Package 'vntrs'

December 21, 2023

Title Variable Neighborhood Trust Region Search

Version 0.1.1

Description An implementation of the variable neighborhood trust region algorithm Bierlaire et al. (2009) ``A Heuristic for Nonlinear Global Optimization'' <doi:10.1287/ijoc.1090.0343>.

Imports trust

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.3

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://loelschlaeger.de/vntrs/

NeedsCompilation no

Author Lennart Oelschläger [aut, cre] (<https://orcid.org/0000-0001-5421-9313>)

Maintainer Lennart Oelschläger <lennart.oelschlaeger@uni-bielefeld.de>

Repository CRAN

Date/Publication 2023-12-21 11:10:02 UTC

R topics documented:

check_controls *Check controls*

Description

This function checks the input `controls`.

Usage

```
check_controls(controls)
```

Arguments

`controls`	Either `NULL` or a named list with the following elements. Missing elements are set to the default values in parentheses.

- `init_runs` (`5`): The number of initial searches.
- `init_min` (`-1`): The minimum argument value for the random initialization.
- `init_max` (`1`): The maximum argument value for the random initialization.
- `init_iterlim` (`20`): The number of iterations for the initial searches.
- `neighborhoods` (`5`): The number of nested neighborhoods.
- `neighbors` (`5`): The number of neighbors in each neighborhood.
- `beta` (`0.05`): A non-negative weight factor to account for the function's curvature in the selection of the neighbors. If `beta = 0`, the curvature is ignored. The higher the value, the higher the probability of selecting a neighbor in the direction of the highest function curvature.
- `iterlim` (`1000`): The maximum number of iterations to be performed before the local search is terminated.
- `tolerance` (`1e-6`): A positive scalar giving the tolerance for comparing different optimal arguments for equality.
- `time_limit` (`NULL`): The time limit in seconds for the algorithm.

Value

The checked and filled list `controls`.

vntrs *Variable neighborhood trust region search*

Description

This function performs variable neighborhood trust region search.

Usage

```
vntrs(f, npar, minimize = TRUE, controls = NULL, quiet = TRUE, seed = NULL)
```

Arguments

f	A function that computes value, gradient, and Hessian of the function to be optimized and returns them as a named list with elements value, gradient, and hessian.
npar	The number of parameters of f.
minimize	If TRUE, f gets minimized. If FALSE, maximized.
controls	Either NULL or a named list with the following elements. Missing elements are set to the default values in parentheses.

- init_runs (5): The number of initial searches.
- init_min (-1): The minimum argument value for the random initialization.
- init_max (1): The maximum argument value for the random initialization.
- init_iterlim (20): The number of iterations for the initial searches.
- neighborhoods (5): The number of nested neighborhoods.
- neighbors (5): The number of neighbors in each neighborhood.
- beta (0.05): A non-negative weight factor to account for the function's curvature in the selection of the neighbors. If beta = 0, the curvature is ignored. The higher the value, the higher the probability of selecting a neighbor in the direction of the highest function curvature.
- iterlim (1000): The maximum number of iterations to be performed before the local search is terminated.
- tolerance (1e-6): A positive scalar giving the tolerance for comparing different optimal arguments for equality.
- time_limit (NULL): The time limit in seconds for the algorithm.

quiet	If TRUE, progress messages are suppressed.
seed	Set a seed for the sampling of the random starting points.

Value

A data frame. Each row contains information of an identified optimum. The first npar columns "p1",...,"p<npar>" store the argument values, the next column "value" has the optimal function values and the last column "global" contains TRUE for global optima and FALSE for local optima.

References

Bierlaire et al. (2009) "A Heuristic for Nonlinear Global Optimization" doi:10.1287/ijoc.1090.0343.

Examples

```
rosenbrock <- function(x) {
  stopifnot(is.numeric(x))
  stopifnot(length(x) == 2)
  f <- expression(100 * (x2 - x1^2)^2 + (1 - x1)^2)
  g1 <- D(f, "x1")
  g2 <- D(f, "x2")
  h11 <- D(g1, "x1")
  h12 <- D(g1, "x2")
```

```
  h22 <- D(g2, "x2")
  x1 <- x[1]
  x2 <- x[2]
  f <- eval(f)
  g <- c(eval(g1), eval(g2))
  h <- rbind(c(eval(h11), eval(h12)), c(eval(h12), eval(h22)))
  list(value = f, gradient = g, hessian = h)
}
vntrs(f = rosenbrock, npar = 2, seed = 1, controls = list(neighborhoods = 1))
```

Index