Package 'lamW'

May 21, 2021

Type Package

Title Lambert-W Function

Version 2.1.0

Date 2021-05-20

Description Implements both real-valued branches of the Lambert-W function (Corless et al, 1996) <doi:10.1007/BF02124750> without the need for installing the entire GSL.

License BSD_2_clause + file LICENSE

Depends R (>= 3.0.2)

Imports Rcpp, RcppParallel (>= 4.3.20)

LinkingTo Rcpp, RcppParallel (>= 4.3.20)

SystemRequirements GNU make

Suggests testthat, covr

NeedsCompilation yes

URL https://github.com/aadler/lamW

BugReports https://github.com/aadler/lamW/issues

Config/testthat/edition 3

Author Avraham Adler [aut, cph, cre] (<https://orcid.org/0000-0002-3039-0703>)

Maintainer Avraham Adler <Avraham.Adler@gmail.com>

Repository CRAN

Date/Publication 2021-05-21 05:00:02 UTC

R topics documented:

`lamW-package` *Lambert-W Function*

Description

Implements both real-valued branches of the Lambert-W function (Corless et al, 1996) <doi:10.1007/BF02124750> without the need for installing the entire GSL.

Details

The DESCRIPTION file:

Package:	lamW
Type:	Package
Title:	Lambert-W Function
Version:	2.1.0
Date:	2021-05-20
Authors@R:	c(person(given="Avraham", family="Adler",role=c("aut", "cph", "cre"), email="Avraham.Adler@gr
Description:	Implements both real-valued branches of the Lambert-W function (Corless et al, 1996) <doi:10.1007
License:	BSD_2_clause + file LICENSE
Depends:	R (>= 3.0.2)
Imports:	Rcpp, RcppParallel (>= 4.3.20)
LinkingTo:	Rcpp, RcppParallel (>= 4.3.20)
SystemRequirements:	GNU make
Suggests:	testthat, covr
NeedsCompilation:	yes
URL:	https://github.com/aadler/lamW
BugReports:	https://github.com/aadler/lamW/issues
Config/testthat/edition:	3
Author:	Avraham Adler [aut, cph, cre] (<https://orcid.org/0000-0002-3039-0703>)
Maintainer:	Avraham Adler <Avraham.Adler@gmail.com>

Index of help topics:

Author(s)

NA Maintainer: NA

`lambertW` *Lambert-W Function*

Description

The Lambert-W function is defined as the multivalued inverse of the function $z = f(W) = We^W$. The function has two branches. The principal branch is defined on the interval $[-\frac{1}{e}, \infty)$ and is always greater than or equal to-1. The second branch is defined on the interval $[-\frac{1}{e}, 0)$ and is always less than or equal to-1. The real-valued function is not defined for values less than $-\frac{1}{e}$.

Usage

```
lambertW0(x)
lambertWm1(x)
```

Arguments

x	vector of values

Details

The Lambert-W function is defined for all real $x \geq -\frac{1}{e}$. It has two values in the interval $(-\frac{1}{e}, 0)$. The values strictly greater than -1 are assigned to the "principal" branch, also referred to as W_0, and the values strictly less than -1 are assigned to the "secondary" branch, referred to as W_{-1}. For non-negative x, only the principal branch exists as the other real-valued branch approaches negative infinity as x approaches 0. The algorithms used to calculate the values predominantly follow those in the reference with some simplifications. There are many applications in which the Lambert-W function is useful, such as combinatorics, physics, and hydrology. The interested reader is directed to the references for more detail.

Value

Both functions return the appropriate values in the intervals for which they are defined. Outside of those intervals, they will return `NaN`, except that `lambertW0(Inf)` will return its limit `Inf` and `lambertWm1(0)` will return its limit `-Inf`.

Author(s)

Avraham Adler <Avraham.Adler@gmail.com>

References

Corless, R. M., Gonnet, G. H., Hare, D. E., Jeffrey, D. J., Knuth, D. E. 1996 "On the Lambert W function", *Advances in Computational Mathematics*, **5**, 329–359, Springer <doi:10.1007/BF02124750>

Fritsch, F. N.; Shafer, R. E. & Crowley, W. P. 1973 "Solution of the transcendental equation (we^w = x)", *Communications of the ACM*, **16**, 123–124, Association for Computing Machinery (ACM) <doi:10.1145/361952.361970>

See Also

This package provides similar functionality to the Lambert functions in the **gsl** package without having to obtain or install the entire GSL.

Examples

```
lambertW0(exp(1))      ## Should equal 1, as 1 * exp(1) = e.
lambertW0(0)           ## Should equal 0, as 0 * exp(0) = 0.
lambertW0(-exp(-1))    ## Should equal -1.
lambertWm1(-exp(-1))   ## Should also equal -1.
A <- -2 * exp(-2)
lambertWm1(A)          ## Should equal -2
```

Index