

`set rng` — Set which random-number generator (RNG) to use

[Description](#)[Syntax](#)[Remarks and examples](#)[Reference](#)[Also see](#)

Description

`set rng` determines which random-number generator (RNG) Stata's [random-number functions](#) and commands will use.

Syntax

```
set rng { default | mt64 | mt64s | kiss32 }
```

Remarks and examples

stata.com

Remarks are presented under the following headings:

[Introduction](#)

[Random-number generators in Stata](#)

Introduction

By default, Stata uses the 64-bit Mersenne Twister (`mt64`) RNG. `mt64s` is a stream RNG based on the 64-bit Mersenne Twister. Earlier versions of Stata used the 32-bit KISS (keep it simple stupid) (`kiss32`) RNG.

With `set rng default` (the default), code running under [version control](#) will automatically use the appropriate RNG—`mt64` in Stata 14 and later and `kiss32` for earlier code.

The scope of `set rng` is the Stata session, do-file, or program in which `rng` is set.

Unless you want to simultaneously draw random numbers in separate instances of Stata, we recommend that you do not change Stata's default behavior for its RNGs. See [\[R\] set rngstream](#) for an introduction to simultaneously drawing random numbers in separate instances of Stata.

See [\[FN\] Random-number functions](#), [\[R\] set seed](#), and [\[R\] set rngstream](#) for more information.

Random-number generators in Stata

The default RNG in Stata is the 64-bit Mersenne Twister. See [Matsumoto and Nishimura \(1998\)](#) for more details. The default RNG in Stata 13 and earlier versions was George Marsaglia's 32-bit KISS generator (G. Marsaglia, 1994, pers. comm.). The KISS generator is still available under version control or via `set rng`. Multiple independent random-number streams (based on the 64-bit Mersenne Twister) are also supported for use in multiple simultaneous instances of Stata; see [\[R\] set rngstream](#) for more information on this. The abbreviations `mt64`, `kiss32`, and `mt64s` are used, respectively, to specify these three generators in Stata commands and functions.

So far, we have discussed two ways you can specify the RNG: with `set rng` and through version control. Another way to specify the RNG is with functions and system parameters explicitly named after the generators. In fact, all random-number functions have variants that are explicitly named after each generator, using the generator abbreviation as the suffix. For example, `runiform_mt64()`, `runiform_kiss32()`, and `runiform_mt64s()` are variants of `runiform()` for each generator. Similarly, we have `rnormal_mt64()`, `rnormal_kiss32()`, `rnormal_mt64s()`, etc.

The system parameters `seed` and `rngstate` also have variants explicitly named after each generator: `seed_mt64`, `seed_kiss32`, `seed_mt64s`, `rngstate_mt64`, `rngstate_kiss32`, and `rngstate_mt64s`.

For example, here is how you can use functions and parameters specific to `mt64` to set the seed, generate random numbers, preserve a state, generate more numbers, and restore the previously preserved state:

```
. set seed_mt64 482637
. generate u = runiform_mt64()
. local state = c(rngstate_mt64)
. generate l = rlogistic_mt64()
. set rngstate_mt64 'state'
```

Note that calling functions and setting parameters specific to, say, `kiss32`, will not change the current RNG, the seed of the current RNG, or the state of the current RNG—unless the current RNG is `kiss32`.

Reference

Matsumoto, M., and T. Nishimura. 1998. Mersenne Twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation* 8: 3–30. <https://doi.org/10.1145/272991.272995>.

Also see

[R] [set](#) — Overview of system parameters

[R] [set rngstream](#) — Specify the stream for the stream random-number generator

[R] [set seed](#) — Specify random-number seed and state

[FN] [Random-number functions](#)

[P] [version](#) — Version control

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.

