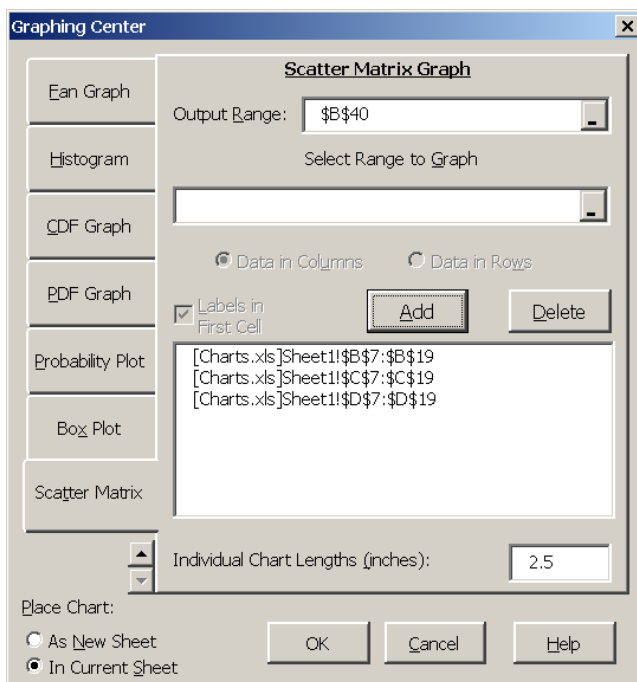
Figure 31. Scatter Matrix Dialog Box.
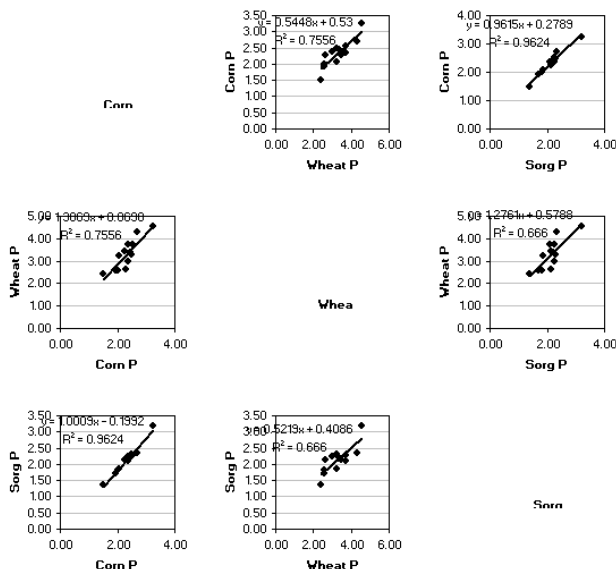
7.0 Scenario Analysis

Simulation models are most useful when used to simulate alternative scenarios. Scenario analysis involves specifying different values for several exogenous or management control variables and simulating the model for the different scenarios. The Simetar Simulation Engine dialog box (Figure 32) provides an input field for entering the *Number of Scenarios*. When the number of scenarios exceeds 1, Simetar executes the =SCENARIO( ) functions in the model.

A separate =SCENARIO( ) function must be specified for each variable to be systematically changed for the alternative scenarios. The =SCENARIO( ) function specifies the values the variable can take on for each scenario. For example, simulating three input variables for example, Hours Product$_i$ for five scenarios (see the example below) is programmed using three =SCENARIO( ) functions as follows:

```
In B21:D25    enter the values for 3 variables for the exogenous variables
In B27        = SCENARIO (B21:B25)
In C27        = SCENARIO (C21:C25)
In D27        = SCENARIO (D21:D25)
```

The values for the first scenario in cells B21:D21 appear in the =SCENARIO( ) after the functions have been entered. During simulation the subsequent scenario values of Hours Product$_{ij}$ (values in rows 22-25) are used, when the Simulation dialog box (Figure 32) is set to simulate 5 scenarios. If the *Number of Scenarios* cell in Figure 32 is set to 1, only the values for the first scenario are used in simulation. The cells containing the =SCENARIO( ) function <u>must</u> be used in the equations of the model for the multiple scenario option to work. For example B27 is used in B30 and B32 below. See Simulate Scenarios Demo.xls for the example provided below.

Figure 32. Scenario Analysis Dialog Box.

	A	B	C	D	E	F
19	Scenario Analysis Table contains the alternative values for each					
20	of the variables across the n scenarios.					
21		Hours Product 1	Hours Product 2	Hours Product 3		
22	Scenario 1	1400	1700	2660		
23	Scenario 2	1200	3060	1500		
24	Scenario 3	2000	1000	2760		
25	Scenario 4	0	2500	3260		
26	Scenario 5	3000	2760	0		
27		Product 1	Product 2	Product 3	Formulas for Product 1	
28	Active Scenario (Hrs)	1400	1700	2660	=SCENARIO(B22:B26)	
29	Stoch Price	2.229	3.119	1.963	=TRANSPOSE(B10:B12)	
30	Stoch Quantity/Hour	87.232	33.395	62.889	=TRANSPOSE(B13:B15)	
31	Receipts	272,266	177,052	328,303	=B29*B30*B28	
32	Resource Cost/Hour	25	35	48	Constants	
33	Net Return	237,266	117,552	200,623	=B31-B32*B28	
34	TNR	555,441	Key Output variable			

The

results of a scenario simulation can be reported to SimData two ways using the *Group Output* option in the Simulation Dialog Box (Figure 32). Grouping the results by *Variable* causes Simetar to present the results in SimData as: Scenario 1-M for Variable 1, then Scenario 1-M for Variable 2, and so on for K output variables. Grouping the results by *Scenario* causes Simetar to present the results as: Variables 1-K for Scenario 1, then Variables 1-K for Scenario 2, and so on for M scenarios. Both formats have their own advantage, use the one which best suits your purpose. It is recommended when using the Scenario option that the *List of Output Variables* include the cells associated with the =SCENARIO( ) functions. This will facilitate verifying that the values in the Scenario Table were appropriately used in the simulation.

The benefit of using the =SCENARIO function is that Simetar runs the model multiple times using <u>exactly the same random deviates</u> (risk) for each scenario. Thus the analysis guarantees that each scenario was simulated using the same risk and the only difference is due to the differences in the scenario variables. The results can be presented as charts and used in risk ranking analyses.
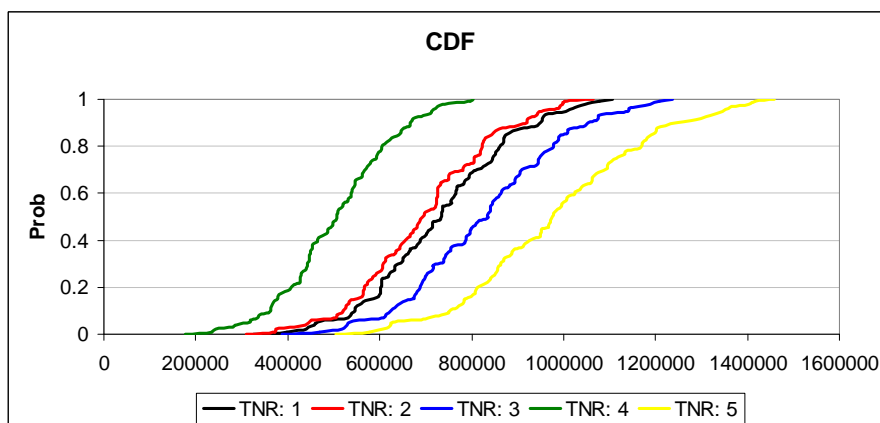
8.0 Sensitivity Analysis

When the *Conduct Sensitivity Analysis* option in the Simulation Engine dialog box is selected, the Simulation Engine dialog box expands to add the sensitivity options in Figure 33. Simetar systematically manipulates one exogenous variable at a time to quantify the sensitivity of the output variables. The *Select Input Variable to Manipulate* cell can refer to any cell in the Excel workbook. The variable to manipulate can be either a constant or a formula. In either case, Simetar uses the initial value as the base and simulates the model using fractional deviations about the base value.
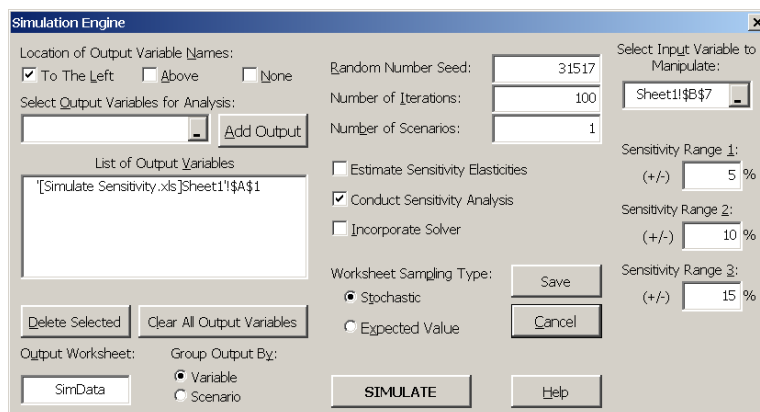
Figure 33. Simulation Sensitivity Dialog Box.

The range of test values for the manipulated input variable are specified using the three *Sensitivity Ranges*. If you are interested in testing the effects of +/- 5, 10, and 15 percent changes in the selected input variable, type these values in the *Sensitivity Range* boxes and

simulate the model. If further investigation shows that the ranges could be +/- 3, 6, and 9 percent, then type in these values and re-simulate the model.

Results of sensitivity analyses are summarized in the SimData worksheet. The results are presented, by output variable, in the following order: the Base value for the Input Variable to Manipulate (or IVM) is 1.0 * IVM, the smallest IMV (say, 0.85 * IVM), the next larger IMV (say, 0.9 * IVM), and so on until the seventh value which is the largest IMV tested (say, 1.15 * IVM). This organization of results facilitates direct comparison of the impacts of the IMV on each of the Output Variables using a Fan Graph.

It is recommended that when sensitivity analyses are being simulated, the list of Output Variables in the Simulation Engine should include the Input Variable to Manipulate. Using this convention, one can easily verify that the Input variable indeed took on the intended values.

9.0 Sensitivity Elasticity Analysis

Figure 34. Estimate Sensitivity Elasticity Option.

The sensitivity of a key output variable (KOV) in a simulation model to changes in several exogenous variables can be measured using sensitivity elasticities (SE_i). A SE_i is like an elasticity, but it quantifies the average percentage change in a KOV to a one percent change in the exogenous variable X. Simetar calculates SE_i values by simulating the model for the base value of each X_i exogenous variable to be tested. Next Simetar changes one X_i at a time by a specified percentage change and simulates the model. The SE_i values are calculated for each X_i value across all iterations and the mean and standard deviation of the SE_i are reported in worksheet SEDATA.

A chart of the SE_i values is provided so the analyst can see which X_i variable has the greatest impact on the KOV. The standard deviation for the SE_i's is displayed in the SE chart as well. To simulate SE values for stochastic simulation model in Excel, select the *Calculate Sensitivity Elasticities* button in the Simulation Engine (Figure 34). This action causes Simetar to expand the Simulation Engine menu to include the inputs for SEs. Select the one KOV to be used for the analysis and select the exogenous variables for which SE's are to be estimated. Specify the percentage change to use for estimating the SE's; 5 percent is usually adequate for this purpose. Simulate the model and review the simulated results in the SimData and SEData worksheets. Edit the SE chart using the Excel chart commands. An example Sensitivity Elasticity chart is

presented in Simulate Sensitivity Elasticities Demo.xls.

10.0 Simulating and Optimization

Stochastic simulation and optimization of a model is complicated because it requires iteratively simulating random shocks to the equations and then optimizing the system. For example, in a two equation supply and demand model with stochastic shocks we would solve for the price that makes demand equal supply or:

$$Q_S = a + b \text{ Price} + cX + (\text{Std Dev} * \text{SND})$$
$$Q_D = a + b \text{ Price} + cY + (\text{Std Dev} * \text{SND})$$
$$E_S = Q_S - Q_D$$

Figure 35. Excel's Solver Dialog Box for Solving an Optimal Control Problem.

If the stochastic shock is zero ($\text{SND} = 0.0$) we simply use Excel's Solver (Figure 35) to solve for the price where ending stocks (E_S) equals zero. See the Sim Solve Demo.xls for an example. Sim Solve Demo.xls demonstrates how a simultaneous equation system can be simulated using the *Incorporate Solver* option in Simetar (Figure 36).

The first step in simulating a stochastic simultaneous equation model is to use Excel's Solver **(Tools > Solver)** to specify the change variable (price, in the example) and the target variable (stocks or E_S, in the example). An example of Excel's Solver dialog box is provided in Figure 35. While the spreadsheet is set to Expected Value, solve the model using Solver, after specifying the Solver parameters, and then open the Simetar Simulation Engine. In the Simulation Engine dialog box select the *Incorporate Solver* option and specify the output variables and simulate the model as usual (Figure

Figure 36. Sim-Solver Dialog Box.

36). It is recommended that the *Output Variables* include the control variable and the target value which Solver is programmed to optimize. This pair of output variables allows one to check Solver's results for each iteration. As should be expected the *Incorporate Solver* option is slow. The reason being that Excel is solving an optimal control problem 100 or more times. Sim-Solver option works well for small models but will not be efficient for large simulation models with numerous (10 or more) simultaneous equations. See the example in Sim Solve Demo.xls.

11.0 Numerical Methods for Ranking Risky Alternatives

The results of a Simetar simulation are written to the SimData worksheet. The results can be analyzed many different ways to help the decision maker determine the most preferred alternative. Functions in Simetar to facilitate analysis of simulation results are described in this section.

11.1 Stochastic Dominance (SD)

11.1.1 First Degree Stochastic Dominance. First degree SD is the least discriminating stochastic dominance method for ranking risky alternatives. However, if the CDFs for the risky alternatives do not cross, this is the preferred method for ranking alternatives. First degree SD can be accessed in Simetar by selecting the 🖉 icon. Select the *1ˢᵗ and 2nd Degree Dominance Table* option Simetar will develop first degree stochastic dominance table (Figure 37). The Stochastic Dominance dialog box (Figure 35) requires the analyst enter the location for the simulated values of the risk alternatives (or scenarios) specify the risk aversion coefficients (RACs). The first degree SD table will be placed in the SD1 spreadsheet. See the example below and in Stochastic Dominance Demo.xls.
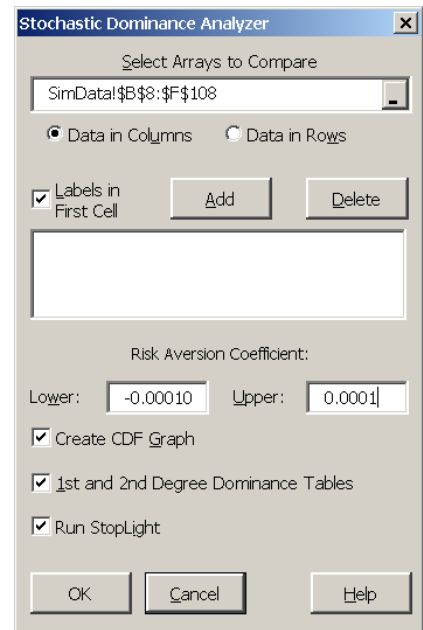
Figure 37. Stochastic Dominance Dialog Box.

11.1.2 Second Degree Stochastic Dominance. Second degree SD assumes the decision maker is risk averse so the RACs must be positive. The 🖉 icon causes Simetar to open the Stochastic Dominance menu (Figure 37) which asks for the simulated values for the risky alternatives and the RACs (say -0.0001 and 0.0001). By selecting the *1ˢᵗ and 2nd Degree Dominance Table* option Simetar will develop a second degree SD output table in the SD1 worksheet. The results for a second degree SD analysis are generally inconclusive. See the example below and in Stochastic Dominance Demo.xls.

First Degree Dominance Table						
	TNR: 1	TNR: 2	TNR: 3	TNR: 4	TNR: 5	
TNR: 1 FDD:		FDD		FDD		The table says that TNR:1 is second degree dominate over the 2 and 4 risky alternatives.
TNR: 2 FDD:				FDD		
TNR: 3 FDD:	FDD	FDD		FDD		
TNR: 4 FDD:						
TNR: 5 FDD:	FDD	FDD	FDD	FDD		The table says that TNR:5 is first degree dominate over the risky alternatives.
Second Degree Dominance Table						
	TNR: 1	TNR: 2	TNR: 3	TNR: 4	TNR: 5	
TNR: 1 SDD:		TNR: 2		TNR: 4		
TNR: 2 SDD:				TNR: 4		
TNR: 3 SDD:	TNR: 1	TNR: 2		TNR: 4		The table says that TNR:3 is second degree dominate over the 1, 2, and 4 risky alternatives.
TNR: 4 SDD:						
TNR: 5 SDD:	TNR: 1	TNR: 2	TNR: 3	TNR: 4		The table says that TNR:5 is second degree dominate over the risky alternatives.

11.1.3 Generalized Stochastic Dominance with Respect to a Function (SDRF). The SDRF option is initiated by selecting the 🖉 icon which opens the dialog box depicted in Figure 37. When specifying the simulation results in *Select Arrays to Compare*, be sure to highlight the label in row one and all of the rows (simulated values) and columns (scenarios or alternatives) to compare. Use the Add button to add scenarios that are not adjacent to the first scenario added in the *Select Array* window. All of the scenarios must have the same number of observations.

The SDRF comparison of risky alternatives uses the *Lower and Upper Risk Aversion Coefficients* (RACs) the user specifies in the dialog boxes (Figure 37). The lower RAC must be less than the upper RAC . No scaling takes place with the user's RAC values. If a RAC is too large in absolute terms (relative to the series to analyze), the STODOM ranking results will show "#VALUE!" rather than ranking each scenario. This result comes about because an exponent overflow is caused by excessively large RACs.

The SDRF results table are written to worksheet SDRF1 (see the example below). The SDRF results table is dynamic so the user can systematically change the RACs in the stochastic dominance results table and observe the effect on scenario rankings. When the SDRF table uses simulation results in the SimData worksheet, the SDRF table will be updated automatically each time Simetar simulates the model.

	A	B	C	D	E	F	G
1		Analysis of Stochastic Dominance with Respect to a Function (SDRF)					
2					© 2005		
3							
4		Efficient Set Based on SDRF at				Efficient Set Based on SDRF at	
5		Lower RAC	0			Upper RAC	0.00001
6		Name	Level of Preference			Name	Level of Preference
7		1 TNR: 5	Most Preferred			1 TNR: 5	Most Preferred
8		2 TNR: 3	2nd Most Preferred			2 TNR: 3	2nd Most Preferred
9		3 TNR: 1	3rd Most Preferred			3 TNR: 1	3rd Most Preferred
10		4 TNR: 2	4th Most Preferred			4 TNR: 2	4th Most Preferred
11		5 TNR: 4	Least Preferred			5 TNR: 4	Least Preferred

11.2 Stochastic Efficiency with Respect to a Function (SERF)

SERF is a new procedure for ranking risky alternatives based on their certainty equivalents (CE) for alternative absolute risk aversion coefficients (ARACs). The CEs for risky alternatives are calculated and the results are presented in a table and a chart by selecting the SERF option in the Simetar toolbar, . The SERF icon opens the SERF Analyzer dialog box (Figure 38). The SERF table and chart are placed in a worksheet named SERFTbl1. The CE values in the table and chart are dynamic so the lower and/or upper ARACs and the utility function can be changed after the dialog box has been run. The SERF procedure defaults to the Exponential Utility Function, yet six more utility functions are available in cell D4 of SERFTbl1. The SERF table values and chart can be calculated assuming a Power Utility Function by typing a "2" in place of the "1" in cell D4. The rule for ranking risky alternatives is that at any given ARAC value, the preferred alternative is the one which is the highest on the Y (or CE) axis. An example of the SERF analysis is available in SERF Analysis Demo.xls.
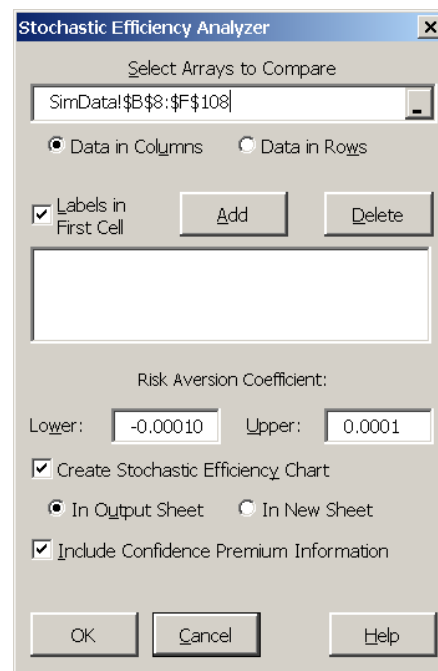

Figure 38. SERF Dialog Box.
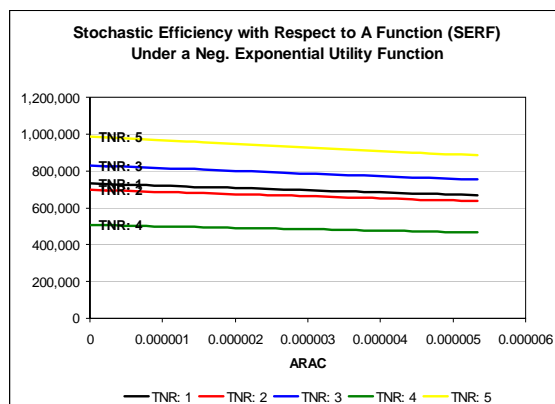
11.3 Risk Premiums

The confidence premium (or the conviction level) with which a decision maker would prefer one alternative over another is visually displayed in the SERF Chart as the vertical distance between the CE lines at each RAC. The SERF analysis also produces a certainty equivalents risk premium (RP) table and chart in the SERFTbl1 worksheet. The RP table compares the absolute differences in the CE's for a base alternative with the other alternatives across RAC values. A chart of the RP's displays the relative position of each alternative to the base over the range of the RACs. The user can change the lower and upper

RACs and the alternative designated as the base. An example of the RP analysis is presented here and in SERF Analysis Demo.xls. The dynamic nature of the SERF option will degrade execution time if the model is re-simulated. If this is a problem, delete the SERFTbl1 Worksheet before re-simulating the model.
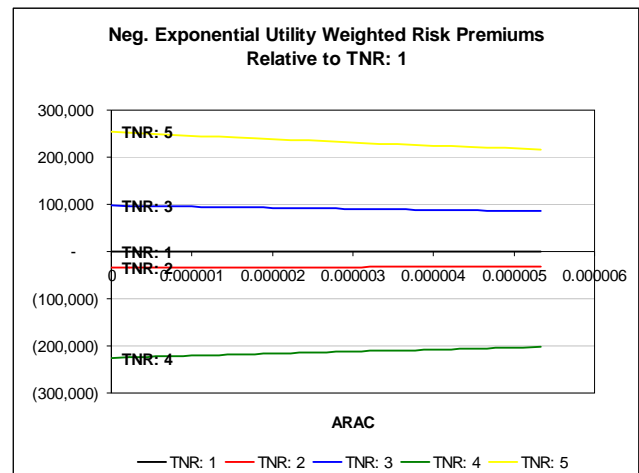
11.4 Target Probabilities for Ranking Risky Alternatives

The probability of a variable taking on a value less than or equal to a specified target value for a simulated distribution can be calculated using the =EDF( ) function in Simetar. Risky alternatives can be ranked with respect to their probabilities exceeding target vales. The =EDF( ) function is programmed as follows:

> = EDF (Array Location, Target Value)
> = EDF (B8:B108, B110)

where: Array Location is the location for the distribution (simulation results) to analyze, and
   Target Value is the location for the target value or an actual number.

An example of how the =EDF( ) function can be used is to first simulate net returns for a business. The probabilities of observing net returns less than particular target values are calculated using =EDF( ). Alternative target values for net returns can be specified by the decision maker. See the Stoplight chart in Section 6.6 for a graphical means of calculating and displaying target probabilities. An sample table of EDF values is presented below from the Simulate Scenarios Demo.xls workbook.

	A	B	C	D	E	F	G	H	I
65	Demonstrate the Quantile and EDF Functions for Analyzing Scenarios of Net Returns.								
66	Prob X<x	TNR: 1	TNR: 2	TNR: 3	TNR: 4	TNR: 5			
67	600,000	17.7%	27.3%	7.8%	77.8%	2.8%	=EDF(SimData!F$9:F$108,Sheet1!$A67)		
68	800,000	68.9%	72.8%	45.1%	99.6%	17.2%	=EDF(SimData!F$9:F$108,Sheet1!$A68)		
69	1,000,000	94.9%	98.2%	85.4%	100.0%	56.4%	=EDF(SimData!F$9:F$108,Sheet1!$A69)		
70									
71	Quantiles	TNR: 1	TNR: 2	TNR: 3	TNR: 4	TNR: 5	=SimData!F8		
72	0.100	543,591	519,786	621,780	360,289	751,556	=QUANTILE(SimData!F$9:F$108,Sheet1!$A72)		
73	0.250	618,722	586,552	700,798	428,431	846,876	=QUANTILE(SimData!F$9:F$108,Sheet1!$A73)		
74	0.500	733,556	693,150	834,415	506,923	975,918	=QUANTILE(SimData!F$9:F$108,Sheet1!$A74)		
75	0.750	846,658	807,150	948,356	592,738	1,117,963	=QUANTILE(SimData!F$9:F$108,Sheet1!$A75)		
76	0.990	1,092,817	1,033,887	1,223,517	799,216	1,437,351	=QUANTILE(SimData!F$9:F$108,Sheet1!$A76)		

11.5 Target Quantiles for Ranking Risky Alternatives

Instead of ranking risky alternatives based on their probability of exceeding a target, some decision makers want to know the target value which has a particular probability of being true, or the quantile for their KOV. This method can be implemented by calculating the value of the key output variable at, say, the 25 percentile. The =QUANTILE( ) function returns the value of a series that is associated with a specified probability. If =QUANTILE( ) is given a series of values, such as, [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] and asked to locate the 35th quantile, then the function returns the value of 3.5 as the 35th quantile value. The array of values to evaluate does not have to be sorted from low to high. An sample table of QUANTILE values is presented above from the Simulate Scenarios Demo.xls workbook. The function is used as:

    =QUANTILE (Array Location, Percentile)
    =QUANTILE(B9:B108,0.56)

where: Array Location is the cell reference for the distribution to be evaluated, and
         Percentile is the percentile to evaluate and is a fraction, such as 0.56.

12.0 Tools for Data Analysis and Manipulation

The Simetar functions developed to facilitate data analysis and manipulation are described in this section. All of the Simetar functions in this section are dynamic so if the historical data for a model or its stochastic variables change, the parameters are automatically updated. This feature is particularly useful when developing simulation models that can use different input data from one application to another. Another feature of Simetar functions is that the formulas are cell locked so the formulas can generally be copied and pasted or dragged to new locations to speed up the data analysis process

12.1 Matrix Operations

Most data in an Excel workbook can be thought of as a matrix. Thirty-three Simetar functions that facilitate the manipulation and analysis of data matrices can be accessed by clicking the [M] icon (Figure 39). The Simetar functions are programmed in C++ and therefore not constrained to Excel's restrictions on array size. The matrix functions are in alphabetical order in the Matrix Operations dialog box:

Center Matrix of a Specified Dimension
Choleski Factorization of a Matrix
Cofactor of a Square Matrix
Column Vector to a Diagonal Matrix
Column Vector to a Matrix
Column Vector to a Toeplitz Matrix
Concatenate Two Matrices
Determinant of a Square Matrix
Eigenvalues of a Square Matrix
Eigenvectors of a Square, Symmetric Matrix
Equicorrelation Matrix of a Specified Dimension
Exponential Power of a Matrix
Factor a Square, Symmetric Matrix
Generalized Inverse of a Matrix
Inner Product of Two Matrices
Invert a Nonsingular Square Matrix
Kronecker Multiply Two Matrices
Mahalanobis Distance of Two Data Matrices
Matrix of 1s
Matrix to a Vector
Multiply Two Matrices
Norm of a Matrix
Orthoganalize a Matrix
Rank of a Matrix
Reduced Row Echelon Form of a Matrix
Reverse a Column or Row of Values
Row Echelon Form of a Matrix
Sequence of Numbers
Sort a Matrix by a Specified Column
Sweep a Square Matrix on a Diagonal Element
Trace of a Square Matrix
Transpose a Matrix
Wishart Matrix of Random Variables
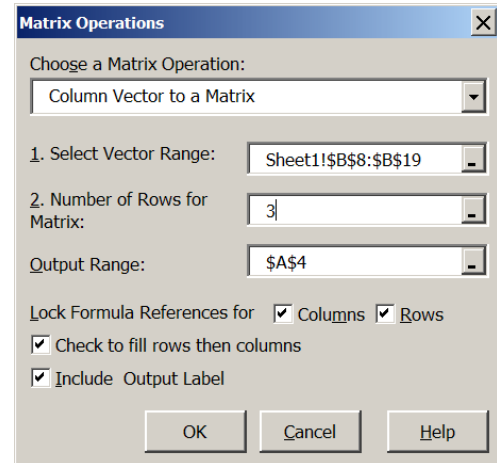
Figure 40. Dialog Box for Changing a Vector to a Matrix.

The most frequently used matrix functions are described in detail in this section. The Simetar Matrix and array functions are dynamic so changes made to the data are automatically observed in the output functions. For example, changes to the input data will change the associated correlation matrix, the Choleski decomposition matrix of the correlation matrix, and subsequent calculations for parameter estimation and stochastic simulation. The matrix functions described in Section 12.0 are demonstrated in the Excel workbook Matrix Operation Functions.xls.

12.1.1 Column Vector to a Matrix. The Matrix Operations dialog box accessed by selecting the [M] icon contains a function to *Change a Column Vector to a Matrix* (Figure 40). The function is dynamic so changes in the original vector are observed in the matrix.

12.1.2 Reverse a Column or Row of Values. A vector of values can be reversed by selecting the *Reverse a Column or Row of Values* in the Matrix Operations menu. The function outputs the data as a column if a column of input is provided and as a row if the input is in a row.
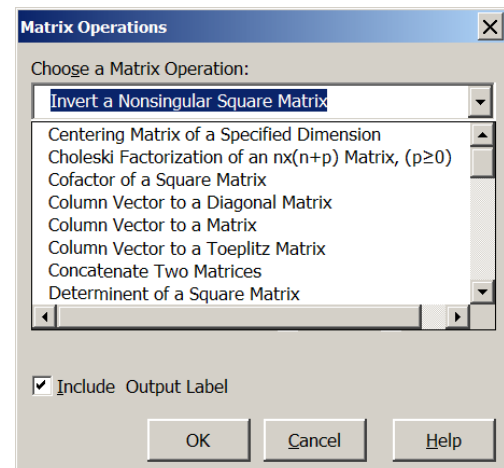
Figure 39. Matrix Operation Menu.

12.1.3 Convert a Matrix to a Vector. The task of converting a matrix of weekly, monthly, or quarterly data to a vector for time series analysis is simplified with the *Matrix to a Vector* function. To use this function indicate the matrix to operate on and the output location for the vector.

12.1.4 Sort a Matrix. An array or a matrix can be sorted in Simetar using the *Sort a Matrix by a Specified Column* in the Matrix Operations menu. The user must specify the *Column to Sort By* as well as the location for the matrix. The sort is dynamic so as the values in the original data matrix change, the values in the sorted matrix will be updated.

12.1.5 Factor a Square Matrix. Simetar can factor a covariance or correlation matrix for simulating a multivariate probability distribution by either the Square Root method or the Choleski method. Both of these methods are accessed via the [M] icon for matrix functions (Figure 41).

12.1.6 Transpose a Matrix (Excel). A matrix can be transposed by selecting the *Transpose a Matrix* option in the Matrix Operations dialog box, specifying the matrix to transpose and the upper-left hand cell to anchor the output matrix. This procedure simplifies Excel's transpose function by eliminating the need to block the area for the transposed matrix and avoids array size limitations in Excel.

12.1.7 Generalized Inverse of a Rectangular Matrix. The *Generalized Inverse of a Matrix* function in the Matrix Operations dialog box uses Simetar's function. Select this option and specify the input matrix (highlight only the numbers) and the output range for the upper left hand value, then select OK. The inverse of the input matrix will appear in the worksheet without row/column names. Copy and paste in the names if needed.

12.1.8 Invert a Nonsingular Square Matrix (Excel). The *Invert a Nonsingular Square Matrix* option in the Matrix Operations dialog box is demonstrated in Figure 42. (Simetar uses Excel's function but provides an easy to use menu.) Select this option and then specify the input matrix (highlight only the numbers) and the output range for the upper left hand value, then click OK. The inverse of the input matrix will appear in the worksheet without row/column names. Copy and paste in the names if needed.

12.1.9 Multiply Two Matrices (Excel). Excel's matrix multiplication, MMULT, function is made easier by selecting the *Multiply Two Matrices* option in the Matrix Operations dialog box. An additional feature is that Simetar's matrix multiplication will handle larger matrices than the Excel function MMULT.

12.1.10 Concatenate Two Matrices. A new matrix of data can be developed by concatenating the data from two
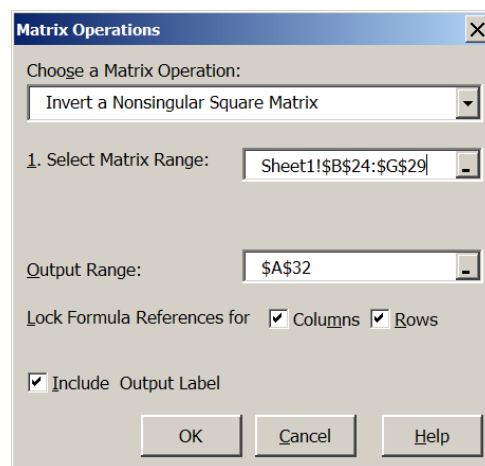
Figure 42. Invert a Square Matrix Dialog Box.
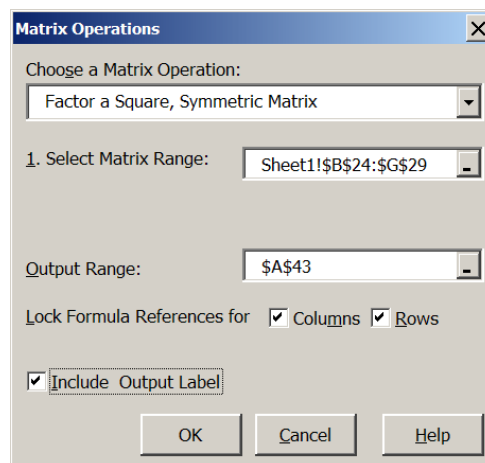
Figure 41. Factor a Square Matrix Dialog Box.

locations in the workbook. The *Concatenate Two Matrices* option in the Matrix Operations menu requires as input the location of the two input arrays or matrices and the output location.

12.1.11 Convert a Vector to a Diagonal Matrix. In simulation it is useful to convert a vector of standard deviations to a diagonal matrix. The Simetar function =MDIAG( ) can be used to convert an array to a diagonal matrix using the *Column Vector to a Matrix* option in the Matrix Operations dialog box.

12.1.12 Find the Determinant of a Square Matrix. The determinant of a square matrix can be calculated by selecting the *Determinant of a Square Matrix* option in the Matrix Operations dialog box. The Excel function =MDETERM (square matrix) is used for this calculation.

12.2 Data Manipulation

Data often comes in the wrong format or orientation. Data may be in an array when we need it in a matrix or vice versa. Sometimes we need to reverse the order of the data or concatenate arrays from different places in the worksheet. Functions to make these data manipulations easy have been included in Simetar and can be accessed by selecting the **M** icon. Additional data manipulation functions are also presented in this section.

12.2.1 Create an Identity Matrix. An indemnity matrix of dimension NxN can be generated using the =MIDEN( ) function in Simetar. The format for the function is =MIDEN (dimension) where dimension is a scalar to specify the number of rows in the square identity matrix.

11.2.2 Create a Sequence of Numbers. A sequence of numbers in an array can be created using the =SEQ( ) function. The =SEQ( ) returns a column of numbers that follow any sequence you specify. The function is programmed as:

=SEQ(No. of Values, Starting Value, Interval or Increment)

where: No. of Values is the number of cells to be highlighted,
Starting Value is the first value in the sequence, and
Interval or Increment is the interval between each value.

For example the sequence of number for 10, 20, 30, …, 200 is generated by programming the function as =SEQ(20,10,10) and a sequence of 2, 4, 6, …, 20 is generated by programming the function as =SEQ(10, 2, 2).

11.2.3 Create a Matrix of Ones. In statistics a J matrix is an array or matrix with a 1.0 in each cell. The Simetar function =MJ( ) is used to create a J matrix. To create a 10x1 array of 1.0s highlight 10 cells in column and type the function =MJ(10). To create a 10x10 matrix of 1.0s, highlight a 10 cells in 10 columns and type =MJ(10,10). Be sure to hit the Control Shift, Enter keys after typing the =MJ( ) function as it is an array function.

11.2.4 Create a Centering Matrix. The =MCENTER() array function that creates an NxN centering matrix when n is specified as the dimension.

11.2.5 Create an Equicorrelation Matrix. The =MEQCORR() array function generates an NxN equicorrelation matrix using any specified correlation coefficient. The =MEQCORR( ) function is an array function so you must highlight the cells for the square equicorrelation matrix and end the function by hitting *Control Shift Enter*. The function is programmed as =MEQCORR(Rho), where Rho is the correlation coefficient.

11.2.6 Create a Toeplitz Matrix. The =MTOEP() array function creates a square symmetric Toeplitz matrix given a column or row of data. To create a Toeplitz matrix of an array in A1:A4, highlight a 4X4 array and type the function as =MTOEP(A1:A4). Be sure to press *Control Shift Enter* as this is an array function.

12.3 Box-Cox Transformation

The =BOXCOX() function can be used to transform the data for a skewed distribution to make it approximately normally distributed. The function uses a user specified exponent to transform the data. The =BOXCOXEXP() function is provided to estimate an appropriate exponent. The format for the Box Cox transformation functions are:

=BOXCOX(Data Array, Power Value, [Shift to Plus])

where: Data Array refers to the location of the Nx1 data series to be transformed,
Power Value is the exponent for the transformation, and
Shift to Plus is an optional term if the data are to be shifted to positive values enter
'TRUE or 1', otherwise enter 'FALSE or 0'.

The =BOXCOX( ) function is an array function so highlight the appropriate number of cells and type the function and press *Control Shift Enter*. See Data Analysis Demo.xls for an example. Once a model has been estimated using a Box-Cox transformation, the =UNBOXCOX() function can be used to transform the forecast values back to original data. The reverse Box-Cox transformer function is:

=UNBOXCOX (Data Array, Power Value, Original Data Array, [Shift to Plus])

where: Data Array is the location for the Nx1 array transform back to the original data,
Power Value is the exponent for the transformation,
Original Data Array is the location for the original data Nx1 array, and
Shift to Plus is an optional term if the data are to be shifted to positive values enter
'TRUE or 1', otherwise enter 'FALSE or 0'.

The maximum likelihood estimation of the Box-Cox transformation exponent function can be calculated using the following function:

=BOXCOXEXP(Data Array, [Shift to Plus], [Lower], [Upper], [Max Iter])

where: Data Array refers to the location of the data n-1 array to be transformed,
Shift to Plus is an optional term if the data are to be shifted to positive values,
Lower is an optional minimum for the search routine, -2 is the default,
Upper is an optional maximum for the search routine, +2 is the default, and

Max Iter is an optional parameter for the search routine.

12.4 Workbook Documentation

12.4.1 Delete Numbers in a Cell. When a cell has both numbers and text, to extract only the text, use the =DELNUM( ) function. See Data Analysis Demo.xls for an example. If cell A1 contains the string "1013 Sycamore Street" and we want the text in cell A2, then in A2 type:

=DELNUM(A1)

12.4.2 Delete Text in a Cell. Often times the numbers in a cell are needed even though the cell contains both numbers and text. For example, the worksheet may have an address in a cell as "1013 Sycamore Street" and we want the number without the text. Rather than re-typing the numbers to a new cell or editing the existing cell use the =DELTEXT( ) function. See Data Analysis Demo.xls for an example. Say the cell A1 has the string "1013 Sycamore Street" and you want just the number to appear in cell B1, then in B1 type:

=DELTEXT (A1)

12.4.3 View Cell Formulas. To show the formula typed in a particular cell use =VFORMULA( ). An advantage of using this function is that you can both see the formula for a cell, say B24, and you can see the value in B24. The =VFORMULA( ) function is dynamic and changes (updates itself) as rows and columns are added to or removed from the worksheet. The Simetar function to view the formula in cell B24 can be typed into any cell (say, C24) as follows:

= VFORMULA (B24)

12.4.4 View All Formulas. In the process of writing and documenting simulation models in Excel we often write formulas that need to be printed. Simetar provides a function to easily view every cell in the worksheet as a formula, and then switch the worksheet back to values. This function can be accessed by clicking the ▪EQ icon in the Simetar toolbar. Click the ▪EQ icon a second time and the worksheet will return to the normal view.

12.4.5 Workbook and Worksheet Name. Functions in Simetar have been provided to dynamically show the name of the workbook or the worksheet in a cell. These functions are useful for documenting a model. The workbook name is shown in any cell that contains the following command:

=WBNAME( )

The worksheet name is shown in any cell that contains the following command:

=WSNAME( )

If you rename the workbook or the worksheet, the function updates the text in the cell after pressing F5.

13.0 Regression Analysis
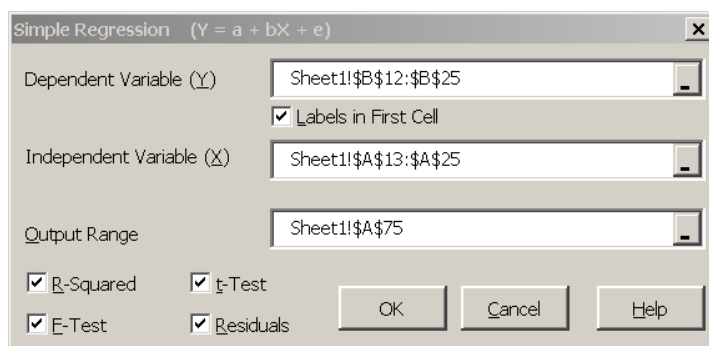
Simple and multiple regression (ordinary least square (OLS), Probit, Logit, GLS, Ridge, 2SLS, and GLS) capabilities are included in Simetar to facilitate estimating parameters for simulation models. Not only are the regression coefficients (beta-hats) useful, but in simulation the residuals are used to quantify the unexplained risk for a random variable. The regression functions in Simetar take advantage of Excel's ability to recalculate all cells when a related value is changed. Thus when an observed X or Y value is changed the betas are recalculated. Also, multiple regression models can be instantly re-estimated for different combinations of the X variables by using restriction switches to ignore individual variables.

13.1 Simple Regression

The parameters for a simple OLS regression are calculated when you select the ŷ-bx̂ icon. The simple regression icon opens the dialog box depicted in Figure 43 so the X and Y variables can be specified. The intercept $(\hat{a})$ and slope $(\hat{b})$ parameters for the equation:

	A	B	C
74	Simple Regression of Y = a + b X		
75		Corn P	
76	Intercept	1.98	
77	Slope	0.049231	
78	R-Square	0.211367	
79	F-Ratio	2.948195	
80	Prob(F)	0.11396	
81	S.E.	0.028672	
82	T-Test	1.717031	

$$\hat{Y} \ = \ \hat{a} \ + \ \hat{b} \, X$$

Simple Regression   (Y = a + bX + e)

Dependent Variable (Y)   Sheet1!$B$12:$B$25

☑ Labels in First Cell

Independent Variable (X)   Sheet1!$A$13:$A$25

Output Range   Sheet1!$A$75

☑ R-Squared   ☑ t-Test
☑ F-Test   ☑ Residuals

OK   Cancel   Help

Figure 43. Simple Regression Dialog Box.

are estimated and placed in the worksheet starting where the *Output Range* specifies. The names of the estimated parameters appear in the column to the left of the parameters. The R^2, F-Ratio, Student's -t test statistics, and residuals are calculated if you select the appropriate boxes.

Be sure that X and Y have the same number of observations when you specify their ranges in the Simple Regression dialog box. This Simetar function is useful for checking the presence of a trend in a random variable Y. In this case, create a column of X values that increment from 1, 2, 3, ..., N and then use Simetar to estimate the regression parameters. A feature to this function is that the coordinates for the X variable are cell reference locked (fixed) so the formula cells can be copied and pasted across the spreadsheet to estimate simple regressions for numerous Y's using a common X or trend variable. An example of the simple regression function in Simetar is provided below and in the Data Analysis Demo.xls workbook.

13.2 Multiple Regression

The Multiple Regression option is accessed through the R²ₑⱼ icon. Multiple regression estimates the least squares $\hat{a}$ and $\hat{b}_i$ parameters for:

$$\hat{Y} = \hat{a} + \hat{b}_1 X_1 + \hat{b}_2 X_2 + \ldots + \hat{b}_n X_n$$

The Multiple Regression dialog box (Figure 44) allows the user to specify the Y and X variables, and the type of output for seven different multiple regression models.

A sample output for a multiple regression is provided below to show the format for the first part of the results. The name of an X variable and its beta are in bold if the variable is statistically significant at the indicated one minus alpha level (e.g., X_1, X_2, X_3, and X_4 in the example). Standard errors for the betas, the t-test statistics and the probability (p) value of the t-statistics are provided for each explanatory variable. The elasticity at the mean for each independent variable as well as the partial and semi-partial correlations for these variables is provided as well. The variance inflation factor is reported for each X variable to indicate the degree of multicolinearity of X_i to other variables in the model. See Multiple Regression Demo.xls for the example presented in this section.

The Restriction row in the parameter block of output values
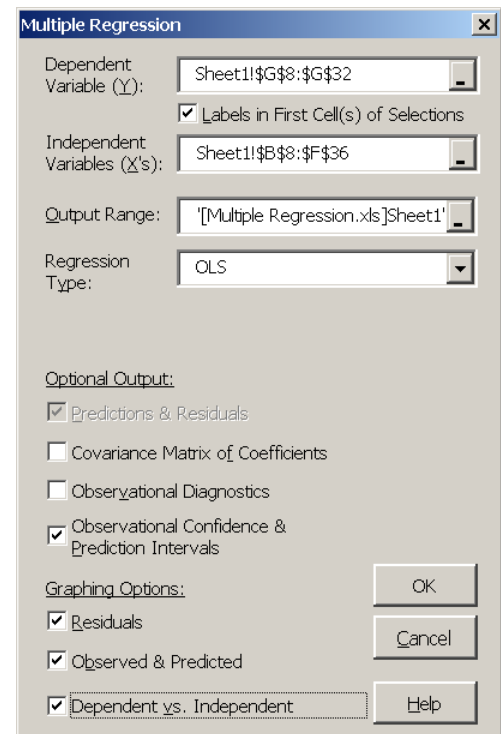
Figure 44. Multiple Regression Dialog Box.

Left table:

A	B	C	D	E	F	G
40 OLS Regression Statistics for Y, 11/18/2005 10:53:10 AM						
41 F-test	88.156	Prob(F)		0.000 Unrestricted Model		
42 MSE^1/2	0.011	CV Regr		0.204 F-test	88.156	
43 R^2	0.961	Durbin-Watso		1.768 R^2	0.961	
44 RBar^2	0.950	Rho		-0.006 RBar^2	0.950	
45 Akaike Infor	-8.883	Goldfeld-Qua		1.618 Akaike Inform	-8.883	
46 Schwarz Infc	-8.638			Schwarz Infor	-8.638	

47 95%	Intercept	X1	X2	X3	X4	X5
48 Beta	0.417	-0.156	0.298	-0.014	0.611	-0.013
49 S.E.	0.776	0.042	0.109	0.003	0.097	0.047
50 t-test	0.537	-3.734	2.730	-4.302	6.307	-0.268
51 Prob(t)	0.598	0.002	0.014	0.000	0.000	0.792
52 Elasticity at Mean		-0.077	0.436	-0.035	0.612	-0.014
53 Variance Inflation Factor		18.155	67.613	98.155	4.997	4.105
54 Partial Correlation		-0.661	0.541	-0.712	0.830	-0.063
55 Semipartial Correlation		-0.174313628	0.127443618	-0.200861965	0.294466122	-0.012500054
56 Restriction						

Right table:

A	B	C	D	E	F	G
40 OLS Regression Statistics for Y, 11/18/2005 10:53:10 AM						
41 F-test	115.837	Prob(F)		0.000 Unrestricted Model		
42 MSE^1/2	0.011	CV Regr		0.199 F-test	88.156	
43 R^2	0.961	Durbin-Watso		1.764 R^2	0.961	
44 RBar^2	0.952	Rho		0.004 RBar^2	0.950	
45 Akaike Infor	-8.963	Goldfeld-Qua		1.378 Akaike Inform	-8.883	
46 Schwarz Infc	-8.766			Schwarz Infor	-8.638	

47 95%	Intercept	X1	X2	X3	X4	X5
48 Beta	0.370	-0.153	0.288	-0.014	0.620	0.000
49 S.E.	0.737	0.039	0.099	0.003	0.089	0.000
50 t-test	0.502	-3.910	2.898	-4.547	6.946	0.000
51 Prob(t)	0.622	0.001	0.010	0.000	0.000	1.000
52 Elasticity at Mean		-0.076	0.421	-0.034	0.621	0.000
53 Variance Inflation Factor		16.708	58.635	88.872	4.454	4.105
54 Partial Correlation		-0.668	0.554	-0.722	0.847	0.000
55 Semipartial Correlation		-0.178026492	0.131961938	-0.207052399	0.316282667	0
56 Restriction						0

allows the user to interactively experiment with various combinations of X variables. After the initial parameter estimation the Restriction coefficients are all blank, meaning that every X variable is included in the unrestricted model. The user can interactively drop and re-include a variable by changing its restriction coefficient from blank to 0. Compare the results in the first example to those in the second example where X_5 was restricted out of the model. The exclusion of X_5 improves the F –test (61.5 vs. 79.2). Three test statistics (F, R^2 and $\overline{R}^2$) for the Unrestricted Model are provided and remain fixed while testing alternative specifications of the model's variables. This is done to facilitate the comparison to the original unrestricted model to the restricted models. If you type a non-zero number in the restriction row, the value becomes the beta-hat coefficient for a restricted regression.

In addition to the ability to exclude and re-include variables in the model, Simetar's multiple regression function allows the analyst to make corrections to the data for the actual observations of the X and Y values, without having to re-run the regression. The Simetar multiple regression routine is not limited in the number of exogenous variables that can be included in the model.

Regression models with 5000 observations and 250 X variables can be estimated with Simetar.

If the analyst specifies more observations for the X variables than for the Y variable, Simetar will forecast the Y values. The forecast values in the "Predicted Y" column of the output uses the betas for the regression and the additional Xs. Probabilistic forecasts of the Y variable are provided as bold values in the Actual Y column of the output. For the example, there are five

	A	B	C	D	E	F	G	H	I	J
65	S.D. Resids	0.009561383	MAPE	0.14581097		Lower	Upper	Lower	Upper	Scatter Plot of A
66	Actual Y	Predicted Y	Residuals	SE Mean Pre	SE Predicted	95% Conf. Int	95% Conf. Int	95% Predict.	95% Pred	X1
67	5.536	5.524	0.012	0.008	0.014	5.507	5.542	5.495	5.554	2.850
68	5.511	5.524	-0.013	0.006	0.013	5.511	5.537	5.497	5.551	2.934
69	5.465	5.479	-0.014	0.006	0.013	5.466	5.492	5.453	5.505	3.011
70	5.467	5.454	0.013	0.004	0.012	5.446	5.463	5.429	5.479	2.818
71	5.465	5.447	0.018	0.004	0.012	5.438	5.456	5.422	5.471	2.962
72	5.452	5.454	-0.002	0.004	0.012	5.445	5.463	5.429	5.479	2.890
73	5.438	5.445	-0.007	0.005	0.012	5.435	5.455	5.420	5.470	2.888
74	5.431	5.421	0.010	0.005	0.012	5.410	5.431	5.395	5.446	2.910
75	5.409	5.407	0.002	0.005	0.012	5.397	5.417	5.382	5.433	2.854
76	5.386	5.388	-0.002	0.006	0.012	5.376	5.400	5.362	5.414	2.820
77	5.359	5.372	-0.013	0.008	0.012	5.360	5.385	5.346	5.399	2.754
78	5.358	5.359	-0.001	0.006	0.013	5.347	5.371	5.333	5.385	2.721
79	5.367	5.372	-0.005	0.006	0.013	5.359	5.385	5.345	5.398	2.663
80	5.385	5.383	0.002	0.005	0.012	5.372	5.394	5.357	5.409	2.599
81	5.388	5.398	-0.010	0.004	0.012	5.388	5.407	5.373	5.423	2.558
82	5.394	5.393	0.001	0.005	0.012	5.383	5.404	5.368	5.419	2.547
83	5.403	5.406	-0.003	0.005	0.012	5.396	5.417	5.381	5.432	2.485
84	5.409	5.400	0.009	0.005	0.012	5.388	5.412	5.375	5.426	2.512
85	5.401	5.393	0.009	0.006	0.012	5.380	5.405	5.366	5.419	2.526
86	5.398	5.403	-0.005	0.007	0.013	5.389	5.417	5.376	5.430	2.340
87	5.378	5.386	-0.008	0.004	0.012	5.377	5.386	5.361	5.411	2.396
88	5.368	5.371	-0.004	0.005	0.012	5.361	5.382	5.348	5.397	2.338
89	5.349	5.356	-0.007	0.006	0.013	5.343	5.369	5.330	5.383	2.338
90	5.368	5.348	0.020	0.006	0.012	5.337	5.360	5.322	5.374	2.269
91	**5.367**	5.335		0.008	0.014	5.317	5.353	5.305	5.364	2.374
92	**5.346**	5.348		0.007	0.013	5.333	5.362	5.320	5.375	2.250
93	**5.306**	5.323		0.008	0.014	5.304	5.343	5.293	5.353	2.337
94	**5.261**	5.299		0.008	0.014	5.279	5.318	5.268	5.329	2.329

extra X values indicated for the regression dialog box (Figure 44) so Simetar calculated the deterministic forecast values in column B and the probabilistic forecast values in column A, starting in row 91 (see the output above). Probabilistic forecasts are calculated assuming normality, the mean equals the deterministic forecast, and the standard deviation is the standard error of the predicted Y in column E for the example. Press F9 to make Excel simulate the probabilistic forecasts. The probabilistic forecasts can be used in a stochastic simulation model.

Observed and Predicted Values for Y

Residuals for the regression are also included in the example output. The residuals for the regression are calculated as $\hat{e}_i = Y_i - \hat{Y}_i$ for each observation i and represent the unexplained risk for the dependent Y variable. The standard error for the mean predicted value (SE mean predicted) is provided for each observation i. In addition the SE of the Predicted Y for each observation is provided in column E of the example output. As indicated in the example output, the SE Predicted Values increase as the forecasted period gets longer. Prediction and confidence intervals for the model are provided in the table (above) and graphically (below) for the alpha equal 5 percent level. The alpha level can be changed by changing the value in line 47 of the output example from 95% to, say, 90% or 99%.

The observed and predicted Y values can be viewed graphically along with the confidence and prediction intervals. For the example program five more Xs than Ys were used to estimate the model, as a result the last five values in the Observed line to the right are the probabilistic forecast values and will change each time the F9 function key is pressed.

The covariance matrix for the betas is an optional output for multiple regressions. The beta covariance matrix is used in simulation when the model is assumed to have stochastic betas. The beta covariance matrix is provided when specified as an option in the multiple regression dialog box (Figure 44).

If requested in the regression dialog box (Figure 44), observational diagnostics are calculated and reported for the unrestricted model (see the example to the right). The column of 1's in the DFBetas Restriction column indicate that the unrestricted model was fit using all of the observed data.

If you change a DFBetas Restriction to 0 for a particular row the model is instantly updated using a dummy variable to ignore the effects for that row of X's and Y. The rule for excluding

	G	H	I	J	K	L M N O P Q R S
13	Probit Regression Statistics for Stocks Over 250. 11/18/2005					
14	95%	Intercept	Price t	Stock t-1	Trend	
15	Beta*	14.988	-2.218	0.004	-0.261	
16	S.E.*	6.938	0.990	0.007	0.150	
17	t-test*	2.160	-2.240	0.592	-1.746	
18	Prob(t)*	0.044	0.037	0.561	0.097	
19	Elasticity	8.125	-6.982	0.652	-1.558	
20	Derivative	5.435	-0.804	0.002	-0.095	
21	Restriction	1	1	1	1	
22	Predictions & Residuals					
23	S.D. Residual	0.298858579		Observational		
24	Actual Stc	Predicted Sto	Residuals	Restriction	Observati	
25	1.000	0.995	0.005	1	1	
26	1.000	0.998	0.002	1	2	

Observed and Predicted Values for Stocks Over 250

×Predicted ●Observed

an observation is if its Studentized Residual is greater than 2 (is bold). This is the case for observation 24 in the sample output. Setting the Restriction value to 0 for observation 24 causes the F statistic to increase from 88 to 107, given that X_5 has not been excluded from the model. The R^2 increases to 96.1 from 96.9 (see Multiple Regression Demo.xls). This result suggests that observation 24 is either an outlier or should be handled with a dummy variable. A priori justification should be used when handling observations in this manner.

13.3 Bivariate Response Regression

13.3.1 Probit Analysis. The PROBIT regression function estimates a logistic regression given dependent and independent variables. Probit regression models can be estimated by using the multiple regression icon R²₉ and selecting the Probit option in the menu, see Figure 44 for the menu. The PROBIT function allows for independent variables to be restricted from the complete model (enter '0' in place of the '1').

In addition, individual observations can be restricted from the regression (enter '0' in place of '1'). The PROBIT Function uses an iteratively re-weighted least squares technique
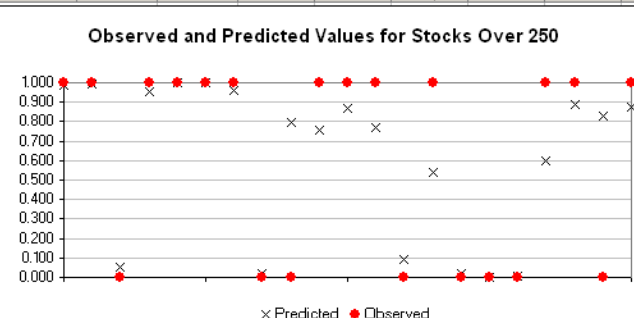
	A	B	C	D	E	F	G	H	I	J	K	L	M
95	Observational Diagnostics								Cutoff Values		0.250		
96	DFBetas		0.408	0.408	0.408	0.408	0.408	0.408	0.500	2.000	1.750	1.000	
97	Restriction	Observation	Intercept	X1	X2	X3	X4	X5	Leverage	Studentized Re:	Covariance	DFFits	DFFit
98	1	1	0.594	-1.593	-0.324	-0.445	0.102	-0.132	0.589	1.710	1.322	2.045	0.016
99	1	2	0.504	0.122	-0.108	0.158	-0.461	-0.280	0.326	-1.494	0.997	-1.038	-0.006
100	1	3	0.311	-0.515	0.399	-0.549	-0.734	-0.443	0.297	-1.597	0.886	-1.038	-0.008
101	1	4	0.205	-0.121	-0.213	0.078	0.021	0.093	0.140	1.290	0.936	0.521	0.002
102	1	5	-0.215	0.469	-0.128	0.250	0.255	0.297	0.149	1.922	0.509	0.604	0.003
103	1	6	0.008	0.002	-0.032	0.025	0.012	0.048	0.155	-0.173	1.651	-0.074	0.000
104	1	7	0.203	-0.123	-0.217	0.149	0.063	0.013	0.176	-0.711	1.433	-0.329	-0.002
105	1	8	-0.293	0.441	0.119	0.045	0.041	0.192	0.224	1.056	1.240	0.566	0.003
106	1	9	0.028	0.032	-0.025	0.040	0.009	-0.037	0.192	0.178	1.724	0.087	0.000
107	1	10	-0.065	-0.020	0.039	-0.047	0.010	0.057	0.267	-0.188	1.899	-0.114	-0.001
108	1	11	-0.714	0.003	0.599	-0.464	0.104	0.000	0.274	-1.427	0.994	-0.877	-0.005
109	1	12	-0.074	0.017	0.016	0.001	0.063	0.032	0.287	-0.155	1.959	-0.098	-0.001
110	1	13	-0.039	0.031	-0.090	0.155	0.253	-0.129	0.304	-0.551	1.822	-0.365	-0.002
111	1	14	0.012	-0.033	0.041	-0.063	-0.078	0.011	0.223	0.187	1.791	0.100	0.001
112	1	15	0.049	0.156	-0.260	0.315	0.276	0.037	0.153	-0.968	1.205	-0.411	-0.002
113	1	16	-0.011	0.003	0.006	-0.007	-0.004	0.021	0.194	0.062	1.745	0.031	0.000
114	1	17	0.053	0.066	-0.139	0.138	0.083	0.079	0.223	-0.330	1.744	-0.177	-0.001
115	1	18	-0.239	0.052	0.338	-0.200	-0.051	-0.283	0.228	0.926	1.360	0.504	0.003
116	1	19	-0.291	0.269	0.192	0.044	0.167	-0.218	0.278	0.918	1.460	0.570	0.003
117	1	20	-0.126	0.266	-0.014	0.052	-0.009	0.300	0.381	-0.578	2.026	-0.454	-0.003
118	1	21	0.090	-0.060	0.057	-0.133	-0.205	0.004	0.165	-0.820	1.338	-0.365	-0.002
119	1	22	-0.013	0.006	0.094	-0.089	-0.089	-0.078	0.209	-0.358	1.702	-0.184	-0.001
120	1	23	0.015	-0.122	0.269	-0.294	-0.256	-0.307	0.309	-0.731	1.694	-0.489	-0.003
121	1	24	0.308	-0.152	-0.700	0.566	0.362	0.625	0.258	2.344	0.354	1.381	0.007

to estimate the model parameters. A sample Probit output for Simetar from the Probit and Logit Demo.xls is summarized below.

13.3.2 Logit Analysis. The LOGIT function estimates a logistic regression given dependent and independent variables. Logit regression models can be estimated by using the multiple regression icon and selecting the Logit option in the menu, see Figure 44 for the menu. The LOGIT function allows for independent variables to be restricted from the complete model. In addition, individual observations can be restricted from the regression. The LOGIT function uses an iteratively re-weighted least squares technique to estimate the model parameters. A sample Logit output for Simetar is presented below from the Probit and Logit Demo.xls.

	G	H	I	J	K
65	Logit Regression Statistics for Stocks Over 250, 12/27/2005				
66	95%	Intercept	Price t	Stock t-1	Trend
67	Beta*	25.383	-3.741	0.007	-0.439
68	S.E.*	13.123	1.861	0.013	0.274
69	t-test*	1.934	-2.011	0.521	-1.602
70	Prob(t)*	0.068	0.059	0.608	0.126
71	Elasticity	8.139	-6.965	0.616	-1.550
72	Derivative	5.529	-0.815	0.001	-0.096
73	Covarianc	1	1	1	1
74		Intercept	Price t	Stock t-1	Trend
75	Intercept	172.206	-23.245	-0.010	-3.173
76	Price t	-23.245	3.462	-0.005	0.421
77	Stock t-1	-0.010	-0.005	0.000	-0.001
78	Trend	-3.173	0.421	-0.001	0.075

Observed and Predicted Values for Stocks Over 250

14.0 Cyclical Analysis and Exponential Forecasting

Functions to facilitate analysis of seasonal and cyclical data are included in Simetar. Seasonal indices and moving average analysis of cyclical data are described in this section. Three different procedures for developing exponential forecasts included in Simetar are described as well.

14.1 Seasonal Index

A seasonal index of any array can be calculated by Simetar using the Forecasting and Cyclical Data icon ![icon] and clicking on the *Seasonal Indexing* tab. The Seasonal or Cyclical Indexing dialog box (Figure 45) allows the user to specify the data series to analyze and the number of periods in the cycle, (say, 4 or 8 or 12). A sample output table is presented below and in Seasonal Analysis Demo.xls.

When the input data are months and the *Number of Periods in the Cycle* is 12 the result will be a 12 month seasonal index. The quarterly index in the example below is developed from five years of quarterly sales to calculate a seasonal sales index.

A seasonal index can be calculated one of two ways, namely: simple average or centered moving average. The simple average index is a more reliable indicator of the seasonal pattern if the data has no trend. If the data series has an underlying trend the *Centered Moving Average* will remove a portion of the variability caused by the trend. The Seasonal and Cyclical Indexing dialog box (Figure 45) assumes the user wants a simple average index.
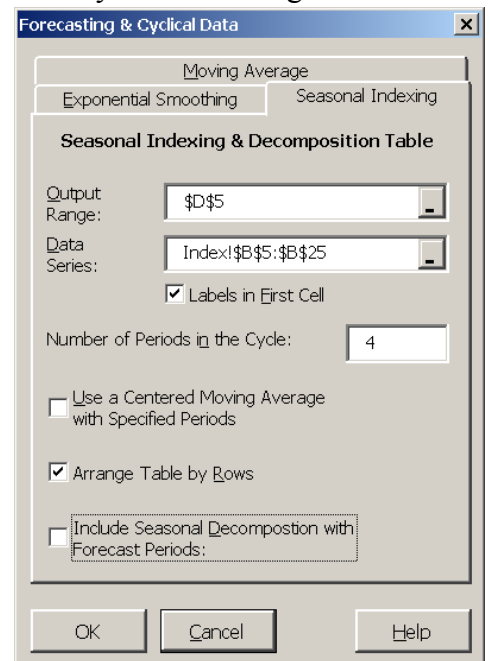
Figure 45. Seasonal or Cyclical Indexing Dialog Box.

	D	E	F	G	H	I	J	K	L	M
5	Cyclical Indices for Sales ($1,000s), 11/18/2005 12:28:26 PM									
6		1	2	3	4	SUM	AVERAGE	ST DEV	INDEX	FRAC. CO
7	1	25.14	15.12	21.41	30.14	91.81	22.9525	5.481165	0.884	0.177
8	2	26.71	18.91	24.15	30.21	99.98	24.995	4.119427	0.962	0.192
9	3	26.12	17.45	23.36	35.61	102.54	25.635	6.555641	0.987	0.197
10	4	27.3	14.89	31.28	38.99	112.46	28.115	8.715614	1.082	0.216
11	5	24.91	16.76	29.67	41.45	112.79	28.1975	8.936178	1.085	0.217
12	SUM	130.18	83.13	129.87	176.4	519.58				
13	AVERAGE	26.036	16.626	25.974	35.28		25.979			
14	ST DEV	0.908815	1.49635	3.815865	4.562157			7.287482		
15	INDEX	1.002	0.640	1.000	1.358				4.0 \ 5.0	
16	FRAC. CONT. INDEX	0.251	0.160	0.250	0.340					1
17	INDEX LCI	0.934	0.464	0.712	1.105					
18	INDEX UCI	1.071	0.816	1.288	1.611		Conf. Level		95%	1.959964

14.2 Seasonal Decomposition Forecasting

A Seasonal Decomposition forecast of a data series can be calculated by Simetar using the Forecasting and Cyclical Data icon and clicking on the *Seasonal Indexing* tab (Figure 46). After indicating where the data series is located and the number of periods in the cycle, click on the last box in the menu to *Include Seasonal Decomposition with Forecast Periods.* This will cause Simetar to calculate the parameters for a seasonal decomposition forecast for the number of periods indicated in the last window of the dialog box, four for the example presented below.

The output for the seasonal decomposition forecast contains two switches that allow the user to alter the type of decomposition model that best fits the data series being forecasted. The options are Additive and Cycle (see the example output below). The default value for the ADDITIVE option, "TRUE", is for an additive model which assumes the seasonal component is additive. If the seasonal effects are multiplicative, use the "FALSE" setting for the ADDITIVE option. The second option, CYCLE, defaults to "TRUE" assuming the series has an underlying cycle. If a cycle is not present change this option to "FALSE".
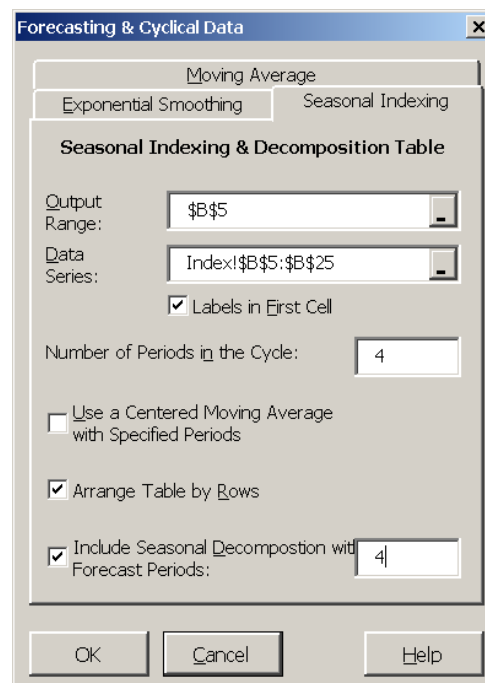
Figure 46. Seasonal Decomposition Forecasting.

The user's requested forecast values are presented in the charts, the trend component forecast is the series of dashes on the linear trend line. The cyclical and seasonal forecasts are the dashes on their respective lines. The composite forecast is the dashes on the actual data line (Sales in the example). The values for these forecast components are indicated in the table after the historical values, the last four values for the example below and in Seasonal Analysis Demo.xls.
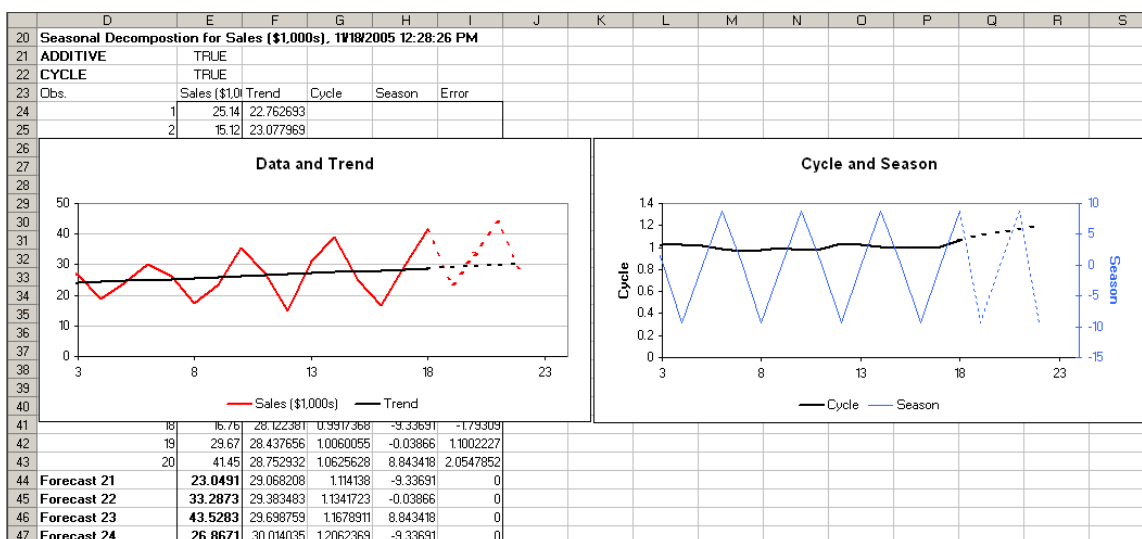
14.3 Moving Average Forecast

A moving average of any series can be calculated by selecting the forecasting icon 🌥 and selecting the *Moving Average* tab (Figure 47). The Moving Average dialog box requires information on the number of periods to include in the moving average and the number of periods to forecast.

Once Simetar has completed the analysis you can change the number of periods for the moving average using the sliding scale to observe how the number of periods affects the goodness of fit measures. The MAPE, WAPE, Thiel U2, RMSE, and MAE are included in the output so you can experiment with different moving average lengths and observe the affects on forecast error. A graph of the historical and predicted values is provided as well. The example of a moving average forecast below comes from the Moving Average Demo.xls workbook.
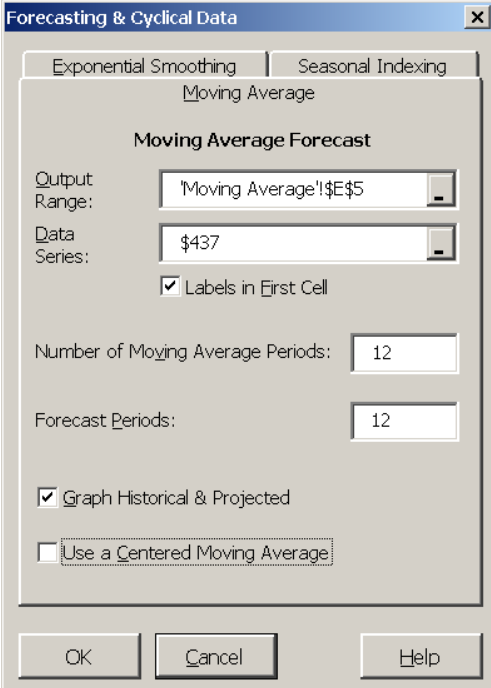
Figure 47. Moving Average Forecast Dialog Box.
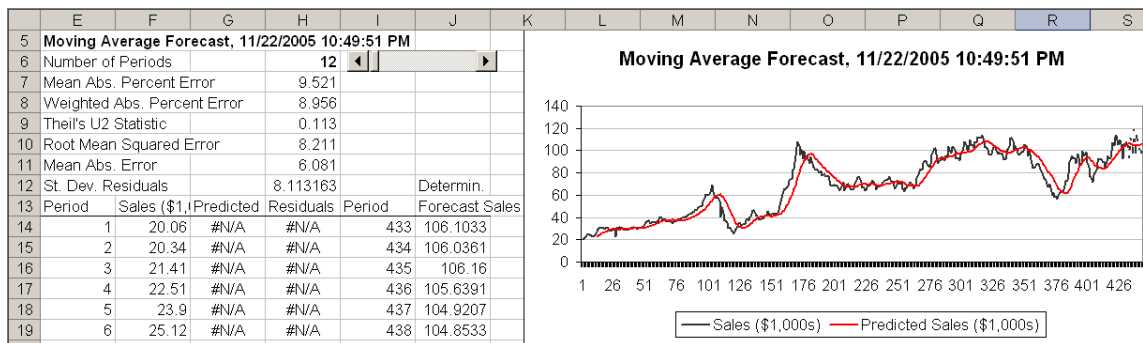
14.5 Exponential Smoothing Forecast

An exponential smoothing forecast for any data series can be developed using the forecasting icon 🌥 and selecting the *Exponential Smoothing* tab (Figure 48). **Before running the Exponential Smoothing option, open Solver to make Excel activate Solver** in the worksheet where you want the forecast model to appear. Solver can be opened and closed by clicking on **Tools > Solver > Close**

Simetar provides three different exponential smoothing estimator/forecasts tools:
- Single exponential smoothing estimates one parameter alpha (*Dampening Factor*).
- Double exponential smoothing or Holt's method estimates parameters for two parameters alpha and beta (*Optional*
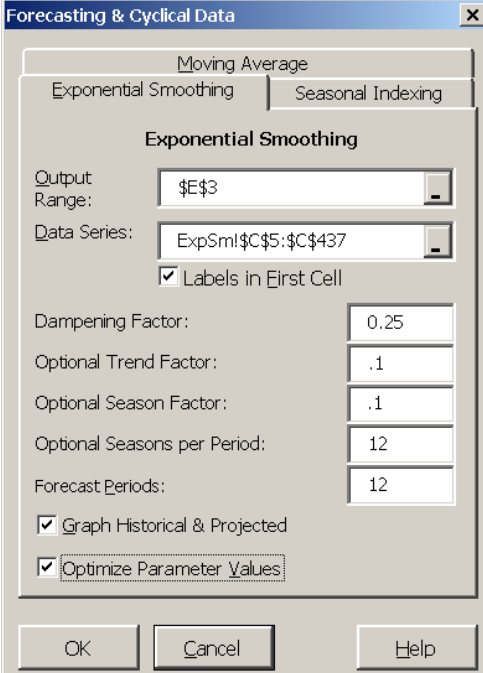
Figure 48. Exponential Smoothing Forecast Dialog Box.

Trend Factor).
- Triple exponential smoothing or Holt-Winter's method estimates three parameters alpha, beta, and gamma (*Optional Seasonal Factor*).

Additionally, Simetar estimates the parameters for the exponential smoothing model with different assumptions about the trend and seasonal component. The options are:
- Holt Method Trend with
  - No trend
  - Dampened additive trend
  - Dampened multiplicative trend
- Holt Winters Seasonal with
  - No seasonal component
  - Additive seasonal component
  - Multiplicative seasonal component

These alternative specifications are effected by changing the Trend Method and the Trend Method options from 0 to 1 or 2 in the output. Re-run Solver after changing any option.

Simetar estimates and forecasts the requested model based on the non-zero initial guesses the user provides in the dialog box or by using SOLVER to optimize the parameters by selecting parameters that minimize the MAPE (Figure 48). Probabilistic forecasts of the exponential smoothing model can be observed by setting the Stochastic Forecast option to "TRUE". The probabilistic forecast values appear at the bottom of the second column of the results. See Exponential Smoothing Demo.xls for the example presented below.

After Simetar estimates the initial model, you can experiment with alternative parameters by using the slide scales for the Level Smoothing Constant, the Trend Smoothing Constant, the Season Smoothing Constant, and the Dampening Parameter, to see what they do to the MAPE, RMSE, MAE, etc.
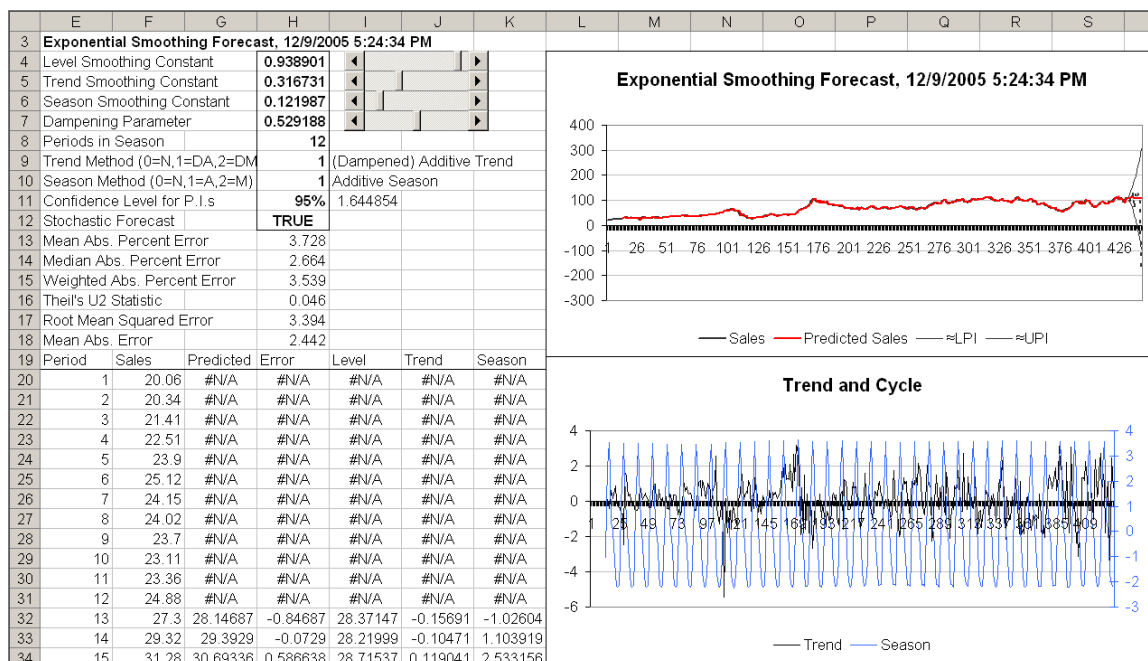
14.4 Measuring Forecast Errors

Five functions are included in Simetar for quantifying forecast errors. The functions are found in most statistics books so the equations are not presented here. An example of the five forecast error statistics is available below and in Forecast Errors Demo.xls.

- Mean Absolute Percent Error function is:
  =MAPE (Array of Residuals, Array of History)

- Weighted Absolute Percent function is:
  =WAPE (Array of Residuals, Array of History)

- Mean Absolute Error function is:
  =MAE (Array of Residuals)

- Root Mean Square Error function is:
  =RMSE (Array of Residuals)

- Theil U2 statistic function is:
  =THEILU2 (Array of Residuals, Array of History, Change)

where: Array of Residuals is the cell reference for the array of errors or residuals,
Array of History is the cell reference for the array of historical data that was used to generate the residuals, and
Change is an optional term to indicate if the statistic is to be calculated in the given levels of the data or as a function of the changes in forecast. FALSE returns the statistic based on levels; TRUE returns the statistic based on changes. The default value is FALSE.

	E	F	G	H	I	J	K
5	Forecast Error Statistics			Values	Formulas		
6	Mean Abs. Percent Error			5.226	=MAPE(D6:D29,B6:B29)		
7	Weighted Abs. Percent Error			5.295	=WAPE(D6:D29,B6:B29)		
8	Theil's U2 Statistic			0.064	=THEILU2(D6:D29,B6:B29)		
9	Root Mean Squared Error			1.714	=RMSE(D6:D29)		
10	Mean Abs. Error			1.400	=MAE(D6:D29)		
11	Mean			26.43083	=AVERAGE(B6:B29)		
12	Std Dev from Mean			3.657701	=STDEVP(B6:B29)		
13	Std Dev of Residuals Trend			1.714291	=STDEVP(D6:D29)		

15.0 Time Series Analysis and Forecasting

Functions for estimating and forecasting time series models in Simetar are presented in this section. Functions used to test for stationarity and number of lags are described first, followed by a general autoregressive model menu for estimating autoregressive (AR) and vector autoregressive (VAR) models. The time series analysis functions facilitate parameter estimation and forecasting with both AR and VAR models to aid in developing probabilistic forecasts for simulation. The time series capabilities of Simetar are demonstrated in Time Series Demo.xls.

15.1 Tests for Stationarity

Time series models should only be estimated for data series that are stationary. A series can generally be made stationary by differencing. An accepted test for determining if a series is stationary is the Dickey-Fuller test. The Dickey-Fuller Test can be calculated using the Simetar function =DF ( ). The =DF( ) function allows the user to test for alternative combinations of differences in an efficient manner to find the combination of adjustments necessary to make a series stationary. The equation used to calculate the DF statistic is:

$$\underbrace{\underbrace{\Delta Y_t \ = \ B_0 \ + \ B_1 Y_{t-1}}_{\text{Dickey-Fuller Test}} \ + \ B_3 \ T_t \ + \ \sum_{i=1}^{n} \ \sigma_i \ \Delta Y_{t-i}}_{\text{Augmented Dickey-Fuller Test}}$$

where: ΔY_t is the first difference of the data series Y,

B_0 is the intercept,

B_1 is the slope parameter estimated for the lagged Y variable (Y_{t-1}),

B_3 is the slope parameters estimated for the trend variable (T), and

σ_i is the parameter for the ΔY_{t-i} for different lengths of higher order lags (i), such as first, second, third, … order lags.

The Dickey-Fuller Test uses the first two components of the above equation and tests for the presence of nonstationarity, in the absence of trend. The Augmented Dickey-Fuller Test includes the third and/or the fourth components of the equation to test for the presence of a trend in the series and for higher order differences. The Simetar function to calculate the Dickey-Fuller Tests on a series of data is:

=DF$\big($Y Values Range, $\big[$Time Trend$\big]$, $\big[$No. of Lag Diffs$\big]$, $\big[$No. of Diff.$\big]\big)$

where: Y Values Range is the location of the data series to be tested (this is all that is necessary for the basic Dickey-Fuller Test),

Time Trend is a true or false switch to indicate whether a trend is to be included in the Augmented Test: "False or 0" for no trend and "True or 1" for a trend,

No. of Lag Diffs is the number of higher order lags to use for the Augmented Test, usually 0, (this is the value for n in the ΔY_{t-i} summation), and

No. of Diff is the number of differences for the original data series Y. This parameter can be used to test for nonstationarity of a specified number of differences, say 2.

Examples of using the =DF( ) function are provided below and in Time Series Demo.xls to demonstrate how it can be used. The basic Dickey-Fuller Test is entered as:

  =DF(Y Values Range)

The Augmented Dickey-Fuller Test that includes a trend is entered as :

  =DF(Y Values Range, 1)

The Augmented Dickey-Fuller Test that has no trend and tests for the presence of a second order autocorrelation lag is entered as:

  =DF(Y Values Range, 0, 2)

The Augmented Dickey-Fuller Test that includes trend and tests for the presence of a second order autocorrelation lag is entered as:

  =DF(Y Values Range, 1, 2)

The null hypothesis for the Dickey-Fuller Tests is: H_0: data series is nonstationary. The critical test statistic for the Dickey-Fuller Test, based on large sample theory, is approximately -2.9 at the 5% level. The null hypothesis is rejected if the DF statistic is less than the -2.9 critical value. The Dickey-Fuller test demonstrated above is in the Tests worksheet of Time Series Demo.xls workbook. The Dickey-Fuller tests for the data are reported for alternative lags, differences, and trend show how the function can help identify the combination of differences, trend, and lags necessary to make the raw data series stationary.

15.2 Number of Lags

For time series analysis it is necessary to determine the optimal number of lags for the AR model after determining the number of differences necessary to make the series stationary. The =ARLAG() function in Simetar suggests the optimal number of lags to use for the AR model. The =ARLAG( ) function returns the number of lags that minimizes the Schwarz criterion given a particular number of differences. The function is programmed as:

  =ARLAG (Y Values Range, [Constant], [No. of Diff])

where: Y Values Range is the range of the time series data to be evaluated,
   Constant is an optional term if the AR model is expected to have a constant term (true or 1) or has no constant (false or 0). The default is to use a constant term (true) if the value is omitted, and
   No. of Diff is the optional number of differences of the original data series Y assumed to make the series stationary.

The =ARLAG( ) function bases its suggestion for the number of lags on the Schwarz criterion test. The test statistic for the Schwarz criterion can be calculated using the following Simetar function

  =ARSCHWARZ (Y Values Range, [Constant], [No. of Diff])

where: All parameters are defined the

	C	D	E	F	G	H	I	J
26	Demonstrate the AR and ARSCHARZ Functions to Test the Number of Lags for							
27	Alternative Assumptions about Constant and Numbers of Differences.							
28		No. of		Calculated				
29	Constant	Differences		No of Lags		Formula		
30				=ARSCHWARZ()				
31	0	1		1.23		=ARSCHWARZ($A$7:$A$91,C31,D31)		
32	0	2		1.45		=ARSCHWARZ($A$7:$A$91,C32,D32)		
33	0	3		1.73		=ARSCHWARZ($A$7:$A$91,C33,D33)		
34	1	1		1.23		=ARSCHWARZ($A$7:$A$91,C34,D34)		
35	1	2		1.45		=ARSCHWARZ($A$7:$A$91,C35,D35)		
36	1	3		1.73		=ARSCHWARZ($A$7:$A$91,C36,D36)		
37				=ARLAG()				
38	0	1		1		=ARLAG($A$7:$A$91,C38,D38)		
39	0	2		2		=ARLAG($A$7:$A$91,C39,D39)		
40	0	3		5		=ARLAG($A$7:$A$91,C40,D40)		
41	1	1		1		=ARLAG($A$7:$A$91,C41,D41)		
42	1	2		2		=ARLAG($A$7:$A$91,C42,D42)		
43	1	3		5		=ARLAG($A$7:$A$91,C43,D43)		

same as the ARLAG function.

A table for implementing the =ARLAG( ) and =ARSCHWARZ( ) functions is demonstrated above. In Excel these functions are dynamic, so you can change the number of differences or the presence of a constant and observe the change in the test statistics. An example of how the =ARLAG( ) and the =ARSCHWARZ( ) functions are used is provided in the Tests worksheet of Time Series Demo.xls workbook. Both tests are demonstrated for 1-4 differences, with and without the constant term. Use the =ARSCHWARZ( ) function to test alternative differences and select the lag structure that minimizes the Schwarz test statistic.

15.3 Sample Autocorrelation Coefficients

In time series modeling it is useful to estimate the sample autocorrelation coefficients and the sample partial autocorrelation coefficients. These coefficients are calculated using the Simetar functions =AUTOCORR( ) and =PAUTOCORR( ). The functions are programmed as:

        =AUTOCORR (Y Values Range, No. of Lags, No. of Diff)
        and
        =PAUTOCORR (Y Values Range, No. of Lags, No. of Diff)

where: Y Values Range is the range of the time series data to be evaluated,
         No. of Lags is the number of higher order lags to test, and
         No. of Diff is the number of differences of the original data series Y to test.

Both of these functions can be used as "scalar" or "array" functions. When used as a scalar, the functions return a single value in the cell which is highlighted. The value returned is the correlation coefficient or the partial autocorrelation coefficient. To use these functions in their array form, highlight three cells in a 3x1 or 1x3 pattern, enter the function name and parameters indicated above, and then press the *Control Shift Enter* keys. Three values will be calculated and placed in the highlighted array. The first value (top or left most) is the autocorrelation or partial autocorrelation coefficient. The next (middle) value is the Student's -t statistic for the coefficient. The last value is the standard error for the coefficient. In the array form these functions can be used to develop tables showing the autocorrelation coefficients and their levels of statistical significance for alternative numbers of lags and differences.

The example on the right demonstrates using the two functions to estimate sample autocorrelation and partial autocorrelation coefficients. The example comes from the Tests worksheet of the Time Series Demo.xls workbook. Four different lags and first and second differences were tested for the data series. Both autocorrelation functions are demonstrated in array form and the partial autocorrelation coefficient function is demonstrated as a scaler to develop a table of test statistics.

	C	D	E	F	G	H	I	J	K	L
48	Demonstrate the Auto-Correlation and Partial Auto-Correlation Coefficients Functions									
49	Sample and Partial Auto Correlation Coefficient Functions Demonstrated as Array Functions									
50	Test Lags of 1, 2, 3, and 4 periods, assuming First or Second Differenced Data									
51				Sample	Auto Cor	Student-t	Std Error for			
52	No Lags	No. Diff		Coefficie	Statistic	Auto Corr Coef		Formula for =Autocorr() in Column F		
53	1	1		-0.2885	-2.6444	0.10911		=AUTOCORR($A$7:$A$91,C53,D53)		
54	2	1		-0.2054	-1.743	0.11784		=AUTOCORR($A$7:$A$91,C54,D54)		
55	3	1		0.17633	1.44494	0.12203		=AUTOCORR($A$7:$A$91,C55,D55)		
56	4	1		-0.1216	-0.9723	0.12503		=AUTOCORR($A$7:$A$91,C56,D56)		
57	1	2		-0.5313	-4.8405	0.10976		=AUTOCORR($A$7:$A$91,C57,D57)		
58	2	2		-0.1176	-0.8566	0.1373		=AUTOCORR($A$7:$A$91,C58,D58)		
59	3	2		0.26795	1.93455	0.13851		=AUTOCORR($A$7:$A$91,C59,D59)		
60	4	2		-0.1289	-0.8915	0.14462		=AUTOCORR($A$7:$A$91,C60,D60)		
61				Partial						
62				Auto Cor	Student-t	Std Error for Partial				
63	No Lags	No. Diff		Coefficie	Statistic	Auto Corr Coef		Formula for =PAutocorr() in Column F		
64	1	1		-0.2885	-2.6444	0.10911		=PAUTOCORR($A$7:$A$91,C64,D64)		
65	2	1		-0.3149	-2.8858	0.10911		=PAUTOCORR($A$7:$A$91,C65,D65)		
66	3	1		0.00915	0.08384	0.10911		=PAUTOCORR($A$7:$A$91,C66,D66)		
67	4	1		-0.1411	-1.293	0.10911		=PAUTOCORR($A$7:$A$91,C67,D67)		
68	1	2		-0.5313	-4.8405	0.10976		=PAUTOCORR($A$7:$A$91,C68,D68)		
69	2	2		-0.5572	-5.0763	0.10976		=PAUTOCORR($A$7:$A$91,C69,D69)		
70	3	2		-0.2534	-2.3085	0.10976		=PAUTOCORR($A$7:$A$91,C70,D70)		
71	4	2		-0.2035	-1.8535	0.10976		=PAUTOCORR($A$7:$A$91,C71,D71)		

15.4 Maximum Likelihood Ratio Test

A maximum likelihood ratio test (LRT) is included as a function in Simetar to facilitate estimation of the number of lags for an unrestricted vector autoregressive (VAR) model. The LRT is estimated for alternative possible lags using the following function:

=LRT (Y Values Range, No. of Lags, Constant, No. of Diff, Error Correction)

where: Y Values Range is the range of the time series data to be evaluated for potential inclusion in a VAR. Two or more data series must be identified.
No. of Lags is the number of lags to test,
Constant is a switch as to whether a constant term (True or 1) is to be included or not (False or 0),
No. of Diff is the number of differences of the original data series to test, and
Error Correction is whether to perform an error correction (True or 1) on the data or not (False or 0).
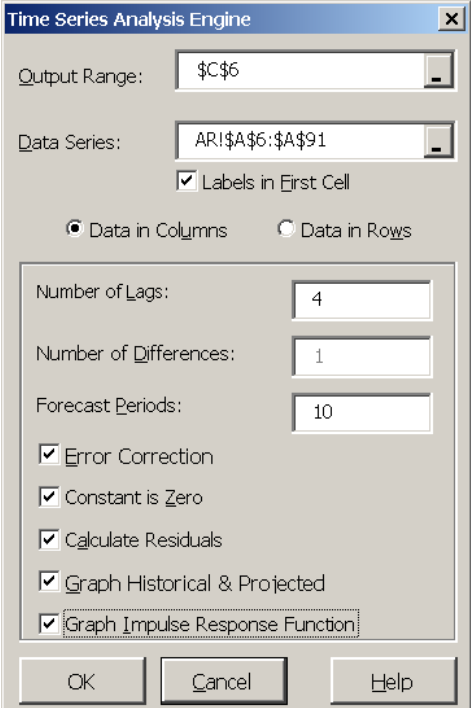
The =LRT( ) is demonstrated in the Tests worksheet of the Time Series Demo.xls workbook. Two data series were tested for 7 different lags assuming three differences, a constant, and error correction. The parameters for the =LRT( ) are displayed in a table below the LRTs so one can easily change a parameter and observe the changes in the LRTs.

	C	D	E	F	G	H	I	J	K
82	Demonstrate the LRT or Liklihood Ratio Test								
83					Lags				
84		1	2	3	4	5	6	7	
85	Wichita	45.25	19.23	4.54	21.71	7.12	7.68	6.65	
86	Salina	-36.21	0.09	7.68	0.98	1.37	1.37	10.13	
87		Constant	Diffs	Error Correction					
88		TRUE	3	TRUE	Change the bold values and observe the LRTs.				
89		Formula in cell D85							
90		=LRT($A$7:$A$91,D84,$D$88,$E$88,$F$88)							

15.5 Estimating and Forecasting Autoregressive (AR) Models

The Time Series Analysis menu (Figure 49) provides the mechanism to program the information necessary to estimate and forecast an auto-regressive (AR) model. The Time Series Analysis menu is activated by selecting the ⬛ icon. If you specify the data to analyze as a single variable (column of data) in the *Data Series* window, the Time Series Analyzer will estimate an AR model. (Specifying two or more columns causes Simetar to estimate a VAR model.) The *Number of Lags* and *Number of Differences* for the original data must be specified for the AR model. In addition, provisions are available in the dialog box to indicate whether or not the *Constant is Zero*. The number of *Forecast Periods* to project

Time Series Analysis Engine

Output Range: $C$6

Data Series: ARI$A$6:$A$91

☑ Labels in First Cell

◉ Data in Columns   ○ Data in Rows

Number of Lags: 4

Number of Differences: 1

Forecast Periods: 10

☑ Error Correction
☑ Constant is Zero
☑ Calculate Residuals
☑ Graph Historical & Projected
☑ Graph Impulse Response Function

OK   Cancel   Help

Figure 49. Time Series Analysis Dialog Box.

using the estimated model is also specified in the dialog box. It is recommended that the Time Series menu be programmed to: (a) calculate the residuals, (b) graph the historical and projected values, and (c) graph the impulse response function (see example below).

The results of estimating an AR model with four lags and one difference or an AR (4,1) model is presented below and in the AR worksheet of the Time Series Demo.xls workbook. Several supporting tests are provided along with the coefficients, namely, the Schwarz test, and two Dickey-Fuller tests. The forecast values for the AR model are provided for 10 periods, as programmed in the dialog box, and are labeled "Forecast." "Impulse Response" values are provided for each forecast period (see th example below). Student-t statistics for the sample and partial autocorrelation coefficients are provided for the 10 periods of forecast output.

	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
6	AR Series Analysis Results for 4 Lags & 1 Difference, 12/18/2004 10:09:44 PM
7		Constant	WichitaL1	WichitaL2	WichitaL3	WichitaL4					Historical & Predicted
8	Wichita	0.407	-0.383	-0.327	0.000	0.000
9	S.E. of Coefficients
10	Wichita	0.333	0.108	0.108	0.000	0.000
11	Restriction Matrix
12	Wichita	1	1	1	0	0
13	Difference	1
14	Character	Dickey-Fu	Aug. Dick	Schwarz	S.D. Resic	MAPE	AIC
15	Wichita	-12.146	-9.921	1.227	2.900186	11.36	2.20
16	Forecast	Impulse	Auto-	t-Statistic	Partial	t-Statistic
17		Response	Correlatio	(AutoCorr	AutoCorr	(Part.Auto	Period
18	32.051	1.000	-0.28853	-2.64439	-0.28853	-2.64439	1
19	32.793	-0.369	-0.2054	-1.74304	-0.31486	-2.88577	2
20	33.202	-0.179	0.176326	1.444937	0.009148	0.083843	3
21	33.209	0.183	-0.12157	-0.97233	-0.14108	-1.29303	4
22	33.479	-0.011	-0.08964	-0.70901	-0.15115	-1.3853	5
23	33.780	-0.054	0.271286	2.133082	0.157704	1.445381	6
24	33.983	0.023	-0.18847	-1.4076	-0.09713	-0.89017	7
25	34.213	0.008	-0.01516	-0.11063	0.00145	0.013291	8
26	34.465	-0.010	0.039689	0.289632	-0.0951	-0.87162	9
27	34.700	0.001	-0.16265	-1.18571	-0.17961	-1.64619	10

The time series output generated by Simetar is dynamic meaning that the beta coefficients in the AR model will update if you change the values in the original data or replace the input data array with another series of data. An added feature is the capability to impose restrictions on the initial AR model by dropping out/re-entering lags in real time. The Restriction Matrix has 1's beneath each lags' coefficient. When the restriction value of 1 is changed to 0 the model is re-estimated without that particular variable or lag. The example AR model in DemoSimetar-Ar was run with 4 lags so the user can experiment with deleting unnecessary lags using the Restriction Matrix. When the 2nd, 3rd and 4th lags are restricted out the standard deviation for the residuals increases slightly from 2.86 to 3.06. As these higher order lags are removed the MAPE increases only about 1.3 percentage points. The AIC is minimized when lags 3 and 4 are removed.

Note that the initial number of lags and differences specified for the AR model determines the number of observations used to estimate the coefficients. When an AR model of 1st differenced data is estimated with four lags initially but the 3rd and 4th lags are restricted out, the resulting coefficients will not equal those for an AR(1) model estimated with two lags. The reason the coefficients are slightly different is that the latter model uses two more observations to estimate the parameters. It is recommended that the restricted AR model be re-estimated using the exact number of lags once the restricted model is acceptable.

As the restrictions on the lags are imposed on the unrestricted model the following test statistics do not change: Dickey-Fuller Test, Augmented Dickey-Fuller Test, and Schwarz Test (see example above). These statistics do not change because they reflect the number of differences

specified for the unrestricted model. For example, the Dickey-Fuller Test statistic for an AR(4,1) model is calculated as =DF(data,,,1) and for an AR(4,2) model it is =DF(data,,,2). The Schwarz Test statistic is based on the number of differences [=ARSCHWARZ(data,,No. of Differences)] and does not change as the number of lags is restricted.

It is possible to interactively analyze the impact of changing the number of differences to the data in the AR model. In the second row of the Restriction Matrix (see the example above) is the word Differences followed by a value, in this case 1. The 1 in the Difference row means the data have been differenced once. To "re-run" the model with second differenced data, type a 2 into the restriction matrix in place of the 1. This change causes Simetar to re-estimate all of the parameters and update the goodness of fit test statistics.

The predicted values over the historical period and their residuals are provided for the AR model. The residuals are also expressed as a fraction of the predicted data. The predicted values and the residuals begin with observation 6, for this example, because the lag/difference structure of an AR(4,1) model uses the first 5 observations.

A graph of the historical and predicted values for the data series is generated by the Time Series function. The thin line represents the original data while the bold line represents the predicted values. Projections beyond the historical data in the graph correspond to the 10 period forecast requested in the dialog box (Figure 43).

A graph of the Impulse Response Function is also included in the forecast. The impulse response values are included in the output, but they are easier to see in the graph. A stationary model will exhibit continuously decreasing impulse responses to a 1 unit change at the outset of the period, as depicted by the graph in the AR Worksheet. The Impulse Response Function graph changes as the lags in the model are restricted out. Not shown in example above are the autocorrelation and partial autocorrelation function graphs for the AR model.

15.6 Estimating and Forecasting Vector Autoregressive (VAR) Models

The Time Series Analysis Engine dialog box (Figure 50) can be used to estimate and forecast VAR models. VAR model analyses begin by selecting the ▨ icon on the Simetar toolbar. To estimate a VAR model, take all the steps used to estimate an AR model with one exception, specify two or more adjacent series in the *Data Series* menu (Figure 50). When two or more data series are specified, Simetar uses the more general estimation procedure for a VAR. The number of lags and differences should be specified based on prior analyses and tests.

The results of estimating and forecasting a two variable unrestricted VAR model are presented in the VAR worksheet of the Time Series Demo.xls Workbook and below. The Time

Figure 50. Time Series Analysis Dialog Box for a VAR

Series function estimated the parameters for the VAR model using 4 lags and 1 difference with a constant, so 18 parameters are presented in the results. Various time series tests statistics for the model are presented below the parameters.

The first and second rows of the Restriction Matrix contain 1's indicating all lags are initially in the model. These restriction values can be changed to 0's to re-fit the VAR in real time by selectively deleting lags for one or both of the variables (see the example below). Changing the 1's to 0's and observing the change in the test statistics will enable the user to instantly experiment with a large number of model specifications. The interaction among the variables and their lags can be tested interactively using this feature in the Simetar VAR. The third row in the Restriction Matrix provides the switch to re-fit the VAR model with alternative numbers of differences, in real time.

Forecasted values for both of the data series are provided in the output section. Impulse responses for the system of variables are also provided. These impulse response values are also summarized in a graph when requested.

	E	F	G	H	I	J	K	L	M	N	O
8	VAR Series Analysis Results for 4 Lags & 1 Difference with Error Correction. 12/18/2004 10:28:14 PM										
9		Constant	WichitaL1	WichitaL2	WichitaL3	WichitaL4	SalinaL1	SalinaL2	SalinaL3	SalinaL4	Error Corr
10	Wichita	0.431	-0.392	-0.392	-0.092	-0.175	0.094	0.004	-0.047	-0.015	0.000
11	Salina	0.272	0.044	0.121	0.097	0.092	-0.779	-0.334	-0.224	-0.009	0.000
12	S.E. of Coefficients										
13	Wichita	0.345	0.118	0.133	0.140	0.125	0.123	0.159	0.157	0.136	0.000
14	Salina	0.333	0.114	0.128	0.136	0.121	0.119	0.153	0.151	0.131	0.000
15	Restriction Matrix										
16	Wichita	1	1	1	1	1	1	1	1	1	0
17	Salina	1	1	1	1	1	1	1	1	1	0
18	Difference	1									
19	Character	Dickey-Fu	Aug. Dick	LRT		LRT Criti	S.D. Resi	MAPE	AIC	SIC	
20	Wichita	-12.146	-9.921	18.196	2.733	2.84	10.94	2.3148616	2.5828396		
21	Salina	-17.935	-9.176			2.75	9.68	2.2488177	2.5167957		
22	Forecast		Impulse Response								
23	Wichita	Salina	Wichita	Salina	Period						
24	31.121	30.950	1.000	1.000	1						
25	30.417	28.194	-0.373	-0.714	2						
26	30.980	30.800	-0.225	0.282	3						
27	31.790	28.960	0.158	-0.275	4						
28	31.974	30.248	-0.116	0.257	5						
29	32.120	29.668	0.061	-0.252	6						
30	32.247	30.510	0.039	0.205	7						
31	32.456	30.163	-0.053	-0.099	8						
32	32.688	30.598	0.023	0.056	9						
33	32.918	30.524	-0.003	-0.057	10						

Actual and predicted values over the historical period are presented in the top chart. Numbers behind the predicted values over the historical period are provided, beginning with period 6. The forecast values have a label with the word "Pred" following the variables name. Residuals for the VAR predicted values are also included in the output.

The residuals from the historical data can be used to simulate the unexplained variability or stochastic components of the random variables. Use the residuals to estimate the standard deviation about the forecasted values. Also use the residuals to estimate the correlation matrix for correlating random values about the forecasts.

16.0 Other Statistical and Data Analysis Functions

16.1 Summary Statistics

The dialog box used to calculate summary statistics for a variable (Figure 45) appears when the Summary Statistics menu item or icon ![icon] is selected. Select in the *Select Range* box and highlight the range (column or row) to analyze. Next click in the *Output Range* box and click the cell where the results are to be placed. All of the statistics and their names (mean, standard deviation, coefficient of variation, minimum, maximum, lower and upper confidence interval, and sum) will be placed in the worksheet starting with the *Output Range* cell if the *Add Output Labels* button is clicked. The standard deviation can be calculated using either the population or the sample formula. The coefficient of variation, sum, count and autocorrelation coefficient are not calculated unless these statistics are specified by selecting their boxes. Experiment with the dynamic nature of Simetar by changing the values in the original data and observing the updated summary statistics. See Data Analysis Demo.xls for an example.
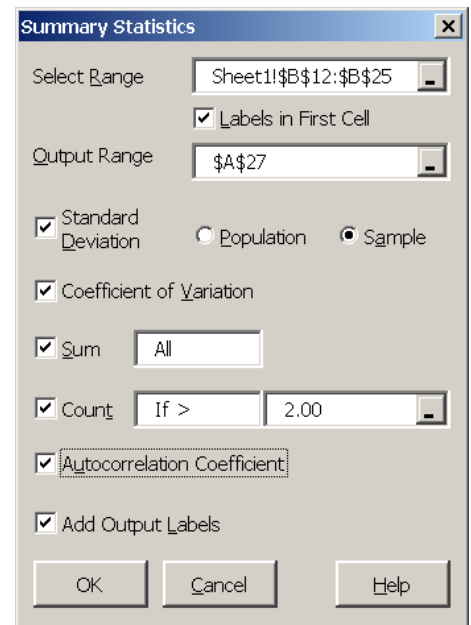
Figure 51. Summary Statistics Dialog Box.

The *Count* and *Sum* options in the Summary Statistics menu are available for conditional counts and sums of the data. Consider the situation where you have 2,500 observations and need to know how many values are less than or equal to 10.0. Perform this calculation by clicking on *Count*, followed by selecting the IF $< =$ box, and then type the target value in the right hand box 10.0. The conditional count will appear with the other statistics.

16.2 Jackknife Estimator

Simetar provides a jackknife function which can be used to estimate parameters for any statistical formula or function in Excel or in Simetar. Given an n-dimensional vector or matrix of data and an associated statistic based on the data, the jackknife procedure sequentially re-estimates the statistic, leaving out the ith row at each iteration, where $i = 1,...,N$. These n statistics are then used to calculate the average statistic, the bias relative to the original statistic, and the jackknife variance of the statistic. The format for the =JACKKNIFE() function is:

    =JACKKNIFE(DataRange, FormulaRef, RetVariance, Delete_D)
    =JACKKNIFE(A2:B20,C2:D3)

Where: Data Range is a reference to a range of data that will be resampled to calculate the jackknife estimator. If this range is an Nx1 vector, then the estimator will be calculated based on sequentially removing the ith row of the vector, where $i = 1,…,N$. Similarly, if this range is an NxK matrix, the estimator will be calculated based on sequentially removing the ith row of the matrix. Thus, multivariate data should be arranged with variables in columns,

FormulaRef is a reference to a range or cell that contains a formula which calculates an estimate based on the given Data Range. The jackknife estimator will be an average of the result of this formula based on the sequentially re-sampled data,

RetVariance is an optional term to include if only the jackknife estimate of the estimator variance is desired. A value of TRUE (or 1) will produce only the variance. A value of FALSE (or 0) will produce the jackknife estimator, bias, and variance. The default value is FALSE, and

Delete_D is an optional term to include if D rows are to be deleted at a time instead of one, where D is a positive integer less than n, the number of rows. The JACKKNIFE Function will then estimate statistics based on removing D adjacent rows at a time sequentially. This method is recommended when dealing with nonlinear statistics and should be used in conjunction with random sub-sampling methods. The default is one.

17.0 Function Evaluation

Two Simetar functions are available for evaluating user specified nonlinear functions. The first, =OPT( ), finds the minimum or maximum of a function given boundary constraints on the control variables. The =OPT( ) function can also be used to find the value of X when a function equals a target value, as zero. The second function, =RINTEGRAL( ) integrates a function over a given range. Both functions provide approximate answers using efficient optimal control search and solve algorithms. The level of precision can be increased, but at a slight cost of longer execution times.

17.1 Optimize a Function

The =OPT( ) function uses the Golden Section method for optimizing a non-linear function specified by the user. The function to optimize (maximize or minimize) can be either typed into the =OPT( ) function as a literal or as an equation typed into a cell. Optimization Function Demo.xls demonstrates both techniques for optimizing functions.

Figure 52. Equation Editor for the Optimization Function =OPT( ).

The easiest method for using the =OPT( ) function is to type =OPT and then click on Excel's Equation editor, f_x, and fill in the blanks in the OPT equation editor form (Figure 52). The optimization function parameters are:

   =OPT (Formula, Constraint Type, Change Variable, Lower Guess, Upper Guess, Max
       Iterations, Precision)

where: Formula is the function to be optimized, as: $= 100 - 25X + 45X^2$ and must be typed into the referenced cell as a formula,

Constraint Type must be typed as the word "Min" or "Max" for minimization or maximization, respectively,

Change Variable is the cell which refers to the X variable in the function and can be any feasible value of X,

Lower Guess is the minimum X,

Upper Guess is the maximum X,
Max Iterations is the maximum number of calculation cycles to use, and
Precision is the degree of accuracy, such as 0.000001.

The value of X which causes the Y function to be optimized will appear in the cell where =OPT( ) is typed. Changing the parameters will cause Excel to calculate a new optimal value if the current solution is at a boundary or more precision can be obtained. Changing the function or input values to the function of course changes the =OPT( ) answer.

17.2 Value of a Function

Given a complex polynomial function that can be programmed in a cell as $Y = f(X)$, Simetar can solve for the value of X where Y equals a target value such as zero. A variation on the =OPT( ) function can be used to solve this type of optimization problem. The parameters for the function are:

> =OPT (Formula, Target Value, Change Variable, Initial Guess, Upper Bound, Max Iterations, Precision)

where: Formula is the cell reference for the function to be optimized,
Target Value is the value of Y when the function is optimized,
Change Variable is the cell referring to the X variable in the function and can be any feasible value of X,
Initial Guess is the lower bound constraint of X,
Upper Guess is the upper bound constraint of X,
Max Iterations is the maximum number of calculation cycles, and
Precision is the degree of accuracy, such as 0.000001.

When the =OPT( ) function fails to find the target value for Y over the range of the function it returns #VALUE in the cell where =OPT( ) is typed. In this case, try another initial guess, the upper bound, the level of precision or the maximum number of iterations. Excel will solve some functions very fast; for example, $Y = X^4$ will find that Y equals 23 at 2.1958 very rapidly. See Optimization Function Demo.xls for this example.

17.3 Integral of a Function

A function can be integrated over a specified range using the =RINTEGRAL( ) function. This function provides an approximate value for the integral using Riemann Integration. The level of precision can be increased by increasing the number of partitions. The easiest way to use the function is to develop a table of parameters and then use Excel's Equation editor after typing =RINTEGRAL, as depicted in Figure 53. An example of integrating a function $Y = 100 - 25X + 45X^2$ over the interval of X equal 0 to 100 is provided in Optimization Function Demo.xls. The parameters for the integration function are:

Figure 53. Equation Editor for the Integral Function.

> =RINTEGRAL (Formula, Variable Ref, Lower Bound, Upper

Bound, Partitions)

where: Formula is the cell reference to the equation to be integrated,
          Variable Ref is the cell reference for the independent variable (X) in the equation,
          Lower Bound is the minimum X for the range of the integration,
          Upper Bound is the maximum X for the range of the integration, and
          Partitions is the number of intervals X range is partitioned into for integration.

The answer will appear in the =RINTEGRAL( ) function cell. It is recommended that you increase the number of partitions until the change in the integral answer is zero. As you increase the number of partitions, response time will slow. For the example in Optimization Function Demo.xls the true value of 14,885,000 is reached at 300,000 partitions in about 25 seconds.

18. 0 Getting Help with Simetar

Simetar Help is provided in two forms: detailed description of the functions and equation editing help. Detailed descriptions are available for all of the Simetar functions by clicking the help icon ❔ on the toolbar. When the help icon is selected the Help Index for Simetar window (Figure 54) appears on the screen. Scroll down to the function of interest and click on the function name. This action results in the requested Simetar Help screen appearing on the screen.

Simetar Help	Windows Help
Help Index for Simetar	NORM
Click on a topic link below to search for specific functions:	Normal random variable
All Functions	Syntax
Matrix Operations	NORM(Mean,StanDev,RandNumber)
Random Variables	Mean — Population expected value of the normal random variate
Regression	StanDev — Population standard deviation of the normal random variate
Time Series	RandNumber — Uniform standard deviate used to generate the random variable
Univariate Parameter Estimation	
Click on a topic below for help with toolbar menu options:	The pdf for the normal distribution given μ and σ is:
Correlation Matrix	
Default User Settings for Simetar	
Empirical Distribution	$$f(x\,|\,\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\,e^{\frac{-(x-\mu)^2}{2\sigma^2}}, x \in (-\infty,\infty), \mu \in (-\infty,\infty), \sigma > 0$$
Forecasting & Cyclical Data	
Graphing Center	
Hypothesis Testing for Data	

Figure 54. Help Index Dialog Box.   **Figure 55. Example of a Simetar Help Screen.**
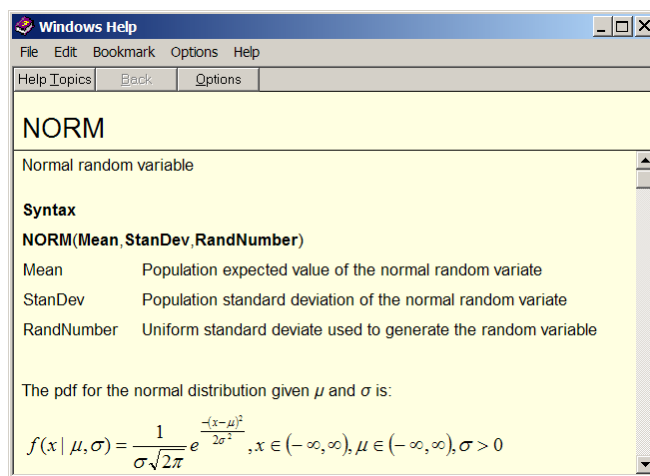
An example of requesting help from the Simetar Help Index for the =NORM( ) function is displayed in Figure 55. The help provided in the screen is designed to supplement the material in this manual. You can either print the help screen, return to the Help Index, or close the help screen by clicking on the appropriate button at the bottom of the Help Screen. Additional help on the function is available from Simetar by clicking on the line "Help on the function" as demonstrated for the =NORM() function below.

Excel provides pop-up help menus to assist with writing or editing equations. To access help for equation programming simply type "equal and the function name" in a cell and then click the "=" button or the f_x icon on the formula bar. An example of how this works for getting help

with the =CSND( ) function is provided in the worksheet example below and Figure 56. In the example the analyst has highlighted three cells (B7:B9) in preparation for using the CSND function as an array. After typing "=CSND" click the **"fx"** button in the formula toolbar at the top of the worksheet and Excel will place a dialog box like Figure 50 on the screen.

The equation help box in Figure 50 indicates the order of parameters for the =CSND( ) function and the names of the parameters. You can fill in the worksheet cell locations for the parameters by clicking the miniature grid to the right of each parameter and painting the appropriate cells with the mouse. After filling in values for the parameters select OK.

The equation editor help function can be used to develop new equations and to de-bug existing equations. Select a cell with an existing equation and click the **"="** or "*fx*" button on the formula bar to see the equation editing help box. Equation editing help screens are available for all Simetar and Excel functions.

	IF	▾ X ✓ fx	=csnd(
	A	B	C	D
1	Correlation Matrix			ISNDs
2	1	0.25	-0.5	0.550576
3		1	0.3	0.708632
4			1	0.894564
5	Generate 3x1 array of CSNDs			
6		=csnd(		
7				
8				

Figure 56. Example of the Equation Help Box.

19.0 Solutions to Problems in Simetar Application

Like all computer programs Simetar 2006 is the result of many enhancements. Each time one function is complete we find two more to add and in the process a better way to do the first function is developed. The program has come a long way given that it began in May 2000. Simetar continues to grow and become more useful.

Most problems are associated with installing Simetar on computers with old operating systems/versions of Excel and operators without administrative privilege. The optimal environment is Windows 2000 operating system with Microsoft Office XP. The Demo programs were developed in this environment. The first time you open one of the Demo workbooks it may warn you of embedded macros – select "Enable Macros" and proceed. Next your Excel may warn you that the Demo has external links – select "No" and proceed. Save the workbook to your hard drive and the next time it is opened you will not have link warnings. The workbook link warnings are caused by your computer storing Simetar in a different location than the developer's computer. Excel will update the links on its own.

This section documents errors we have observed. Most of the problems occur because Excel's Calculation is set to Manual or the Operating System burps and sets Calculation to Manual during your Excel session. **Set Calculation to Automatic and leave it there and check it if errors occur.**

19.1 My program was working when I saved it, but now the Simetar functions have #NAME

Sometimes Simetar and Excel gets confused and you need to remind Excel that Simetar is loaded, to do this follow these steps:
    Tools > Add-Ins >   Uncheck the box for Simetar
    Then repeat the process
    Tools > Add-Ins >   Check the box for Simetar

19.2 File Not Found dialog box with a file name of "PBJ.XLA" listed, appears when I open a workbook

Click the "Cancel" button and Excel will update the links to Simetar and PBJ using the current location of these files on your computer. This error occurs when the workbook was created on a different computer. Save the workbook and the next time it is opened there will be no problem.

19.3 Simetar Functions returns #NAME! instead of values

Sometimes Simetar and Excel gets confused and you need to remind Excel that Simetar is loaded, to do this follow these steps:
    Tools > Add-Ins >   Uncheck the box for Simetar
    Then repeat the process
    Tools > Add-Ins >   Check the box for Simetar

    If your computer is running Excel 97, load Service Pack 2. If your computer is running Excel 95, get a newer version of Excel.

19.4 Scenario names in Stochastic Dominance tables appear as #NUM!

Press Function key F9
Set calculation to automatic by following these steps:
        Tools > Options > Calculation, set the calculation option to Automatic

19.5 Statistics for the first stochastic variable in SimData Worksheet appear as #DIV/0!

Press Function key F9 if the problem goes away, do the following:
    Check Tools > Options > Calculation, set the Calculation option to Automatic

    Check if the variable is a constant. If it is then the means will not be zero but the standard deviation and coefficient of variation will be #DIV/0!

19.6 Values for SERF table and chart in SERFTbl1 do not change when you change the ARACs or the utility function

Check Tools > Options > Calculation, set the Calculation option to Automatic

19.7 Results from Testing a Single Variable for Normality returns #VALUE! in place of values

Check Tools > Options > Calculation, set the Calculation option to Automatic
Delete the formats in the cells for the output range that may be left over from previous sessions.

19.8 Results of Compare Two Data Series returns #DIV/0! and #NUM! in place of values

Check Tools > Options > Calculation set the Calculation option to Automatic

19.9 Multiple regression returns #DIV/0! for standard deviation of residuals and/or MAPE is #VALUE!

Check Tools > Options > Calculation, set the Calculation option to Automatic

19.10 Multiple regression does not update the beta hats and goodness of fit statistics when a restriction value is changed

Check Tools > Options > Calculation, set the Calculation option to Automatic

19.11 Multiple regression does not update the beta hats and goodness of fit statistics when one of the X or Y observations is changed

Check Tools > Options > Calculation, set the Calculation option to Automatic

19.12 Multiple regression, time series, and other menu enabled functions return numbers instead of the names for the X and/or Y variables

The dialog boxes allow you to enter "Labels in First Cell," you did not include the label in the first cell, so Simetar used the first observation as the name of each X variable and/or for Y. Include the variables label when dialog boxes are used to enter data for functions.

19.13 Time series (AR and/or VAR) procedure returns #VALUE! instead of the coefficients

Check Tools > Options > Calculation, set the Calculation option to Automatic

19.14 Stochastic variables (cells) in the worksheet do not change when the Enter or F9 Keys are pressed

Check Tools > Options > Calculation, set the Calculation option to Automatic

Check the Simetar Toolbar to see if worksheet sampling has been set to "Expected Value", if so click the Expected Value button on the Simetar Toolbar.

19.15 Stochastic variables (cells) in the worksheet are fixed at zero or the mean and do not change when F9 is pressed

The "Expected Value" button in the Expected Value button on the Toolbar is turned on. Turn the option off by clicking on the Expected Value button.

19.16 The CDF or SERF chart has numbers instead of names on the lines and/or the scenario names in the legend are numbers

The "Labels in First Cell" option was turned on so the program used the first observation for each scenario as the scenario names.
Be sure that the label in the first row starts with a letter, not a number, as 1998 or 2000.

19.17 Results and calculations in the simulation output worksheet, SimData, are gone

Simetar writes the iteration results to worksheet SimData after each run. It uses as many columns of the worksheet required for the output variables in the Simulation Engine. If you had tables from a previous simulation run in the columns needed for the current run, they got over written. When you place summary tables, tests, or chart data in SimData, rename the worksheet so it will be protected from the next simulation.

19.18 Simulation used to run fast and now it has slowed down

- Another Excel workbook which contains stochastic variables may be open. When Simetar simulates the stochastic variables in the open workbook, Excel also simulates the workbooks that are minimized.
- The SERF option is dynamic and can slow the simulation down if the model is simulating more than 500 iterations and SERF is tied to the SimData worksheet.
- Simulation can be slowed down if the SimData output is being used to calculate a large number of CDF and PDF charts.
- The number of Key Output Variables that Simetar is collecting for statistical analysis may have been expanded from previous runs.
- The number of Scenarios is greater than previous runs.
- The SimSolver option in the Simulation Engine is turned on.

19.19 A Simetar matrix or array function returns a single value when you expected an array or matrix of answers.

Press F2 to edit the function; if it is typed correctly press three keys: *Control Shift Enter*.
Any time an array function is used, you MUST end by pressing these three keys: *Control Shift Enter*.

19.20 Hypothesis Test statistics appear wrong.

- Re-do the test and be careful to indicate no labels in the first row and only include the data.
- Change the variable labels or names so they begin with a letter, as Y1988, not 1988 and re-do the test.
- The t-tests are two tailed tests, so thye will not be the values you expect for a one-tailed test.

19.21 After installation, if the Excel Tool Bar does not show "Simetar," it can be re-loaded to the toolbar using the following steps:

Tools > Select Add-Ins ... > scroll down and click the box for Simetar > OK

20.0 List of All Simetar Functions

Following is a list and short description of all functions in Simetar:

Function Name	**Description**
· ANOVA	One way analysis of variance
· ARLAG	Recommends the number of lags in an autoregressive model
· ARSCHWARZ	Schwarz criterion associated with recommended number of lags
· AUTOCORR	Autocorrelation function for a univariate time series
· BANDWIDTH	Bandwidth function in kernel density estimation
· BERNOULLI	Bernoulli random variable
· BERNOULLIDIST	Bernoulli distribution function
· BINOMINV	Binomial random variable
· BLOCKIT	Column Vector to a Matrix
· BOOTSTRAPPER	Bootstrap resampling of a univariate or multivariate series
· BOXCOX	Box-Cox transformation of a data series for normalization
· BOXCOXEXP	Estimate of the Box-Cox exponent in a Box-Cox transformation
· BOXM	Box's M statistic for testing multivariate variances
· CAUCHY	Cauchy random variable
· CAUCHYDIST	Cauchy distribution function
· CDFDEV	Indicate goodness of fit between sample data & known distribution data
· CELLSUB	Replace an item or items in a block of data
· CERTEQ	Certainty equivalent of a data series assuming a utility function
· CMOVAVG	Centered moving average
· CONCAT	Concatenate two or more matrices
· COSDIST	Cosine distribution function
· COSINV	Cosine random variable
· CSND	Correlated standard normal deviates
· CUSD	Correlated uniform standard deviates
· DELNUM	Remove the numbers from a string of text and numbers
· DELTEXT	Remove the text from a string of text and numbers
· DEMPIRICAL	Discrete empirical distribution random variable
· DEXPONDIST	Double exponential distribution function
· DEXPONINV	Double exponential random variable
· DF	Dickey-Fuller test statistic
· DIRICHINV	Dirichlet random variable
· EDF	Empirical distribution function
· EMP	Empirical random variable
· EMPCOPULA	Empirical copula function
· EMPIRICAL	Empirical random variable
· EPANDIST	Epanechnikov distribution function
· EWMA	Exponentially weighted moving average
· EXPONINV	Exponential random variable
· EXTVALDIST	Extreme value distribution function
· EXTVALINV	Extreme value random variable
· GEOMDIST	Geometric distribution function
· GEOMINV	Geometric random variable
· GMDIF	Gini's mean difference
· GRK	GRK random variable
· GRKS	GRKS random variable
· GRKSDIST	GRKS distribution function
· GUMBELDIST	Gumbel distribution function
· GUMBELINV	Gumbel random variable
· HOTELLTDIST	Hotelling T-squared distribution function
· HOTELLTINV	Hotelling T-squared random variable
· HYPERGEOMINV	Hypergeometric random variable
· IMPULSE	Impulse response function in a vector autoregression
· INVGAUS	Inverse Gaussian random variable
· INVGAUSDIST	Inverse Gaussian distribution function
· IQR	Inner quartile range of a sample
· ITERATION	Show the iteration number during simulation

·	ITERSUM	Sum a value across iterations during a simulation
·	JACKKNIFE	Jackknife estimate of statistic, bias, and variance
·	KDEINV	Random variable based on a kernel density estimate
·	KTAU	Kendall's Tau measure of concordance
·	LOGISTICDIST	Logistic distribution function
·	LOGISTICINV	Logistic random variable
·	LOGIT	Logit binary response regression
·	LOGLOGDIST	Log-log distribution function
·	LOGLOGINV	Log-log random variable
·	LOGLOGISTICDIST	Log-logistic distribution function
·	LOGLOGISTICINV	Log-logistic random variable
·	LR	Linear regression (OLS)
·	LRAIC	Akaike information criterion for a regression
·	LRBIG	Linear regression (OLS) for large data sets
·	LRDFBETA	Observational diagnostics for a regression
·	LRDHATMAT	Diagonal of the hat matrix
·	LRDW	Durbin-Watson test statistic in a regression
·	LREGLS	Estimated generalized least squares (EGLS)
·	LRGLS	Generalized least squares (GLS)
·	LRGQ	Goldfeld-Quandt test statistic for a regression
·	LROBS	Regression observation count and degrees of freedom
·	LRPARTCORR	Partial correlation function in a regression
·	LRRESID	Residuals and predicted values in a regression
·	LRRHO	Autocorrelation coefficient in the errors of a regression
·	LRRIDGE	Ridge regression
·	LRSEMICORR	Semi-partial correlation function in a regression
·	LRSIC	Schwarz information criterion for a regression
·	LRT	Likelihood ratio test in univariate or multivariate autoregression estimation
·	LRVIF	Variance inflation factor for a regression
·	LRWLS	Weighted least squares (WLS)
·	MAE	Mean absolute error
·	MAHANGLE	Mahalanobis angle of a data matrix
·	MAPE	Mean absolute percent error
·	MCENTER	Centering matrix of a specified dimension
·	MCHOL	Choleski factorization of an $nx(n+p)$ matrix, ()
·	MCOFACTOR	Cofactor of a square matrix
·	MCOR	Correlation matrix
·	MCOV	Covariance matrix
·	MDAPE	Median absolute percent error
·	MDET	Determinant of a square matrix
·	MDIAG	Diagonalize a vector or matrix
·	MDIST	Squared Mahalanobis distance of two data matrices
·	MEDAVG	Median average
·	MEQCORR	Equicorrelation matrix of a specified dimension
·	MEVAL	Eigenvalues of a square matrix
·	MEXP	Exponential power of a matrix
·	MGINVERSE	Generalized inverse of a matrix
·	MIDEN	Identity matrix
·	MINV	Inverse of a square matrix
·	MIP	Inner product of two matrices
·	MJ	Matrix of 1s
·	MKRON	Kronecker multiply two matrices
·	MLEBETA	Beta MLE of parameter(s)
·	MLEBINOM	Binomial MLE of parameter(s)
·	MLEDEXPON	Double Exponential MLE of parameter(s)
·	MLEEXPON	Exponential MLE of parameter(s)
·	MLEGAMMA	Gamma MLE of parameter(s)
·	MLEGEOM	Geometric MLE of parameter(s)
·	MLELOGISTIC	Logistic MLE of parameter(s)
·	MLELOGLOG	Log-Log MLE of parameter(s)
·	MLELOGLOGISTIC	Log-Logistic MLE of parameter(s)
·	MLELOGNORM	Lognormal MLE of parameter(s)
·	MLENEGBIN	Negative Binomial MLE of parameter(s)
·	MLENORM	Normal MLE of parameter(s)

·	MLEPARETO	Pareto MLE of parameter(s)
·	MLEPOISSON	Poisson MLE of parameter(s)
·	MLEUNIFORM	Uniform MLE of parameter(s)
·	MLEWEIB	Weibull MLE of parameter(s)
·	MNORM	Norm of a matrix
·	MOMBETA	Beta MOM of parameter(s)
·	MOMBINOM	Binomial MOM of parameter(s)
·	MOMDEXPON	Double Exponential MOM of parameter(s)
·	MOMEXPON	Exponential MOM of parameter(s)
·	MOMGAMMA	Gamma MOM of parameter(s)
·	MOMGEOM	Geometric MOM of parameter(s)
·	MOMLOGISTIC	Logistic MOM of parameter(s)
·	MOMLOGLOG	Log-Log MOM of parameter(s)
·	MOMLOGLOGISTIC	Log-Logistic MOM of parameter(s)
·	MOMLOGNORM	Lognormal MOM of parameter(s)
·	MOMNEGBIN	Negative Binomial MOM of parameter(s)
·	MOMNORMNormal MOM of parameter(s)	
·	MOMPARETO	Pareto MOM of parameter(s)
·	MOMPOISSON	Poisson MOM of parameter(s)
·	MOMUNIFORM	Uniform MOM of parameter(s)
·	MOMWEIB	Weibull MOM of parameter(s)
·	MORTH	Orthoganalize a matrix
·	MOVAVG	Moving average
·	MPROD	Multiply two or more conformable matrices
·	MRANK	Rank of a matrix
·	MRECH	Row Echelon Form of a matrix
·	MRRECH	Reduced row echelon form of a matrix
·	MSE	Mean squared error
·	MSQRT	Factor a square, symmetric matrix
·	MSTACK	Stack two or more matrices
·	MSVD	Singular value decomposition of a matrix
·	MSWEEP	Sweep a square matrix on a diagonal element
·	MTOEP	Column vector to a Toeplitz matrix
·	MTPNORM	Modified two-piece normal random variable
·	MTPNORMDIST	Modified two-piece normal distribution function
·	MTRACE	Trace of a square matrix
·	MULTINOMDIST	Multinomial distribution function
·	MULTINOMINV	Multinomial random vector
·	MULTSORT	Sort a matrix by a specified column
·	MVCHT	LRT for complete homogeneity of multiple data matrices
·	MVCV	Multivariate coefficient of variation
·	MVEMP	Multivariate empirical random vector
·	MVEMPIRICAL	Multivariate empirical random vector
·	MVEPANDIST	Multivariate Epanechnikov distribution function
·	MVLOGNORM	Multivariate lognormal random vector
·	MVNORM	Multivariate normal random vector
·	MVNORMDIST	Multivariate normal distribution function
·	MVPDENSITY	Percentile based on a multivariate kernel density estimator
·	MVTINV	Multivariate student's t random variable
·	NEGBINOMINV	Negative binomial random variable
·	NORM	Normal random variable
·	NORMAD	Anderson Darling statistic for test of normality
·	NORMCHI	Chi-squared statistic for a test of normality
·	NORMCVM	Cramer von Mises statistic for test of normality
·	NORMKS	Kolmogorov Smirnov statistic for test of normality
·	NORMSW	Shapiro-Wilks statistic for test of normality
·	OPT	Find an iterative optimum solution
·	PARETO	Pareto random variable
·	PARETODIST	Pareto distribution function
·	PAUTOCORR	Partial autocorrelation function for a univariate time series
·	PDENSITY	Percentile based on a Kernel density estimator
·	PERTDIST	Project evaluation and review technique (PERT) distribution function
·	PERTINV	Project evaluation and review technique (PERT) random variable
·	PNORM	Power normal random variable

·	PNORMDIST	Power normal distribution function
·	POISSONINV	Poisson random variable
·	PROBIT	Probit binary response regression
·	QUANTILE	Find the quantile of an empirical CDF given the probability
·	RANDSORT	Randomly sort a vector
·	RANDWALK	Generate a random walk series
·	RANKCORREL	Rank correlation of two data series
·	REVERSE	Reverse the order of a vector
·	RINTEGRAL	Riemann integral of a bounded function
·	RMSE	Root mean squared error
·	RUSD	Rank correlation matrix
·	SCENARIO	Return a value associated with different scenarios in a simulation
·	SEMICIRCDIST	Semicircle distribution function
·	SEMICIRCINV	Semicircle random variable
·	SEQ	Sequence of numbers
·	SIMETARCR	Returns copyright information for Simetar
·	STRETCHIT	Matrix to a vector
·	TEMPIRICAL	Truncated empirical random variable
·	TGAMMADIST	Truncated gamma distribution function
·	TGAMMAINV	Truncated gamma random variable
·	THEILU2	Theil's U2 statistic for forecasts
·	TNORM	Truncated normal random variable
·	TNORMDIST	Truncated normal distribution function
·	TPNORM	Two-piece normal random variable
·	TPNORMDIST	Two-piece normal distribution function
·	TRANS	Transpose a matrix
·	TRIANGLE	Triangle random variable
·	TRIANGLEDIST	Triangle distribution function
·	TSDECOMP	Time series decomposition
·	TWEIBDIST	Truncated Weibull distribution function
·	TWEIBINV	Truncated Weibull random variable
·	TWOSLS	Two stage least squares (2SLS)
·	UNBOXCOX	Convert a Box-Cox transformed value back to the original level
·	UNIFORM	Uniform random variable
·	UNIFORMDIST	Uniform distribution function
·	USND	Uncorrelated standard normal deviate
·	UUSD	Uncorrelated uniform standard deviate
·	VARAIC	Akaike information criterion in univariate or multivariate autoregression models
·	VAREST	Univariate or multivariate autoregression estimation function
·	VARLRT	Likelihood ratio test in univariate or multivariate autoregression estimation
·	VARRESID	Predictions & residuals in univariate or multivariate autoregression models
·	VFORMULA	View the formula in the referenced cell
·	WAPE	Weighted absolute percent error
·	WBNAME	Return the name of the workbook
·	WEIBDIST	Weibull distribution function
·	WEIBINV	Weibull random variable
·	WILKSLDIST	Approximate cdf of the Wilks' Lambda random variable
·	WILKSLINV	Wilks Lambda random variable
·	WISHDIST	Wishart distribution function
·	WISHINV	Wishart random matrix
·	WSNAME	Return the name of the worksheet

21.0 Cross Reference of Functions and Demonstration Programs

Topic	Demonstration Program Name
ANOVA	Data Analysis Tools Demo.xls
ANOVA test	Hypothesis Tests Demo.xls
AR and VAR models estimated	Time Series Demo.xls
AR model dynamic probabilistic forecast	Time Series Forecasting Demo.xls
AR model estimation	Time Series Functions Demo.xls
AR model estimation	Time Series Analysis Tools Demo.xls
ARLAG function	Time Series Functions Demo.xls
ARSCHWARZ function	Time Series Functions Demo.xls
AUTOCORR function	Time Series Functions Demo.xls
Additive seasonal decomposition forecasting with cycle	Seasonal Decomposition Forecasts Demo.xls
Additive seasonal decomposition forecasting without cycle	Seasonal Decomposition Forecasts Demo.xls
Amortize land debts	Farm Simulator Demo.xls
Amortize loans with monthly payments	Monthly Payments Demo.xls
Augmented Dickie Fuller test	Time Series Functions Demo.xls
Autocorrelation coefficients	Time Series Forecasting Demo.xls
Autocorrelation coefficients	Time Series Analysis Tools Demo.xls
Autocorrelation coefficients	Time Series Demo.xls
Autocorrelation test	Time Series Functions Demo.xls
BERNOULLI function application	Simulate Alternative Distributions Demo.xls
BOXCOX function	Data Analysis Tools Demo.xls
BOXCOXEXP function	Data Analysis Tools Demo.xls
Bad (singular) correlation/covariance matrix	Bad Correlation Matrix Demo.xls
Bernoulli distribution	Conditional Probability Distributions Demo.xls
Bernoulli distribution	Probability Distribution Demo.xls
Bernoulli distribution	Probability Distributions Demo.xls
Bernoulli distribution parameter estimation	Trend Regression to Reduce Risk Demo.xls
Beta distribution	Probability Distribution Demo.xls
Beta distribution	Simulate All Probability Distributions Demo.xls
Beta distribution	Probability Distributions Demo.xls
Bingo	Games of Chance Demo.xls
Binomial distribution	Simulate All Probability Distributions Demo.xls
Binomial distribution	Probability Distributions Demo.xls
Boot strap simulation	Simulate All Probability Distributions Demo.xls
Bootstrap for singular matrix	Bad Correlation Matrix Demo.xls
Bootstrap simulation	Probability Distributions Demo.xls
Bootstrapper distribution	Probability Distribution Demo.xls
Box plot chart of risky alternatives	Analysis of Simulation Results Demo.xls
Box's M test	Data Analysis Tools Demo.xls
Box-Cox transformation	Data Analysis Tools Demo.xls
Business model	Deterministic Demo.xls
Simplified Business model of net returns	Business Model with Risk Demo.xls
CDF chart of random variables	Analysis of Simulation Results Demo.xls
CDFDEV function	Univariate Parameter Estimator Demo.xls
CDFs for 12 distributions	Test Parameters Demo.xls
CV stationarity for Normal distributions	CV Stationarity Normal Demo.xls
CV stationarity for empirical distributions	CV Stationarity Empirical Demo.xls
Capital Investment Analyzer ©	Net Present Value Internal Rate of Return Demo.xls
Cauchy distribution	Probability Distribution Demo.xls
Cauchy distribution	Simulate All Probability Distributions Demo.xls
Cauchy distribution	Probability Distributions Demo.xls
Centering a matrix	Matrix Operation Tools Demo.xls
Centering matrix of size n	Matrices Demo.xls
Chi-Squared distribution	Probability Distribution Demo.xls
Chi-Squared distribution	Probability Distributions Demo.xls
Chi-Squared test	Data Analysis Tools Demo.xls
Choleski decomposition of a covariance matrix	Parameter Estimation Tools Demo.xls
Coin toss	Games of Chance Demo.xls
Column vector to a matrix	Matrix Operation Tools Demo.xls
Compare means and variance for multivariate distributions	Hypothesis Tests Demo.xls

Compare means and variance for univariate distributions	Hypothesis Tests Demo.xls
Compare means for two distributions -- ANOVA	Hypothesis Tests Demo.xls
Compare means for two series	Data Analysis Tools Demo.xls
Compare two data series	Analysis of Simulation Results Demo.xls
Compare two multivariate distributions	Validation Tests Demo.xls
Compare two tests -- t and F tests	Data Analysis Tools Demo.xls
Compare two univariate distributions	Validation Tests Demo.xls
Complete Homogeneity test	Data Analysis Tools Demo.xls
Concatenate data from two locations	Matrix Operation Tools Demo.xls
Concatenate two matrices	Matrices Demo.xls
Conditional distribution for simulating sales bonus	Conditional Probability Distributions Demo.xls
Conditional probability distributions	Conditional Probability Distributions Demo.xls
Confidence interval for seasonal index	Seasonal Analysis Demo.xls
Confidence intervals for Multiple Regression forecasts	Probabilistic OLS Forecasts Demo.xls
Convert a matrix to a vector	Matrices Demo.xls
Convert a vector to a matrix	Matrices Demo.xls
Corporate federal income taxes	Income Tax Demo.xls
Corporate income taxes	Farm Simulator Demo.xls
Correct for CV non-stationarity Normal distribution	CV Stationarity Normal Demo.xls
Correlated standard normal deviates	Probability Distributions Demo.xls
Correlated uniform standard deviates	Probability Distributions Demo.xls
Correlating Normal, Empirical, Uniform in a MV distribution	Multivariate Mixed Probability Distribution Demo.xls
Correlation matrix calculated	Data Analysis Tools Demo.xls
Correlation matrix t test of rho vs. zero	Data Analysis Tools Demo.xls
Correlation matrix test simulated vs. historical	Data Analysis Tools Demo.xls
Correlation matrix validation for MV distributions	Validation Tests Demo.xls
Correlation significance test	Hypothesis Tests Demo.xls
Correlation test of MVE method	Multivariate Empirical Distribution Demo.xls
Cosine distribution	Simulate All Probability Distributions Demo.xls
Cosine distribution	Probability Distributions Demo.xls
Cost of a project with risk	Project Management Demo.xls
Covariance matrix calculated	Data Analysis Tools Demo.xls
Covariance matrix estimation	Parameter Estimation Tools Demo.xls
Covariance matrix estimation	Matrix Operation Tools Demo.xls
Crop Insurance premium estimation	Insurance Premium Demo.xls
Cumulative distributions for ranking risky alternatives	Stochastic Dominance Demo.xls
Cycle length estimation	Probabilistic Cycle Forecasts Demo.xls
Cyclical decomposition of times series data	Exponential Smoothing Demo.xls
Cyclical decomposition of times series data	Moving Average Demo.xls
Cyclical index	Cyclical Analysis Tools Demo.xls
Cyclical index	Exponential Smoothing Demo.xls
Cyclical index	Moving Average Demo.xls
Cyclical index	Seasonal Analysis Demo.xls
DELNUM function	Data Analysis Tools Demo.xls
DELTEXT function	Data Analysis Tools Demo.xls
DEMPIRICAL function application	Simulate Alternative Distributions Demo.xls
DF Betas	Parameter Estimation Tools Demo.xls
DF function	Time Series Functions Demo.xls
DF function	Time Series Analysis Tools Demo.xls
Decomposition forecasting	Seasonal Decomposition Forecasts Demo.xls
Decomposition forecasts	Seasonal Index Forecasts Demo.xls
Decomposition of a time series	Cyclical Analysis Tools Demo.xls
Delivery time and inventory management	Inventory Management Demo.xls
Determinant of a square matrix	Matrices Demo.xls
Determinate of a square matrix	Matrix Operation Tools Demo.xls
Deterministic farm model	Deterministic Demo.xls
Deterministic simulation NPV and IROR	Net Present Value Internal Rate of Return Demo.xls
Deterministic simulation model	Cotton Model Demo.xls
Dice	Games of Chance Demo.xls
Dickie Fuller (DF) test	Time Series Functions Demo.xls
Dickie Fuller test	Time Series Forecasting Demo.xls
Dickie Fuller test	Time Series Analysis Tools Demo.xls
Dickie Fuller test	Time Series Demo.xls
Discrete empirical distribution	Probability Distributions Demo.xls

Monte Carlo sampling method	Latin Hypercube vs Monte Carlo Demo.xls
Monte Carlo sampling method	Latin Hypercube Demo.xls
Moving average forecast	Cyclical Analysis Tools Demo.xls
Moving average forecast	Moving Average Demo.xls
Moving average forecasts	Moving Average Forecasts Demo.xls
Moving average seasonal index	Seasonal Index Forecasts Demo.xls
Multi peril crop insurance analyzer	Crop Insurance Demo.xls
Multinomial distribution	Simulate All Probability Distributions Demo.xls
Multinomial distribution	Probability Distributions Demo.xls
Multiple Regression forecast stochastic w/ SE of predictions	Probabilistic OLS Forecasts Demo.xls
Multiple Regression forecast stochastic w/ Std Dev	Probabilistic OLS Forecasts Demo.xls
Multiple Regression forecast with stochastic betas	Probabilistic OLS Forecasts Demo.xls
Multiple Regression harmonic and dummy variable regression	Regression for Seasonal Forecasts Demo.xls
Multiple Regression linear trend regression	Trend Forecasts Demo.xls
Multiple Regression multiple regression model	Parameter Estimation Tools Demo.xls
Multiple Regression non-linear trend regression	Trend Forecasts Demo.xls
Multiple Regression probabilistic forecasting	Multiple Regression Forecasts Demo.xls
Multiple Regression regression with restrictions	Parameter Estimation Tools Demo.xls
Multiple Regression to estimate risk for a random variable	Multiple Regression to Reduce Risk Demo.xls
Multiple enterprise business	Business Demo.xls
Multiple enterprise business	Farm Simulator Demo.xls
Multiple enterprise business	Feedlot Demo.xls
Multiple regression	Parameter Estimation Tools Demo.xls
Multiple regression forecasting	Multiple Regression Forecasts Demo.xls
Multiple regression model vs. trend model vs. mean model	Multiple Regression to Reduce Risk Demo.xls
Multiple regression to reduce risk	Trend Regression to Reduce Risk Demo.xls
Multiple regression with probabilistic forecast	Multiple Regression Demo.xls
Multiple year financial statement	Net Present Value Demo.xls
Multiplicative seasonal decomposition forecasting with cycle	Seasonal Decomposition Forecasts Demo.xls
Multiplicative seasonal decomposition forecasting without cycle	Seasonal Decomposition Forecasts Demo.xls
Multiply two matrices	Matrix Operation Tools Demo.xls
Multiply two matrices	Matrices Demo.xls
Multivariate Student's t distribution	Probability Distributions Demo.xls
Multivariate empirical distribution	Multivariate Empirical Distribution Demo.xls
Multivariate empirical distribution	Simulate All Probability Distributions Demo.xls
Multivariate empirical distribution -- 1 and 2 steps	Probability Distributions Demo.xls
Multivariate lognormal distribution	Probability Distributions Demo.xls
Multivariate mixed distribution	Multivariate Mixed Probability Distribution Demo.xls
Multivariate mixed distribution	Simulate All Probability Distributions Demo.xls
Multivariate mixed distribution	Probability Distributions Demo.xls
Multivariate normal distribution	Multivariate Normal Distribution Demo.xls
Multivariate normal distribution	Simulate All Probability Distributions Demo.xls
Multivariate normal distribution -- 1 and 2 steps	Probability Distributions Demo.xls
Multivariate test of two distributions	Data Analysis Tools Demo.xls
NORMAL function application	Simulate Alternative Distributions Demo.xls
NPV	Farm Simulator Demo.xls
NPV	Project Feasibility Demo.xls
NPV - Net Present Value	Investment Management Demo.xls
NPV and IROR simulated for 20 year investment	Net Present Value Internal Rate of Return Demo.xls
NPV for alternative discount rates	Feedlot Demo.xls
NPV optimization for a business	Deterministic Optimal Control Demo.xls
NPV simulated for a business	Net Present Value Demo.xls
Negative binomial distribution	Probability Distribution Demo.xls
Negative binomial distribution	Simulate All Probability Distributions Demo.xls
Negative binomial distribution	Probability Distributions Demo.xls
Negative ending cash reserves	Feedlot Demo.xls
Negative ending cash reserves	Financial Risk Management Demo.xls
Net present value for a risky business	Net Present Value Demo.xls
Net returns for one enterprise	Truncated Normal Distribution Demo.xls
Norm of a square matrix	Matrix Operation Tools Demo.xls
Norm of a square matrix	Matrices Demo.xls
Normal distribution	Probability Distribution Demo.xls
Normal distribution	Probability Distributions Demo.xls
Normal distribution	Test Simetar Demo.xls

Probability-Probability (PP) plot chart	Analysis of Simulation Results Demo.xls
Probit regression	Probit and Logit Demo.xls
Production function with risk	Production Function Demo.xls
Production function with risk	Stochastic Production Function Demo.xls
Production insurance (MPCI)	Financial Risk Management Demo.xls
Project management analysis	Project Management Demo.xls
Project management and evaluation	Project Evaluation Demo.xls
QUANTILE function	Analysis of Simulation Results Demo.xls
Quantile-Quantile (QQ) plot chart	Analysis of Simulation Results Demo.xls
RANDSORT function application	Simulate Alternative Distributions Demo.xls
RMSE	Forecast Errors Demo.xls
RMSE -- Root mean square error	Measuring Forecast Errors Demo.xls
Random sort of objects	Probability Distributions Demo.xls
Random walk distribution	Probability Distributions Demo.xls
Rank insurance strategies	Financial Risk Management Demo.xls
Rank of a matrix	Matrix Operation Tools Demo.xls
Rank of a matrix	Matrices Demo.xls
Rank risky alternatives with SERF	SERF Analysis Demo.xls
Rank risky alternatives with SERF	Simulate Scenarios Demo.xls
Rank risky marketing strategies	Financial Risk Management Demo.xls
Ranking alternative portfolios	Portfolio Analysis Demo.xls
Ranking risky alternatives based on NPV	Net Present Value Demo.xls
Ranking risky alternatives with several methods	SDRF and SERF Ranking Demo.xls
Ranking risky alternatives with several methods	Analysis of Simulation Results Demo.xls
Ranking risky marketing options	Futures and Options Demo.xls
Ranking univariate distributions	Univariate Parameter Estimator Demo.xls
Real rate of return to equity	Investment Management Demo.xls
Regression forecasting	Probabilistic OLS Forecasts Demo.xls
Replacement of machinery compliment by item	Machinery Demo.xls
Residuals from regression to measure risk	Multiple Regression to Reduce Risk Demo.xls
Restricted Multiple Regression estimations	Parameter Estimation Tools Demo.xls
Revenue insurance (CRC)	Financial Risk Management Demo.xls
Reverse the order of a vector	Matrix Operation Tools Demo.xls
Reverse the order of data in a vector	Matrices Demo.xls
Risk premiums for ranking risky alternatives	SDRF and SERF Ranking Demo.xls
Risk premiums for ranking risky alternatives	Analysis of Simulation Results Demo.xls
Risky cost of projects	Project Management Demo.xls
Risky investment analysis	Project Evaluation Demo.xls
Root mean square error -- RMSE	Measuring Forecast Errors Demo.xls
Row echelon for of a matrix	Matrix Operation Tools Demo.xls
Row echelon of a matrix	Matrices Demo.xls
SCENARIO function	Scenario Analysis Demo.xls
SDRF for ranking risky alternatives	Stochastic Dominance Demo.xls
SDRF ranking of risky alternatives	Crop Insurance Demo.xls
SERF and SDRF for ranking risky alternatives	Portfolio Analysis Demo.xls
SERF application	SERF Analysis Demo.xls
SERF ranking of risky alternatives	Crop Insurance Demo.xls
Sampling without replacement	Probability Distributions Demo.xls
Scatter matrix	Matrix Operation Tools Demo.xls
Scenario analysis	Feedlot Demo.xls
Scenario analysis	Analysis of Simulation Results Demo.xls
Scenario analysis	Net Present Value Demo.xls
Scenario analysis of a simple business	Simulate Scenarios Demo.xls
Scenario application to simple profit model	Scenario Analysis Demo.xls
Scenario simulation	Simulate All Probability Distributions Demo.xls
Scenario simulation and ranking	Simulate Scenarios Demo.xls
Schwarz criteria for number of lags	Time Series Forecasting Demo.xls
Schwarz criteria for number of lags	Time Series Analysis Tools Demo.xls
Schwarz criteria for number of lags	Time Series Demo.xls
Schwarz test	Time Series Functions Demo.xls
Seasonal decomposition of monthly & quarterly data	Seasonal Analysis Demo.xls
Seasonal forecast of monthly & quarterly data	Seasonal Analysis Demo.xls
Seasonal index	Seasonal Index Forecasts Demo.xls
Seasonal index	Cyclical Analysis Tools Demo.xls