

New functions for Random samples generation using Stata 15

G. Aguilera-Venegas, J.L. Galán-García,
M.Á. Galán-García, Y. Padilla-Domínguez,
P. Rodríguez-Cielos

University of Málaga, Spain

The 25th UK Stata Conference

5 & 6 September 2019. London

Contents

- 1 Generating random samples from Statistical Distributions
 - Authors' Background
 - Random sample generation using Stata
- 2 Pros and cons of current functions and commands
- 3 Our approach
 - Our commands
 - Comparisons
 - Examples
- 4 Conclusions

Authors' Background

- Random samples generators using CAS (Computer Algebra Systems)
 - Derive
 - Maxima
- Random samples generators using Stata 13
- A very important application of generating random samples: Simulations
 - Accelerated Time Simulations (ATS)
 - Traffic control (GRAM, ATISMART, ATISMART+)
 - Baggage handling (ATISBAT)
 - In progress: ATS in biological and medical applications
 - Probabilistic Cellular Automata (PCAEGOL)

Random sample generation using Stata

- Build-in Stata 16 functions

- `rbeta`, `rbinomial`, `rcauchy`, `rchi2`, `rexponential`, `rgamma`,
`rhipergeometric`, `rigaussian`, `laplace`, `rlogistic`, `rnbinomial`,
`rnormal`, `rpoisson`, `rt`, `runiform`, `runiformint`, `rweibull`,
and `rweibullph`

- Users' contributions

- `rndwei`, `rndexp`, `rndivg`, `rndlog`, `rndlgn`, `rndf`, `rndchi`,
`rndt`, `rndnbx`, `rndbb`, `rndpoi`, ...
 - `rsample`

Pros and cons of current functions and commands

- Pros
 - Stata functions are fast
 - `rsample` works for generic distributions
 - `rsample` optionally plots the generated sample
- Cons
 - Stata functions only for specific distributions
 - Stata functions do not plot the generated sample
 - `rsample` very slow when the size is high
 - `rsample` needs the user to introduce suitable limits
 - The size in `rsample` cannot be easily changed

Our commands

- Include new distributions not considered in Stata functions
- Are fast even for high sizes
- Work with suitable limits automatically computed
- Can easily change the size of the sample
- Optionally plot the generated sample
- Optionally compute the Median Squared Error
- Display time spent in the generation
- `scauchy`, `ssexponential`, **`slognormal`**, `snormal`, **`spareto`**,
`sweibull`, `sbinomial`, `suniformint`

New characteristics of `Our` commands

- Other continuous and discrete distributions in progress
- A general function to deal with all considered distributions is also in progress
- Optionally choose among our algorithm, Stata function or `rsample`
- Therefore, the previous advantages are now available for Stata functions and `rsample`:
 - Plot the generated sample
 - Suitable limits automatically computed
 - Easily change the size of the sample
 - Compute the Median Squared Error
 - Display time spent in the generation

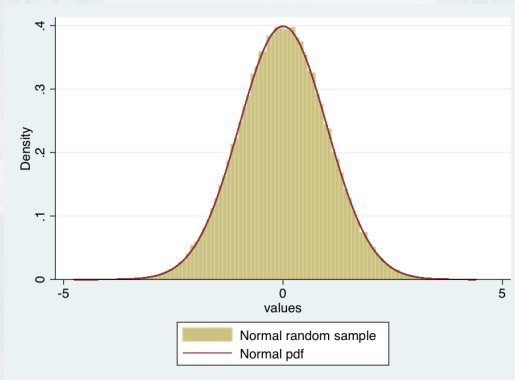
Comparisons

Distribution	Command	Time	Error	Plot
Normal(0,1)	rnormal	1.150e-07	1.030e-06	No
	snormal	1.360e-07	9.772e-07	Yes
	rsample	.00044102	.00001524	Yes
Pareto(8,1)	rpareto	Not available in Stata functions		
	spareto	1.090e-07	9.739e-07	Yes
	rsample	.00044182	.00029966	Yes

Examples

- `snormal 10000000`
- `snormal 100000, pl(1)`

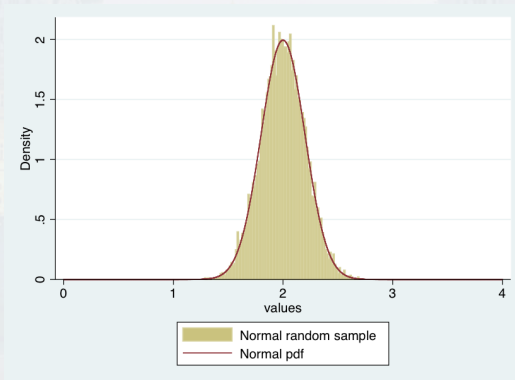
Examples



Examples

- `snormal 10000000`
- `snormal 100000, pl(1)`
- `snormal 100000, mse(1)`
- `snormal 10000, m(2) s(0.2) le(0) ri(4) mse(1) pl(1) nr(10)`

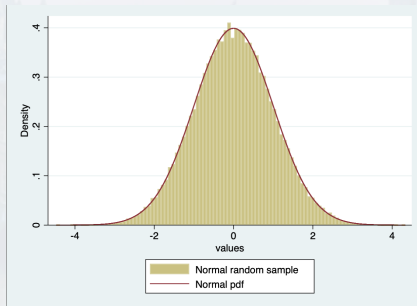
Examples



Examples

- `snormal 10000000`
- `snormal 100000, pl(1)`
- `snormal 100000, mse(1)`
- `snormal 10000, m(2) s(0.2) le(0) ri(4) mse(1) pl(1) nr(10)`
- `snormal 100000, me(2) mse(1) pl(1)`

Examples



```
. snormal 100000, me(2) mse(1) pl(1)
number of observations (_N) was 0, now 100,000
The mean of the mean squared errors is 1.153e-07 and it is stored in r(mmse)
The generated sample is stored in variable sample.
Total time: 4.671 and it is stored in r(time)
Mean time for getting a value of the sample: .00004671 and it is stored in r(mtime)
```

Conclusions

- New commands for random numbers generation from distributions not available in Stata
- Same time order in computation as build-in stata functions
- Deal with our algorithm, the stata functions or `rsample` (optionally)
- Computation of media squared error (optionally)
- Display mean time spend (optionally specifying the number of iterations)
- Plot the generated random sample (optionally)
- Computation of suitable limits automatically (user can change them)
- Great improvement in the time, error and default bounds regarding `rsample`

New functions for Random samples generation using Stata 15

G. Aguilera-Venegas, J.L. Galán-García,
M.Á. Galán-García, Y. Padilla-Domínguez,
P. Rodríguez-Cielos

University of Málaga, Spain

The 25th UK Stata Conference

5 & 6 September 2019. London