Package 'ZEP'

May 7, 2026

Type Package

Title Procedures Related to the Zadeh's Extension Principle for Fuzzy Data

Version 0.3.1

Description
Procedures for calculation, plotting, animation, and approximation of the outputs for fuzzy numbers (see A.I. Ban, L. Coroianu, P. Grzegorzewski ``Fuzzy Numbers: Approximations, Ranking and Applications" (2015)) based on the Zadeh's Extension Principle (see de Barros, L.C., Bassanezi, R.C., Lodwick, W.A. (2017) <doi:10.1007/978-3-662-53324-6_2>).

License GPL-3

NeedsCompilation no

Imports FuzzyNumbers, methods, animation, grDevices

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 3.5.0)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Author Maciej Romaniuk [cre, aut] (ORCID: <https://orcid.org/0000-0001-9649-396X>, (Systems Research Institute Polish Academy of Sciences, Newelska 6, 01-447 Warszawa, Poland)),
Abbas Parchami [aut] (ORCID: <https://orcid.org/0000-0002-0593-7324>, (Mahani Math Center, Afzalipour Research Institute, Shahid Bahonar University of Kerman, Kerman, Iran, Department of Statistics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Iran)),
Przemyslaw Grzegorzewski [aut] (ORCID: <https://orcid.org/0000-0002-5191-4123>, (Faculty of Mathematics and Information Science, Warsaw University of Technology, Koszykowa 75, 00-662 Warsaw, Poland))

Maintainer Maciej Romaniuk <mroman@ibspan.waw.pl>

Repository CRAN

Date/Publication 2025-10-30 11:30:02 UTC

Contents

AnimateListZEP 2
AnimateZEP 4
ApplyZEP 5
approximationMehodsInside 7
DpqDistance 7
FuzzyApproximation 8
PlotZEP 9

Index **12**

`AnimateListZEP` *Function for animation of the whole list of fuzzy numbers*

Description

`AnimateListZEP` animates the whole list consisting of fuzzy numbers.

Usage

```
AnimateListZEP(
  listOfValues,
  FUN,
  knots = 10,
  grid = TRUE,
  approximation = FALSE,
  method = "NearestEuclidean",
  sleep = 1,
  ...
)
```

Arguments

`listOfValues`	List of the input fuzzy numbers.
`FUN`	Function used for the input fuzzy number with the help of the Zadeh's principle.
`knots`	Number of the alpha-cuts used during calculation of the output.
`grid`	If `TRUE`, then additional grid is plotted.
`approximation`	If `TRUE`, the approximated output is calculated.
`method`	The selected approximation method.
`sleep`	Interval between frames in the animation.
`...`	Additional parameters passed to other functions.

Details

The function takes the list of input fuzzy numbers `listOfValues` (which should be described by one of the classes from `FuzzyNumbers` package) and applies the function `FUN` using the Zadeh's principle. The output is given as animation of consecutive fuzzy numbers or their approximations (when `approximation` is set to `TRUE` and the respective `method` is selected). To properly find the output, value of `FUN` is calculated for many alpha-cuts of `listOfValues`. The number of these alpha-cuts is equal to `knots` (plus 2 for the support and the core). If the approximation is used, then the approximated fuzzy number is shown with green line.

The input fuzzy number from a list `listOfValues` should be given by fuzzy number described by classes from `FuzzyNumbers` package.

Value

The figures are animated: the series of the input and output fuzzy numbers (for the Zadeh's principle and the applied function) or their approximation (if selected).

Examples

```
library(FuzzyNumbers)

# prepare list of fuzzy numbers

a <- seq(0,5,by=1)

fuzzyList <- list()

for (i in 1:length(a)) {

 fuzzyList[[i]] <- TrapezoidalFuzzyNumber(i,i+1,2*i+1,3*i+1)

}

# check the list
fuzzyList

# now some animations for various functions and then with approximation
AnimateListZEP(fuzzyList,FUN=function(x) x^2)

AnimateListZEP(fuzzyList,FUN=function(x) sin(x))

AnimateListZEP(fuzzyList,FUN=function(x) x^3+1,approximation = TRUE)
```

AnimateZEP *Animate output for the Zadeh's principle*

Description

AnimateZEP applies the selected function to a fuzzy number using the Zadeh's principle, and then animates the output.

Usage

```
AnimateZEP(
  value,
  FUN,
  knots = 10,
  approximation = FALSE,
  method = "NearestEuclidean",
  sleep = 0.05,
  ...
)
```

Arguments

value	Input fuzzy number.
FUN	Function used for the input fuzzy number with the help of the Zadeh's principle.
knots	Number of the alpha-cuts used during calculation of the output.
approximation	If TRUE, the approximated output is calculated.
method	The selected approximation method.
sleep	Interval between frames in the animation.
...	Additional parameters passed to other functions.

Details

The function takes the input fuzzy number value (which should be described by one of the classes from FuzzyNumbers package) and applies the function FUN using the Zadeh's principle. The output is given by a fuzzy number or its approximation (when approximation is set to TRUE and the respective method is selected). To properly find the output, value of FUN is calculated for many alpha-cuts of value. The number of these alpha-cuts is equal to knots (plus 2 for the support and the core). The output fuzzy number and its approximation are animated for the decreasing value of alpha (i.e., the consecutive alpha-cuts). If the approximation is used, then the approximated fuzzy number is shown with green line.

The input fuzzy number value should be given by fuzzy number described by classes from FuzzyNumbers package.

Value

One (or two) figures are animated: the output fuzzy number (for the Zadeh's principle and the applied function), and its approximation (if selected).

Examples

```
library(FuzzyNumbers)

# prepare complex fuzzy number

A <- FuzzyNumber(-5, 3, 6, 20, left=function(x)
pbeta(x,0.4,3),
right=function(x) 1-x^(1/4),
lower=function(alpha) qbeta(alpha,0.4,3),
upper=function(alpha) (1-alpha)^4)

# animate the output fuzzy number

AnimateZEP(A,FUN=function(x)x^3+2*x^2-1)

# find and animate the approximated output via the Zadeh's principle

AnimateZEP(A,FUN=function(x)x^3+2*x^2-1,approximation=TRUE)
```

ApplyZEP *Function to apply the Zadeh's principle*

Description

ApplyZEP applies the selected function to a fuzzy number using the Zadeh's principle.

Usage

```
ApplyZEP(
  value,
  FUN,
  knots = 10,
  approximation = FALSE,
  method = "NearestEuclidean",
  ...
)
```

Arguments

value	Input fuzzy number.
FUN	Function used for the input fuzzy number with the help of the Zadeh's principle.
knots	Number of the alpha-cuts used during calculation of the output.
approximation	If TRUE, the approximated output is calculated.
method	The selected approximation method.
...	Additional parameters passed to other functions.

Details

The function takes the input fuzzy number value (which should be described by one of the classes from FuzzyNumbers package) and applies the function FUN using the Zadeh's principle. The output is given by a fuzzy number or its approximation (when approximation is set to TRUE and the respective method is selected). To properly find the output, value of FUN is calculated for many alpha-cuts of value. The number of these alpha-cuts is equal to knots (plus 2 for the support and the core).

The input fuzzy number value should be given by fuzzy number described by classes from FuzzyNumbers package.

Value

The output is a fuzzy number described by classes from FuzzyNumbers package (piecewise linear fuzzy number without approximation, various types with the approximation applied).

Examples

```
library(FuzzyNumbers)

# prepare complex fuzzy number

A <- FuzzyNumber(-5, 3, 6, 20, left=function(x)
pbeta(x,0.4,3),
right=function(x) 1-x^(1/4),
lower=function(alpha) qbeta(alpha,0.4,3),
upper=function(alpha) (1-alpha)^4)

# find the output via the Zadeh's principle

ApplyZEP(A,FUN=function(x)x^3+2*x^2-1)

# find the approximated output via the Zadeh's principle

ApplyZEP(A,FUN=function(x)x^3+2*x^2-1,approximation=TRUE)
```

approximationMehodsInside

A vector containing names of the built-in approximation methods.

Description

'approximationMehodsInside' is a vector containing names of the built-in approximation methods.

Usage

```
approximationMehodsInside
```

Format

An object of class `character` of length 3.

Value

This function returns a vector of strings.

Examples

```
# check the names

approximationMehodsInside
```

DpqDistance

Function to calculate D(p,q) distance.

Description

`DpqDistance` calculates the generalized D(p,q) distance between two fuzzy numbers.

Usage

```
DpqDistance(value1, value2, p = 2, q = 1/2)
```

Arguments

`value1`	First fuzzy number.
`value2`	Second fuzzy number.
`p`	Value of the power (and the the root) applied in the distance calculation.
`q`	Value of the weight for the second fuzzy number (for the first one this weight is calculated as `1-q`, respectively).

Details

The function calculates the generalized D(p,q) distance between two fuzzy numbers `value1` and `value2`, where `p` is the value of the applied power, and `q` is the weight between these two fuzzy numbers.

All of the input values should be given by fuzzy numbers described by classes from `FuzzyNumbers` package.

Value

The output is a numerical value (the calculated distance).

Examples

```
library(FuzzyNumbers)

# prepare two fuzzy numbers

A <- TrapezoidalFuzzyNumber(0,1,2,3)

B <- TrapezoidalFuzzyNumber(1,3,4,6)

# calculate the distance

DpqDistance (A,B)
```

FuzzyApproximation *Function for approximation with the help of methods other than in FuzzyNumbers package*

Description

`FuzzyApproximation` approximates the given fuzzy number.

Usage

```
FuzzyApproximation(value, method = "ExpectedValueCore", piecewise = FALSE, ...)
```

Arguments

`value`	Fuzzy number to approximate.
`method`	The selected approximation method.
`piecewise`	If `piecewise=TRUE` is set, then the methods "Naive","NearestEuclidean" (from the FuzzyNumbers package) produce piecewise linear fuzzy number as the output, otherwise they result in trapezoidal fuzzy number.
`...`	Additional parameters passed to other functions (like approximation method from the FuzzyNumbers package).

Details

The function approximates the fuzzy number given by `value` with the method selected by `method`. The following approximations are possible: `ExpectedValueCore`–preserving the expected value and the core of `value`, `TriangSuppPreserving`–constructs the triangular fuzzy number based on minimization of `DpqDistance`, preserving the support of `value`, `AmbiguityValuePreserving`–minimizing the Euclidean distance, while preserving the ambiguity and value, and the approximation methods from the FuzzyNumbers package (namely: `Naive`, `NearestEuclidean`, `ExpectedIntervalPreserving`, `Sup`

The input value should be given by a fuzzy number described by classes from `FuzzyNumbers` package.

Value

The output is a fuzzy number (triangular or trapezoidal one) described by classes from `FuzzyNumbers` package.

Examples

```
library(FuzzyNumbers)

# prepare complex fuzzy number

A <- FuzzyNumber(-5, 3, 6, 20, left=function(x)
pbeta(x,0.4,3),
right=function(x) 1-x^(1/4),
lower=function(alpha) qbeta(alpha,0.4,3),
upper=function(alpha) (1-alpha)^4)

# find approximation

FuzzyApproximation (A)
```

`PlotZEP` *Plot input and output for the Zadeh's principle*

Description

`PlotZEP` applies the selected function to a fuzzy number using the Zadeh's principle, and plots the input and output.

Usage

```
PlotZEP(
  value,
  FUN,
  knots = 10,
  grid = TRUE,
```

```
  alternate = FALSE,
  approximation = FALSE,
  xRange = NA,
  yRange = NA,
  method = "NearestEuclidean",
  ...
)
```

Arguments

value	Input fuzzy number.
FUN	Function used for the input fuzzy number with the help of the Zadeh's principle.
knots	Number of the alpha-cuts used during calculation of the output.
grid	If TRUE, then additional grid is plotted.
alternate	If TRUE, the second type of the layout of figures is used.
approximation	If TRUE, the approximated output is calculated.
xRange	If NA, the support of the input fuzzy number is used for to plot x-axis (first and second plot), otherwise the given vector is applied.
yRange	If NA, the support of the output fuzzy number (or its approximation) is used for to plot x-axis (third and fourth plot), otherwise the given vector is applied.
method	The selected approximation method.
...	Additional parameters passed to other functions.

Details

The function takes the input fuzzy number value (which should be described by one of the classes from FuzzyNumbers package) and applies the function FUN using the Zadeh's principle. The output is given by a fuzzy number or its approximation (when approximation is set to TRUE and the respective method is selected). To properly find the output, value of FUN is calculated for many alpha-cuts of value. The number of these alpha-cuts is equal to knots (plus 2 for the support and the core). The input and output fuzzy numbers are plotted together with the applied function. If the approximation is used, then also the approximated fuzzy number is shown (green line).

The input fuzzy number value should be given by fuzzy number described by classes from FuzzyNumbers package.

Value

Three (or four) figures are plotted: the input fuzzy number, the respective output (for the Zadeh's principle and the applied function), and the function. The output fuzzy number can be approximated with the selected method and also plotted.

Examples

```
library(FuzzyNumbers)

# prepare complex fuzzy number
```

```
A <- FuzzyNumber(-5, 3, 6, 20, left=function(x)
pbeta(x,0.4,3),
right=function(x) 1-x^(1/4),
lower=function(alpha) qbeta(alpha,0.4,3),
upper=function(alpha) (1-alpha)^4)

# plot the figures

PlotZEP(A,FUN=function(x)x^3+2*x^2-1)

# find and plot the approximated output via the Zadeh's principle

PlotZEP(A,FUN=function(x)x^3+2*x^2-1,approximation=TRUE)
```

Index

∗ **datasets**
approximationMehodsInside, 7

AnimateListZEP, 2
AnimateZEP, 4
ApplyZEP, 5
approximationMehodsInside, 7

DpqDistance, 7

FuzzyApproximation, 8

PlotZEP, 9