Package 'mcreplicate'

July 22, 2025

Description Multi-core replication function to make it easier to do fast
Monte Carlo simulation. Based on the mcreplicate() function from the
'rethinking' package. The 'rethinking' package requires installing 'rstan',
which is onerous to install, while also not adding capabilities to this
function.

Title Multi-Core Replicate

Version 0.1.2

Date 2021-05-29

Imports parallel

License AGPL (>= 3)

Encoding UTF-8

RoxygenNote 7.1.1

Suggests testthat (>= 3.0.0), covr

Config/testthat/edition 3

NeedsCompilation no

Author Christopher Gandrud [aut, cre],
Olivier Binette [ctb],
AUTUMN41 [ctb]

Maintainer Christopher Gandrud <christopher.gandrud@gmail.com>

Repository CRAN

Date/Publication 2021-06-20 06:20:02 UTC

Contents

`mc_replicate`   *Multi-core replicate.*

Description

Use multiple cores for repeated evaluation of an expression. This also works on Windows using a parallel socket cluster.

Usage

```
mc_replicate(
  n,
  expr,
  mc.cores = detectCores(),
  cluster,
  varlist,
  envir,
  packages,
  refresh = 0.1
)
```

Arguments

n	integer; the number of replications.
expr	the expression (a language object, usually a call) to evaluate repeatedly.
mc.cores	number of cores to use.
cluster	logical. If TRUE then clustering, rather than forking, is used to replicate the specified function in parallel. Note: if you are using Windows, only cluster is available.
varlist	Only used on Windows! Character vector of variable names to export on each worker. Default is all variables in the current environment which do not begin with a ".". See clusterExport for more information.
envir	Only used on Windows! Environment from which to export variables. Default is the environment from which this function was called. See clusterExport for more information.
packages	Only used on Windows! Environment from which to export variables. Default is all loaded packages. See clusterExport for more information.
refresh	Not on Windows! status update refresh interval

Value

A vector, matrix, or list of length n.

Source

Modified from: Richard McElreath (2020). rethinking: Statistical Rethinking book package. R
package version 2.13. https://github.com/rmcelreath/rethinking

Examples

```
one_sim <- function(n = 100, control_prob = 0.1, rel_effect = 0.01) {
  treat_prob <- control_prob + (control_prob * rel_effect)
  cy <- rbinom(n = n, size = 1, prob = control_prob)
  ty <- rbinom(n = n, size = 1, prob = treat_prob)
  mean(ty) - mean(cy)
  }

  mc_replicate(10, one_sim(), mc.cores = 2)

  # On Windows, when no particular packages or additional variables are needed
# mc_replicate(10, one_sim(), , mc.cores = 2, packages = NULL,
#              varlist = "one_sim", envir = environment())
```

Index