Package ‘wowa’

May 24, 2022

Type Package

Title Weighted Ordered Weighted Average

Version 1.0.2

Date 2022-05-22

Maintainer Gleb Beliakov <gleb@deakin.edu.au>

Author Gleb Beliakov [aut, cre],
Daniela Calderon [aut]

Description Introduce weights into Ordered Weighted Averages and extend bivariate means based on n-ary tree construction. Please refer to the following:
G. Beliakov, J.J. Dujmovic (2016) <doi:10.1016/j.ins.2015.10.040>,

License LGPL-3

LazyData FALSE

Imports Rcpp (>= 1.0.0)

LinkingTo Rcpp

RoxygenNote 5.0.1

NeedsCompilation yes

Copyright Gleb Beliakov

Repository CRAN

Date/Publication 2022-05-24 08:30:01 UTC

R topics documented:

wowa .......................................................... 2
wowa.ImplicitWOWA ...................................... 3
wowa.OWA .................................................. 4
wowa.WAM .................................................. 5
wowa.WAn .................................................. 5
Various weighted multivariate extensions of bivariate and OWA functions, including implicit, quantifier-based and binary tree based WOWA.

Usage

```r
wowa()
```

Details

Lists the functions implemented in this package.

Value

`output` No return value, called for printing only.

Author(s)

Gleb Beliakov, Daniela L. Calderon, Deakin University

References


Examples

```r
wowa()
```
**wowa.ImplicitWOWA**  
**Implicit Weighted OWA Computation Function**

**Description**

Function for Calculating implicit Weighted OWA function

**Usage**

```r
wowa.ImplicitWOWA(x, p, w, n)
```

**Arguments**

- `x` The vector of inputs
- `p` The weights of inputs x
- `w` The OWA weightings vector
- `n` Dimension of the vector x

**Value**

- output The value of the Impicit Weighted OWA

**Author(s)**

Gleb Beliakov, Daniela L. Calderon, Deakin University

**References**


**Examples**

```
n <- 4
example <- wowa.ImplicitWOWA(c(0.3, 0.4, 0.8, 0.2), c(0.3, 0.25, 0.3, 0.15),
                             c(0.4, 0.35, 0.2, 0.05), n)
example
```

---

**wowa.OWA**  
*Ordered weighted average function*

---

**Description**

Function for computing the ordered weighted averages

**Usage**

```
wowa.OWA(n, x, w)
```

**Arguments**

- `n`: Dimension of the vector `x`
- `x`: The vector of inputs
- `w`: The OWA weights

**Value**

- `output`: The value of the ordered weighted average.

**Author(s)**

Gleb Beliakov, Daniela L. Calderon, Deakin University

**References**


**Examples**

```
n <- 4
wowa.OWA(n, c(0.3, 0.4, 0.8, 0.2), c(0.4, 0.35, 0.2, 0.05))
```
**wowa.WAM**

**WAM computation**

**Description**

Function for calculating the Weighted Arithmetic Mean

**Usage**

```
wowa.WAM(n, x, w)
```

**Arguments**

- **n**: Dimension of the array `x`
- **x**: The vector of inputs
- **w**: The vector of weights

**Value**

- **output**: The value of the WAM function

**Author(s)**

Gleb Beliakov, Daniela L. Calderon, Deakin University

**References**


**Examples**

```
n <- 4
wowa.WAM(n, c(0.3,0.4,0.8,0.2), c(0.3,0.25,0.3,0.15) )
```

**wowa.WAn**

**Extension of binary averaging**

**Description**

Function for calculating a binary tree multivariate extension of a binary averaging function

**Usage**

```
wowa.WAn(x, w, n, Fn, L)
```
**Arguments**

- **x**: Vector of inputs
- **w**: The weightings vector
- **n**: Dimension of the array x (and w)
- **Fn**: Bivariate symmetric mean that is extended to n arguments
- **L**: The number of levels of the binary tree (see docs)

**Value**

- **output**: The output is Weighted n-variate mean extending Fn

**Author(s)**

Gleb Beliakov, Daniela L. Calderon, Deakin University

**References**


wowa.weightedf

Weighted extension of the OWA function

Description

Function for extending order weighted averages and other multivariate symmetric functions

Usage

wowa.weightedf(x, p, w, n, Fn, L)

Arguments

- x: The vector of inputs
- p: The weights of inputs x
- w: The OWA weightings vector
- n: The dimension of the vector x
- Fn: Base n-variate symmetric function defined in R
- L: The number of levels of the n-ary tree (see docs)

Value

output: The output is the weighted ordered weighted average.

Author(s)

Gleb Beliakov, Daniela L. Calderon, Deakin University

References

Examples

```r
Fn <- function(n, x, w) {
    out <- 0.0
    for(i in 1:n) out <- out + x[i] * w[i]
    #print(out)
    return(out)
}

n <- 4

example <- wowa.weightedf(c(0.3, 0.4, 0.8, 0.2),
                           c(0.3, 0.25, 0.3, 0.15),
                           c(0.4, 0.35, 0.2, 0.05),
                           n, Fn, 10)

example
```

**wowa.weightedOWAQuantifier**

*WOWA value computation Function*

**Description**

Function for calculating the value of the quantifier-based WOWA function

**Usage**

```r
wowa.weightedOWAQuantifier(x, p, w, n, spl)
```

**Arguments**

- **x**
  - The vector of inputs
- **p**
  - The weights of inputs x
- **w**
  - The OWA weightings vector
- **n**
  - The dimension of the array x
- **spl**
  - A structure that keeps the spline knots and coefficients computed in weighte-
    dOWAQuantifierBuild function

**Value**

- **output**
  - The output is quantifier-based WOWA value

**Author(s)**

Gleb Beliakov, Daniela L. Calderon, Deakin University
References


Examples

```r
n <- 4
pweights=c(0.3,0.25,0.3,0.15);
wweights=c(0.4,0.35,0.2,0.05);
tempspline <- wowa.weightedOWAQuantifierBuild(pweights, wweights , n)
wowa.weightedOWAQuantifier(c(0.3,0.4,0.8,0.2), pweights, wweights, n, tempspline)
```

**wowa.weightedOWAQuantifierBuild**

*RIM quantifier of the Weighted OWA function*

**Description**

Function for building the RIM quantifier of the Weighted OWA function.

**Usage**

`wowa.weightedOWAQuantifierBuild(p, w, n)`

**Arguments**

- `p` The weights of inputs `x`
- `w` The OWA weightings vector
- `n` The dimension of the vectors `p,w`

**Value**

- `output` A structure which has fields: spl, which keeps the spline knots and coefficients for later use in `wowa.weightedOWAQuantifier`, and `Tnum`, the number of knots in the monotone spline
Author(s)

Gleb Beliakov, Daniela L. Calderon, Deakin University

References


Examples

```r
n <- 4
pweights=c(0.3,0.25,0.3,0.15);
wwweights=c(0.4,0.35,0.2,0.05);
tspanline <- wowa.weightedOWAQuantifierBuild(pweights,wwweights,n)
wowa.weightedOWAQuantifier(c(0.3,0.4,0.8,0.2), pweights, wweights, n, tspline)
```
Index

* Implicit Wowa
  wowa.ImplicitWowa, 3
* OWA
  wowa.OWA, 4
* WAM
  wowa.WAM, 5
* WAn
  wowa.WAn, 5
* Wowa
  wowa, 2
  wowa.ImplicitWowa, 3
  wowa.WAM, 5
  wowa.WAn, 5
  wowa.weightedOWAQuantifier, 8
  wowa.weightedOWAQuantifierBuild, 9
* weightedOWAQuantifier
  wowa.weightedOWAQuantifier, 8
  wowa.weightedOWAQuantifierBuild, 9
* weighteddf
  wowa.weighteddf, 7
* wowa
  wowa, 2
  wowa, 2
  wowa.ImplicitWowa, 3
  wowa.OWA, 4
  wowa.WAM, 5
  wowa.WAn, 5
  wowa.weighteddf, 7
  wowa.weightedOWAQuantifier, 8
  wowa.weightedOWAQuantifierBuild, 9