Package ‘ao’

February 29, 2024

Title Alternating Optimization
Version 0.3.3
Description Alternating optimization of (high-dimensional) functions is an iterative procedure for optimizing jointly over all parameters by alternately optimizing parameter subsets.

BugReports https://github.com/loelschlaeger/ao/issues
License GPL-3
Encoding UTF-8
RoxygenNote 7.3.1
Imports checkmate, cli, oeli, utils
Suggests knitr, rmarkdown, testthat (>= 3.0.0)
Config/testthat/edition 3
VignetteBuilder knitr
Depends R (>= 4.0.0), optimizeR (>= 1.0.4)
NeedsCompilation no
Author Lennart Oelschläger [aut, cre]
  (<https://orcid.org/0000-0001-5421-9313>)
Maintainer Lennart Oelschläger <oelschlaeger.lennart@gmail.com>
Repository CRAN
Date/Publication 2024-02-29 21:12:46 UTC

R topics documented:
ao .............................................................. 2

Index 4

1
Alternating Optimization

Description

This function performs alternating optimization of the function \( f \).

Usage

\[
\text{ao}( \quad f, \\
p, \\
..., \\
\text{partition} = \text{as.list}(1: \text{length}(p)), \\
\text{base.optimizer} = \text{optimizerR::Optimizer}\$\text{new}(\"\text{stats::optim}\"), \\
\text{iterations} = 10, \\
\text{tolerance} = 1e-06, \\
f\_partition = \text{vector}(\text{mode} = \"\text{list}\", \text{length} = \text{length}(\text{partition})), \\
\text{joint.end} = \text{FALSE}, \\
\text{verbose} = \text{FALSE} 
)
\]

Arguments

- **f**: A function to be optimized, returning a single numeric. The first argument of \( f \) must be a numeric of the same length as \( p \) followed by any other arguments specified by the \(...\) argument.
- **p**: A numeric vector, the starting parameter values for the optimization.
- **...**: Additional arguments to be passed to \( f \).
- **partition**: A list of vectors of indices of \( p \), specifying the partition of the parameter vector in the alternating optimization process. The default is \text{as.list}(1: \text{length}(p)), i.e. each parameter is optimized separately. Parameter indices can be members of multiple subsets.
- **base.optimizer**: An Optimizer object, which can be created via \text{Optimizer}. It numerically solves the optimization problems in the partitions.
- **iterations**: An integer, the maximum number of iterations through partitions before the alternating optimization process is terminated. Can also be Inf, in which case tolerance is responsible for the termination. The default is 10.
- **tolerance**: A non-negative numeric. The alternating optimization terminates prematurely (i.e., before iterations is reached) if the euclidean distance between the current estimate and the one from the last iteration is smaller than tolerance. The default is 1e-6.
- **f\_partition**: A list of the same length as partition. The \( i \)-th element can be a function that computes the value of \( f \) for the \( i \)-th parameter set defined in partition. The function must be of the form function(theta_part, theta_rest, ...), where
• \( \theta_{\text{part}} \) receives the parameter set for the current partition (this argument can be named differently),
• \( \theta_{\text{rest}} \) receives the remaining parameters (this argument must be named \( \theta_{\text{rest}} \)),
• \ldots\) receives the additional arguments to \( f \). Alternatively, it can be NULL, in which case \( f \) is used.

\textbf{joint_end} \quad \text{If TRUE, the parameter set is optimized jointly after the alternating optimization process is terminated. The default is FALSE.}

\textbf{verbose} \quad \text{If TRUE, full tracing details are printed during the alternating optimization process. The default is FALSE.}

\textbf{Value}

A list with the elements

• \( \text{estimate} \), the optimal parameter vector found,
• \( \text{value} \), the value of \( f \) at \( \text{estimate} \),
• \( \text{sequence} \), a \texttt{data.frame} of the function values, estimates and computation times in the single iterations and partitions,
• \( \text{seconds} \), the overall computation time in seconds.

\textbf{Examples}

```r
# definition of the Himmelblau function

# alternating minimization separately for x_1 and x_2
# parameter restriction: -5 \leq x_1, x_2 \leq 5
ao(
  f = himmelblau, p = c(0, 0), partition = list(1, 2), iterations = Inf,
  base_optimizer = optimizeR::Optimizer$new(
    which = "stats::optim", lower = -5, upper = 5, method = "L-BFGS-B"
  )
)
```

Index

ao, 2
Optimizer, 2