Package: pwlapprox2d (via r-universe)

November 2, 2026

Title Approximates Univariate Continuous Functions Through Piecewise Linear Regression

Version 0.1.0

Description Allows users to find a piecewise linear regression approximation to a given continuous univariate function within a specified error tolerance. Methods based on Warwicker and Rebennack (2025) ``Efficient continuous piecewise linear regression for linearising univariate non-linear functions'' <doi:10.1080/24725854.2023.2299809>.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports nloptr

Suggests knitr, rmarkdown

VignetteBuilder knitr

Config/pak/sysreqs cmake

Repository https://jwarwicker.r-universe.dev

Date/Publication 2025-11-18 10:22:18 UTC

RemoteUrl https://github.com/jwarwicker/pwlapprox2d

RemoteRef HEAD

RemoteSha a2a93909ec3a749925b6c0be915893b661fd4c72

Contents

adaptive_pwl_fit 2
optimize_main 3

Index **4**

`adaptive_pwl_fit` *Adaptive Piecewise Linear Approximation of a Continuous Function*

Description

Approximates a continuous function f on a domain $[a, b]$ by adaptively discretizing the domain and building a piecewise linear (PWL) envelope until the maximum error between the PWL and f is within a given tolerance.

Usage

```
adaptive_pwl_fit(
  f,
  domain,
  tol = 0.001,
  max_iter = 50,
  initial_points = 5,
  smallconst = 1e-04
)
```

Arguments

`f`	A continuous function \(f(x) \) to approximate.
`domain`	Numeric vector of length 2 specifying the interval $[a, b]$.
`tol`	Numeric tolerance for maximum allowed approximation error (default 1e-3).
`max_iter`	Maximum number of refinement iterations (default 20).
`initial_points`	Initial number of discretization points (default 5).
`smallconst`	Numeric small constant used in building PWL envelope (default 1e-4).

Value

A list with components:

PWL A matrix of piecewise linear segments: slope, intercept, lower bound, upper bound.

data The final discretization points (x, y) used in fitting.

max_error Maximum absolute error between \(f \) and the PWL approximation.

Examples

```
f <- function(x) log(x)
domain <- c(1, 10)
res <- adaptive_pwl_fit(f, domain, tol = 1e-4, initial_points = 10, smallconst = 0.01)

cat("x,y\n")
for(i in 1:nrow(res$data)) {
  cat(paste(res$data[i, 1], res$data[i, 2], sep = ","), "\n")
}
```

`optimize_main` *Core Optimization Function*

Description

Core Optimization Function

Usage

```
optimize_main(
  choice = 1,
  accuracy = 0.01,
  init_points = 50,
  max_iter = 100,
  verbose = TRUE
)
```

Arguments

`choice`	Integer (1-6) selecting which built-in function to optimize
`accuracy`	Desired precision
`init_points`	Initial number of sampling points (minimum 2)
`max_iter`	Maximum number of iterations
`verbose`	Logical. Print progress.

Value

A list containing the optimization result

Index

adaptive_pwl_fit, 2

optimize_main, 3