Package 'qrng'

June 11, 2016

Version 0.0-3

Encoding UTF-8

Title (Randomized) Quasi-Random Number Generators

Description Functionality for generating (randomized) quasi-random numbers in high dimensions.

Author Marius Hofert [aut, cre], Christiane Lemieux [aut]

Maintainer Marius Hofert <marius.hofert@uwaterloo.ca>

Depends R (>= 3.0.0)

Imports

Suggests

Enhances

License GPL-2 | GPL-3

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-06-11 22:26:54

R topics documented:

qrng *Compute Quasi-Random Sequences*

Description

Computing Korobov, generalize Halton and Sobol quasi-random sequences.

Usage

```
korobov(n, d, generator, randomize = FALSE)
ghalton(n, d, method = c("generalized", "halton"))
sobol(n, d, randomize = FALSE)
```

Arguments

n	Number n of points to be generated ≥ 2.
d	Dimension d.
generator	A numeric of length d or length 1 (in which case it is appropriately extended to length d). All numbers must be in $\{1, \ldots, n\}$ and must be (coercible to) integers.
randomize	A logical indicating whether the point set should be randomized (for sobol(): a digital shift).
method	A character string indicating which sequence is generated, generalized Halton or (plain) Halton.

Details

Note that these procedures call fast C code. The following restrictions apply:

korobov() n,d must be $\leq 2^{31} - 1$.

ghalton() n must be $\leq 2^{32} - 1$ and d must be ≤ 360.

sobol() n must be $\leq 2^{31} - 1$ and d must be ≤ 360.

The choice of parameters for korobov() is crucial for the quality of this quasi-random sequence (only basic sanity checks are conducted). For more details, see l'Ecuyer and Lemieux (2000).

The generalized Halton sequence uses the scrambling factors of Faure and Lemieux (2009).

Value

korobov() and ghalton() return an (n, d)-matrix; for $d = 1$ an n-vector is returned.

Author(s)

Marius Hofert and Christiane Lemieux

References

Faure, H., Lemieux, C. (2009). Generalized Halton Sequences in 2008: A Comparative Study. *ACM-TOMACS* **19**(4), Article 15.

l'Ecuyer, P., Lemieux, C. (2000). Variance Reduction via Lattice Rules. *Stochastic Models and Simulation*, 1214–1235.

Lemieux, C., Cieslak, M., Luttmer, K. (2004). RandQMC User's guide. See https://www.math.uwaterloo.ca/~clemieux/randq

Examples

```
n <- 1021 # prime
d <- 4 # dimension

## Korobov's sequence
generator <- 76 # see l'Ecuyer and Lemieux
u <- korobov(n, d = d, generator = generator)
pairs(u, gap = 0, pch = ".", labels = as.expression(
      sapply(1:d, function(j) bquote(italic(u[.(j)])))))

## Randomized Korobov's sequence
set.seed(271)
u <- korobov(n, d = d, generator = generator, randomize = TRUE)
pairs(u, gap = 0, pch = ".", labels = as.expression(
      sapply(1:d, function(j) bquote(italic(u[.(j)])))))

## Generalized Halton sequence (randomized by definition)
set.seed(271)
u <- ghalton(n, d)
pairs(u, gap = 0, pch = ".", labels = as.expression(
      sapply(1:d, function(j) bquote(italic(u[.(j)])))))

## Randomized Sobol sequence (with digital shift)
set.seed(271)
u <- sobol(n, d, randomize = TRUE)
pairs(u, gap = 0, pch = ".", labels = as.expression(
      sapply(1:d, function(j) bquote(italic(u[.(j)])))))
```

Index