GAUSSX 10 Update
By Econotron Software, Inc.
www.Econotron.com

GAUSS X ™

Now Available!

New Features and Enhancements:

- ▸ x64 version
- ▸ GAUSS 11 support
- ▸ COPULA - Gaussian copulas
- ▸ CORR - Rank correlation matrix
- ▸ MVRND - Random draws from correlated multivariate distributions
- ▸ Stepwise regression
- ▸ LHS - Latin Hypercube Sample
- ▸ Statlib update

Gaussx 10 has been redesigned for platform compatibility.

Gaussx for Windows runs as a 32-bit Windows application when it runs under a 32-bit version of GAUSS, and as a 64-bit application when run under a 64-bit version of GAUSS. A summary of the new routines implemented since Gaussx 9.0 are described below. A more detailed description of these routines is available at

http://www.econotron.com/gaussx/readme2.htm

A full description of Gaussx is available at

http://www.econotron.com/gaussx

Product Details

▸ 64-bit support

Gaussx will run under either GAUSS x86 or GAUSS x64. In each case Gaussx recognizes which version of GAUSS that is being used, and configures itself accordingly.

▸ GAUSS 11 support

Gaussx has been updated to support the new GAUSS 11 interface, and to incorporate the new functionality of GAUSS 11. Gaussx supports GAUSS 6 through 11.

▸ Copulas

A copula is used in statistics as a general way of formulating a multivariate distribution with a specified correlation structure:

Example:

```
let rmat[3,3] = 1 .5 .2 .5 1 .6 .2 .6 1;
q=copula(1000,rmat,1);
v1=normal_cdfi(q[.,1],0,1);
v2=expon_cdfi(q[.,2],2);
v3=gamma_cdfi(q[.,3],1.5,2.5);
```

q is a 1000x3 copula matrix with a Kendall Tau correlation structure given by rmat. This copula is then used to create three correlated random deviates drawn from the normal, exponential and gamma distributions.

▸ CORR

Computes a correlation matrix for different correlation types - Pearson, Kendall Tau b and Spearman Rank.

MVRND

Creates a matrix of (pseudo) correlated random variables using specified distributions.

Example:

```
dist = "normal" $| "expon" $| "gamma";
let p[3,3] = 0 1 0 0  2 0 0 0  1.5 2.5 0 0;
let rmat[3,4] = 1 .5 .2 .5 1 .6 .2 .6 1;
s = mvrnd(1000, 3, dist, p, rmat, 2);
```

This example creates s, which is a 1000x3 matrix of correlated random variates consisting of the three distributions shown in dist, with the correlation structure specified by the Spearman rank matrix rmat.

STEPWISE

In a situation where there are a large number of potential explanatory variables, STEPWISE can be useful in ascertaining which combination of variables are significant, based on the F statistic. It includes the capability of scaling data, and expanding a given data set to include cross and/or quad terms. This is an exploratory, rather than a rigorous tool.

Example:

```
oplist = { .4 .25 };
indx = stepwise(y~xmat, 0, oplist);
{xnew, xname} = xmat[.,indx];
```

This example shows how a stepwise regression is applied to a matrix of potential explanatory variables xmat, using .4 and .25 for the F statistic probability of entry and exit.

Latin Hypercube Sample - LHS

LHS has the advantage of generating a set of samples that more precisely reflect the shape of a sampled distribution than pure random (Monte Carlo) samples. The Gaussx implementation provides standard LHS, nearly orthogonal LHS, and correlation LHS.

Example:

```
n = 30; k = 6;
fill = 0; ntry = 1000; crit = 2;
dsgn = fill | ntry | crit;
p = lhs(n,k,dsgn);
x = weibull_cdfi(p,1,1.5);
```

In this example, a 30x6 nearly orthogonal Latin Hypercube Sample is derived using the best condition number as the criteria. This creates a 30x6 matrix of probabilities, which are then used to create a set of Weibull distributed variates, each column being orthogonal to every other column.

STATLIB - Statistical Distribution library

The STATLIB library has been updated; it now includes 51 continuous distributions, and 9 discrete distributions. This library can be used independently of Gaussx, or as part of Gaussx - for example in an ML context.

For each of the distributions given below, the following functionality is provided:

fn_llf	likelihood function
fn_pdf	probability density function
fn_cdf	cumulative density function
fn_cdfi	inverse cumulative density function
fn_rnd	random number generator

Continuous functions:

beta	beta4	boxcox
burr	cauchy	chisq
chisq_scaled	erf	expon
f	f_scaled	fatiguelife
fisk	foldednormal	frechet
gamma	ged	gengamma
genlogistic	genpareto	halfnormal
invgamma	invgauss	johnson_sb
johnson_sl	johnson_su	laplace
lev	levy	loggamma
logistic	loglog	lognorm
maxwell	ncchisq	ncf
nct	normal	pareto
pearson	pert	power
rayleigh	reciprocal	sev
skewnormal	students_t	t_scaled
triangular	uniform	vonmises
weibull		

Discrete functions:

Bernoulli	Binomial	Geometric
Hypergeometric	Logarithmic	Negative Binomial
Poisson	Step	Uniform

In the context of ML estimation, the parameters of a particular distribution can be estimated from a set of data, or a parameter can be replaced by a linear or non-linear function, whose parameter can also be estimated. Threshold estimates for distributions where the data is non-negative is also supported.

Example:

```
x = seqa(0,.2,6);
a = 2; b = 4;
p = beta_pdf(x,a,b);

param b0 b1;
 value = .1 1 ;
FRML eq1 v = b0 + b1*x;
FRML eq2 llfn = chisq_llf(y,v);
ML (d,p,i) eq1 eq2
method = nr nr nr;
```

The first example shows pure GAUSS code for estimating the pdf for a beta distribution. The second shows how the parameters of a function which is used to replace a parameter in a distribution can be evaluated.