

# CREATING SIMULATED DATASETS

By G. David Garson

North Carolina State University  
School of Public And International Affairs



@c 2012 by G. David Garson and Statistical Associates Publishing. All rights reserved worldwide in all media. No permission is granted to any user to copy or post this work in any format or any media.

The author and publisher of this eBook and accompanying materials make no representation or warranties with respect to the accuracy, applicability, fitness, or completeness of the contents of this eBook or accompanying materials. The author and publisher disclaim any warranties (express or implied), merchantability, or fitness for any particular purpose. The author and publisher shall in no event be held liable to any party for any direct, indirect, punitive, special, incidental or other consequential damages arising directly or indirectly from any use of this material, which is provided “as is”, and without warranties. Further, the author and publisher do not warrant the performance, effectiveness or applicability of any sites listed or linked to in this eBook or accompanying materials. All links are for information purposes only and are not warranted for content, accuracy or any other implied or explicit purpose. This eBook and accompanying materials is © copyrighted by G. David Garson and Statistical Associates Publishing. No part of this may be copied, or changed in any format, sold, or used in any way under any circumstances other than reading by the downloading individual.

## Contact:

G. David Garson, President  
Statistical Publishing Associates  
274 Glenn Drive  
Asheboro, NC 27205 USA

Email: [gdavidgarson@gmail.com](mailto:g davidgarson@gmail.com)  
Web: [www.statisticalassociates.com](http://www.statisticalassociates.com)

## Table of Contents

Overview.....	4
Random numbers in SPSS .....	4
The SPSS random number generator .....	4
Available distributions .....	5
Creating lists of random numbers in SPSS.....	7
Simulation of Regression with Random Values .....	9
Syntax .....	10
Dataset created in the SPSS data editor.....	10
Output from a sample regression run, significant variables highlighted .....	11
Frequently Asked Questions .....	11
Can I generate cumulative, inverse, and other functions of distributions as well as random variates?.....	11
How can I generate an ID variable?.....	12
Bibliography.....	13

# Creating Simulated Data

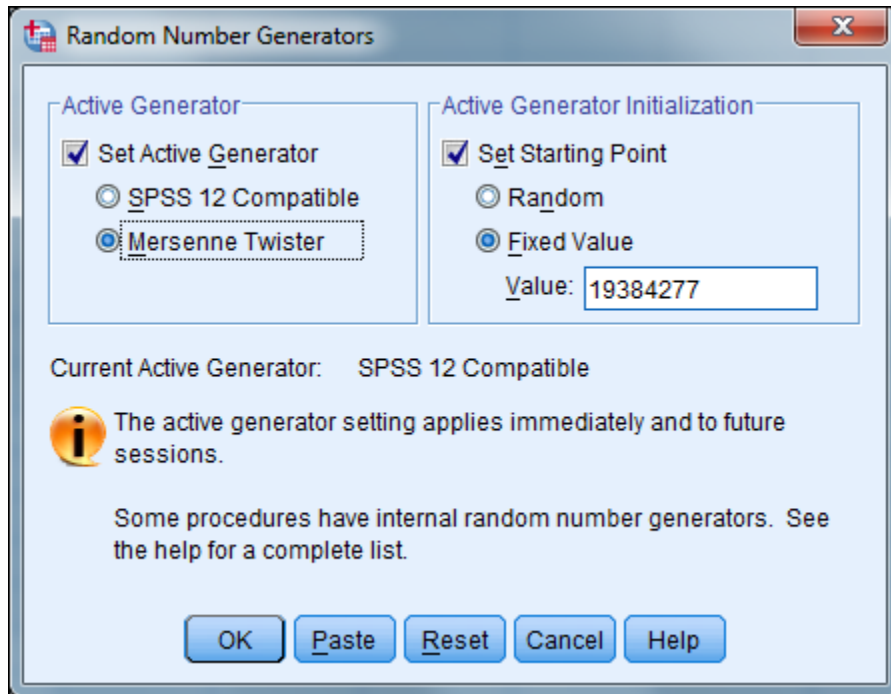
## Overview

Statistical packages such as SPSS can generate new variables reflecting the random variate of any of over a dozen specific distributions. Different distributions require different parameters with the syntax code parentheses, as explained below.

## Random numbers in SPSS

### The SPSS random number generator

The SPSS random number generator is invoked in the SPSS menu system under Transform > Random Number Generators, as illustrated below. The Mersenne twister algorithm is considered more reliable and is used unless replicating results from SPSS version 12 or earlier. Also, the researcher may request a random starting point or may set a fixed value for the starting point. Setting the same fixed value as on an earlier occasion allows the researcher to repeat sequences of pseudorandom numbers. The “Random” selection is the default, however, causing SPSS to automatically change the random number seed whenever a random number is generated for use in transformations such as functions listed below. Initializing the seed to a fixed value is only necessary when it is desired to replicate a sequence of random numbers.



## Available distributions

When creating a list of random numbers, a variety of random variates for specified distributions are available in SPSS:

- `NORMAL(stddev)`. For example, `NORMAL(5)` returns a normally distributed pseudorandom number from a distribution with mean 0 and a standard deviation of 10.
- `RV.NORMAL(mean, stddev)`. For example, `RV.NORMAL(10,4)` returns a normally distributed pseudorandom number from a distribution with a mean of 10 and a standard deviation of 4.
- `UNIFORM(max)`. For example, `UNIFORM(100)` returns a uniformly distributed pseudorandom number between 0 and 100.
- `UNIFORM(min, max)`. `UNIFORM(100, 200)` returns a random value from a uniform distribution with a minimum of 100 and maximum of 200.
- `RV.BERNOULLI(prob)`. `RV,BERNOULLI(.5)` returns a Bernoulli distributed random variate with a .5 probability parameter.
- `RV.BETA(shape1, shape2)`. `RV.BETA(m,n)` returns a random value from a Beta distribution with specified shape parameters.

- RV.BINOM(n, prob). RV.BINOM(100,.25) returns a binomially distributed random variate with 100 trials, each with .25 probability.
- RV.CAUCHY(loc, scale). RV.CAUCHY(loc, scale) returns a random value from a Cauchy distribution with specified location and scale parameters.
- RV.CHISQ(df). RV.CHISQ(45) returns a random value from a chi-square distribution with 45 degrees of freedom.
- RV.EXP(scale). RV.EXP(m) returns a random value from an exponential distribution with specified scale parameter of m.
- RV.F(df1, df2). RV.F(2, 23) returns a random value from an F distribution with 2 and 23 degrees of freedom.
- RV.GAMMA(shape, scale). RV.GAMMA(m, n) returns a random value from a Gamma distribution with specified m shape and n scale parameters.
- RV.GEOM(prob). RV.GEOM(.667) returns a random value from a geometric distribution with a probability of .667.
- RV.HALFNRM(mean, stddev). RV.HALFNRM(100, 10) returns a random value from a half normal distribution with a mean of 100 and a standard deviation of 10
- RV.HYPER(total, sample, hits). RV.HYPER(500, 100, 25) returns a random value from a hypergeometric distribution with 500 total, 100 sample, and 25 hits.
- RV.IGAUSS(loc, scale). RV.IGAUSS(m, n) returns a random value from an inverse Gaussian distribution with m location and n scale parameters.
- RV.LAPLACE(mean, scale). RV.LAPLACE(100, m) returns a random value from a Laplace distribution with a mean of 100 and a scale parameter of m.
- RV.LOGISTIC(mean, scale). RV.LOGISTIC(100, m) returns a random value from a logistic distribution with a mean of 100 and a scale parameter of m.
- RV.LNORMAL(a, b). RV.LNORMAL(a, b) returns a random value from a log-normal distribution with specified parameters.
- RV.NEGBIN(threshold, prob). RV.NEGBIN(m, .5) returns a random value from a negative binomial distribution with a threshold of m and a probability of .5.

- RV.PARETO(threshold, shape). RV.PARETO(m, n) returns a random Pareto-distributed value with a threshold of m and shape parameter of n.
- RV.POISSON(mean). RV.POISSON(.5) returns a Poisson-distributed random value with a mean or rate of .5.
- RV.T(df). RV.T(99) returns a random value from a Student's t distribution with 99 degrees of freedom.
- RV.WEIBULL(a, b). RV.WEIBULL(a, b) returns a random value from a Weibull distribution with the specified parameters.

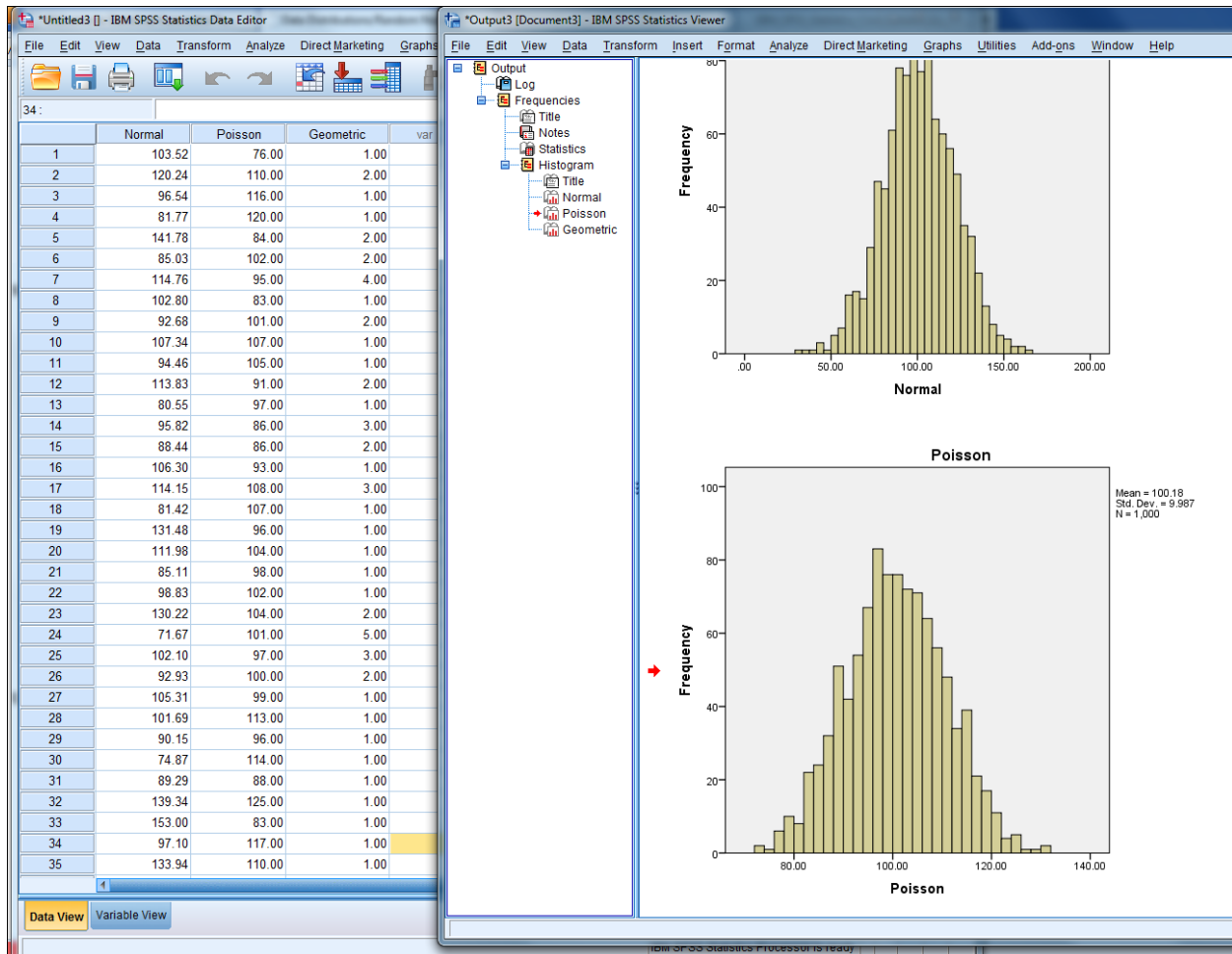
## Creating lists of random numbers in SPSS

Actual creation of random number lists is done in SPSS syntax (command, paste) mode. Below, for example, is SPSS syntax to create individual random variables with normal, Poisson, and geometric distributions.

```
* Create 1,000 cases
* Return normally distributed values with mean = 100 and s.d. = 20.
* Also return Poisson distributed values with a mean of 100.
* Also return random value from a geometric distribution with p = .667.
NEW FILE.
INPUT PROGRAM.
  LOOP #I=1 TO 1000.
    COMPUTE Normal = RV.NORMAL(100,20).
    COMPUTE Poisson = RV.POISSON(100).
    COMPUTE Geometric = RV.GEOM(.667).
  END CASE.
  END LOOP.
END FILE.
END INPUT PROGRAM.
EXECUTE.

* Print out all three histograms but no tables.
FREQUENCIES VARIABLES = ALL
  /HISTOGRAM
  /FORMAT = NOTABLE.
```

This yields results randomly similar to those below, including adding columns to the working dataset and generating the requested graphs.



As a second illustration of using SPSS syntax to create multiple random numbers in a given distribution, the following syntax generates 10 random number variables X1 through X10 with 1,000 cases each from a uniform distribution, from a minimum of 500 to a maximum of 1,000. To change the number of variables, change the "10" in the VECTOR statement and the #J loop. To change the number of cases, change the "1000" in the #I loop. To change the distribution, change the COMPUTE statement.

```

NEW FILE.
INPUT PROGRAM.
  VECTOR X(10).
  LOOP #I = 1 TO 1000.
    LOOP #J = 1 TO 10.
      COMPUTE X(#J) = RV.UNIFORM(500,1000).
    END LOOP.
  END CASE.
END LOOP.

```



```

END FILE.
END INPUT PROGRAM.
EXECUTE.

```

Below is what is created in the working dataset for this run.

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	var
1	958.63	985.72	686.62	528.30	654.68	572.98	890.59	883.54	505.68	575.03	
2	514.46	912.82	643.70	834.24	940.27	948.56	962.99	860.46	554.04	539.50	
3	519.20	639.59	529.94	568.68	723.14	579.68	766.38	651.18	714.15	779.60	
4	860.02	712.54	527.35	877.24	831.12	592.18	590.46	764.36	846.20	975.60	
5	798.91	520.10	681.78	674.80	763.84	564.84	665.68	574.89	800.28	958.16	
6	586.73	545.09	949.27	602.20	743.86	901.65	766.66	958.24	769.98	823.17	
7	956.11	546.42	622.30	505.68	775.85	924.01	653.84	628.43	675.28	638.44	
8	875.78	509.77	699.61	889.71	859.56	854.61	656.95	866.27	516.30	623.49	
9	758.77	564.08	986.31	906.96	809.85	863.93	819.96	617.90	681.14	699.85	
10	502.31	823.59	743.86	682.37	756.44	829.09	539.91	981.19	893.45	663.53	
11	664.81	644.53	853.41	522.59	706.84	681.46	687.63	945.25	989.76	919.36	
12	884.00	730.45	908.94	790.16	521.84	866.72	843.92	607.67	860.13	933.61	
13	596.31	580.99	538.50	864.24	855.07	613.57	536.80	761.43	986.50	641.29	
14	611.40	793.65	663.42	680.63	751.01	666.36	685.39	538.86	836.39	849.03	
15	923.95	989.73	838.81	725.73	880.47	866.08	502.73	858.67	671.63	531.95	
16	523.35	788.48	564.23	790.97	840.11	672.71	730.58	887.26	651.21	514.44	
17	722.24	955.02	689.98	883.91	565.45	512.39	910.84	504.88	785.61	548.68	
18	597.17	685.60	563.79	986.49	531.21	980.85	637.20	924.49	915.88	746.08	
19	536.55	698.06	927.85	985.45	723.17	980.24	889.47	677.99	637.14	524.90	
20	725.05	636.56	817.38	693.70	669.21	856.77	817.58	672.54	674.35	771.23	
21	960.95	779.35	790.77	895.49	943.23	728.08	676.40	780.44	769.16	658.09	
22	664.76	899.34	584.49	697.51	625.16	949.41	611.01	961.00	965.25	876.20	
23	612.96	782.65	842.35	580.01	514.30	599.60	546.63	562.55	682.05	770.22	
24	996.75	684.02	671.60	599.07	739.03	724.23	681.26	628.32	722.01	709.57	
25	995.91	812.44	655.39	811.81	698.95	818.64	734.84	794.65	710.04	621.38	
26	528.98	792.41	850.36	853.51	583.84	973.15	731.78	617.83	912.80	924.30	
27	645.11	826.49	911.32	806.42	760.37	781.36	577.08	731.59	723.96	608.86	

## Simulation of Regression with Random Values

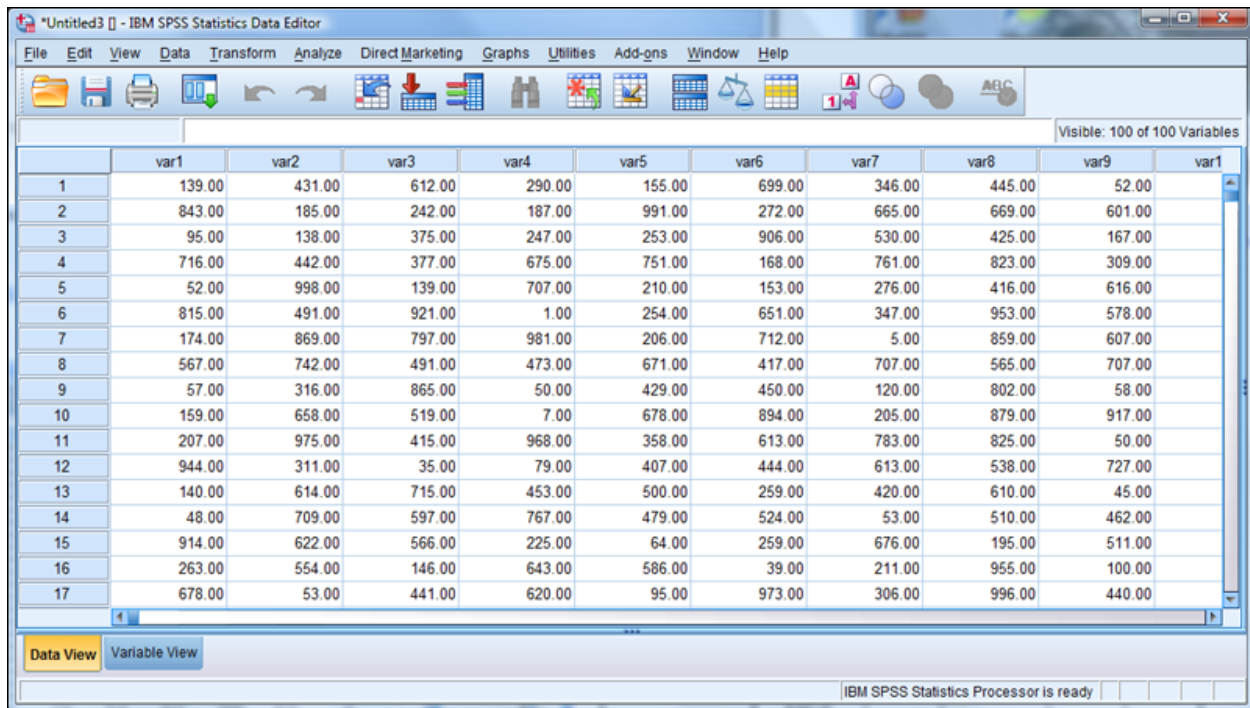
The SPSS syntax below creates a simulated dataset with 100 variables and 1,000 cases. The simulated dataset is then used in an ordinary linear regression, using var100 as the dependent variable and the other 00 as predictors. This dependent variable is assigned a value of 1 for the first 500 cases and a value of 0 for the last 500 cases, as determined by loop #1. Independent variables vary from .00 to 999.00, as determined by the compute statement. The program creates a random value dataset in the SPSS data editor as shown below.

An instructional use of the randomly generated dataset is to demonstrate that approximately 5% of variables will test as significant even for random data. This too is shown in a figure below. For this particular run, five variables tested significant.

## Syntax

```
new file.
input program.
numeric var1 to var100.
vector vctr=var1 to var99.
loop #1=1 to 1000.
if (#1 le 500) var100=1.
if (#1 gt 500) var100=0.
loop #j=1 to 99.
compute vctr(#j)=trunc(1000*rv.uniform(0,1)).
end loop.
end case.
end loop.
end file.
end input program.
execute.
regression vars=var1 to var100
/dependent=var100/enter var1 to var99.
```

## Dataset created in the SPSS data editor



Visible: 100 of 100 Variables

	var1	var2	var3	var4	var5	var6	var7	var8	var9	var10
1	139.00	431.00	612.00	290.00	155.00	699.00	346.00	445.00	52.00	
2	843.00	185.00	242.00	187.00	991.00	272.00	665.00	669.00	601.00	
3	95.00	138.00	375.00	247.00	253.00	906.00	530.00	425.00	167.00	
4	716.00	442.00	377.00	675.00	751.00	168.00	761.00	823.00	309.00	
5	52.00	998.00	139.00	707.00	210.00	153.00	276.00	416.00	616.00	
6	815.00	491.00	921.00	1.00	254.00	651.00	347.00	953.00	578.00	
7	174.00	869.00	797.00	981.00	206.00	712.00	5.00	859.00	607.00	
8	567.00	742.00	491.00	473.00	671.00	417.00	707.00	565.00	707.00	
9	57.00	316.00	865.00	50.00	429.00	450.00	120.00	802.00	58.00	
10	159.00	658.00	519.00	7.00	678.00	894.00	205.00	879.00	917.00	
11	207.00	975.00	415.00	968.00	358.00	613.00	783.00	825.00	50.00	
12	944.00	311.00	35.00	79.00	407.00	444.00	613.00	538.00	727.00	
13	140.00	614.00	715.00	453.00	500.00	259.00	420.00	610.00	45.00	
14	48.00	709.00	597.00	767.00	479.00	524.00	53.00	510.00	462.00	
15	914.00	622.00	566.00	225.00	64.00	259.00	676.00	195.00	511.00	
16	263.00	554.00	146.00	643.00	586.00	39.00	211.00	955.00	100.00	
17	678.00	53.00	441.00	620.00	95.00	973.00	306.00	996.00	440.00	

Data View Variable View

IBM SPSS Statistics Processor is ready

## Output from a sample regression run, significant variables highlighted

Note this is partial output.

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	.890	.286		3.080	.002
	var1	1.147E-005	.000	.007	.199	.842
	var2	4.939E-005	.000	.029	.867	.386
	var3	3.282E-005	.000	.019	.588	.570
	var4	-6.585E-005	.000	-.037	-1.507	.269
	var5	-8.213E-005	.000	-.048	-1.442	.150
	var6	-4.195E-005	.000	-.025	-.736	.462
	var7	-6.584E-006	.000	-.004	-.113	.910
	var8	-2.841E-005	.000	-.017	-.507	.612
	var9	-2.214E-005	.000	-.013	-.381	.703
	var10	4.099E-005	.000	.023	.698	.486
	var11	-6.273E-006	.000	-.037	-1.115	.265
	var12	5.671E-005	.000	.033	.906	.369
	var13	-4.753E-005	.000	-.027	-.816	.415
	var14	7.374E-005	.000	.042	1.276	.202
	var15	-1.387E-005	.000	-.008	-.241	.810
	var16	-3.837E-005	.000	-.021	-.627	.531
	var17	2.698E-005	.000	.015	.454	.650
	var18	-1.899E-005	.000	-.012	-.349	.727
	var19	.000	.000	-.368	-2.823	.043
	var20	-3.897E-005	.000	-.022	-.646	.519
	var21	-4.445E-005	.000	-.026	-.779	.436
	var22	-8.738E-005	.000	-.050	-1.496	.136
	var23	1.179E-005	.000	.007	.211	.833
	var24	-4.587E-005	.000	-.027	-.889	.422
	var25	-8.890E-006	.000	-.005	-.158	.875
	var26	1.389E-006	.000	.001	.024	.981
	var27	6.544E-005	.000	.039	1.183	.245
	var28	.000	.000	-.079	-2.398	.017
	var29	2.551E-005	.000	.014	.434	.664
	var30	3.188E-005	.000	.018	.534	.593
	var31	-6.374E-005	.000	-.037	-1.128	.259

## Frequently Asked Questions

### Can I generate cumulative, inverse, and other functions of distributions as well as random variates?

Yes. SPSS supports the following functions. All operate as prefixes, similar to RV (ex., RV.Poisson, CDF.Poisson, IDF.Poisson, etc.). See SPSS (2007) for the specific parameters required for each distribution.

- CDF Cumulative distribution function.
- CDF.d\_spec(x,a,...) returns a probability p that a variate with the specified distribution (d\_spec) falls below x for continuous functions and at or below x for discrete functions.

- IDF Inverse distribution function. Inverse distribution functions are not available for discrete distributions. An inverse distribution function `IDF.d_spec(p,a,...)` returns a value `x` such that `CDF.d_spec(x,a,...)=p` with the specified distribution (`d_spec`).
- PDF Probability density function.
- `PDF.d_spec(x,a,...)` returns the density of the specified distribution (`d_spec`) at `x` for continuous functions and the probability that a random variable with the specified distribution equals `x` for discrete functions.
- RV Random number generation function.
- `RV.d_spec(a,...)` generates an independent observation with the specified distribution (`d_spec`).
- NCDF Noncentral cumulative distribution function.
- `NCDF.d_spec(x,a,b,...)` returns a probability `p` that a variate with the specified noncentral distribution falls below `x`. It is available only for beta, chi-square, F, and Student's t.
- NPDF Noncentral probability density function.
- `NCDF.d_spec(x,a,...)` returns the density of the specified distribution (`d_spec`) at `x`. It is available only for beta, chi-square, F, and Student's t.
- SIG Tail probability function. A tail probability function `SIG.d_spec(x,a,...)` returns a probability `p` that a variate with the specified distribution (`d_spec`) is larger than `x`. The tail probability function is equal to 1 minus the cumulative distribution function.

## How can I generate an ID variable?

For existing data, run this in syntax to add an ID variable:

```
COMPUTE ID=$CASENUM.
EXECUTE.
```

For a blank data sheet, where you wish to have 100 cases, run this syntax:

```
NEW FILE.
INPUT PROGRAM.
  LOOP #I=1 TO 100.
    COMPUTE ID=$CASENUM.
    END CASE.
  END LOOP.
END FILE.
END INPUT PROGRAM.
```

---

EXECUTE.

## Bibliography

Abramowitz, M. & Stegun, I. A., eds. (1970). *Handbook of mathematical functions*. NY: Dover Publications.

SPSS, Inc. (2007). *SPSS 16 Command Syntax Manual*. Chicago, SPSS. See the section on "Random Variable and Distribution Functions," pp. 69-93.

---

@c 2006, 2008 G. David Garson and Statistical Associates Publishers. Worldwide rights reserved in all languages and on all media. Do not copy or post in any language or format. Last updated: 9/8/2012.

---

## Statistical Associates Publishing Blue Book Series

Association, Measures of  
Assumptions, Testing of  
Canonical Correlation  
Case Studies  
Cluster Analysis  
Content Analysis  
Correlation  
Correlation, Partial  
Correspondence Analysis  
Cox Regression  
Creating Simulated Datasets  
Crosstabulation  
Curve Fitting & Nonlinear Regression  
Data Levels  
Delphi Method  
Discriminant Function Analysis  
Ethnographic Research  
Evaluation Research  
Event History Analysis  
Factor Analysis  
Focus Groups  
Game Theory  
Generalized Linear Models/Generalized Estimating Equations  
GLM (Multivariate), MANOVA, and MANCOVA  
GLM (Univariate), ANOVA, and ANCOVA  
GLM Repeated Measures  
Grounded Theory  
Hierarchical Linear Modeling/Multilevel Analysis/Linear Mixed Models  
Integrating Theory in Research Articles and Dissertations  
Kaplan-Meier Survival Analysis  
Latent Class Analysis  
Life Tables  
Literature Reviews  
Logistic Regression  
Log-linear Models,  
Longitudinal Analysis  
Missing Values Analysis & Data Imputation  
Multidimensional Scaling

Multiple Regression  
Narrative Analysis  
Network Analysis  
Ordinal Regression  
Parametric Survival Analysis  
Partial Least Squares Regression  
Participant Observation  
Path Analysis  
Power Analysis  
Probability  
Probit Regression and Response Models  
Reliability Analysis  
Resampling  
Research Designs  
Sampling  
Scales and Standard Measures  
Significance Testing  
Structural Equation Modeling  
Survey Research  
Two-Stage Least Squares Regression  
Validity  
Variance Components Analysis  
Weighted Least Squares Regression

**Statistical Associates Publishing**  
**<http://www.statisticalassociates.com>**  
**[sa.publishers@gmail.com](mailto:sa.publishers@gmail.com)**