Package ‘LearningRlab’

October 30, 2018

Version 1.3

Date 2018-10-23

Title Statistical Learning Functions

Author Dennis Monheimius [aut],
Eduardo Benito [aut, cre],
Juan Jose Cuadrado [aut],
Universidad de Alcala de Henares [aut]

Maintainer Eduardo Benito <eduardo.benito@edu.uah.es>

Depends magick, crayon

Suggests knitr

Description Aids in learning statistical functions incorporating the result of calculus done with each function and how they are obtained, that is, which equations and variables are used. Also for all these equations and their related variables detailed explanations and interactive exercises are also included. All these characteristics allow to the package user to improve the learning of statistics basics by means of their use.

License Unlimited

VignetteBuilder knitr

NeedsCompilation yes

Type Package

Repository CRAN

Date/Publication 2018-10-30 19:10:05 UTC

R topics documented:

averageDeviation_ *Average Absolute Deviation Function*

Description

This function calculates the average absolute deviation of a numbers vector.

Usage

```
averageDeviation_(x)
```

Arguments

`x`	Should be a numbers vector

Details

To calculate the average deviation, the user should give a numbers vector. The result is the sum of the differences in absolute value between each vector element and the mean, divided by the number of elemets. The average absolute deviation formule is the following:

Value

Numeric, the average absolute deviation of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <`dennis.monhemimius@edu.uah.es`>
Eduardo Benito, <`eduardo.benito@edu.uah.es`>
Juan Jose Cuadrado, <`jjcg@uah.es`>
Universidad de Alcala de Henares

Examples

```
#data creation
data <- c(1:20)
result = averageDeviation_(data)
```

`drawVector` *Draw Vector Function*

Description

This function prints all the elements of a vector

Usage

```
drawVector(buffer)
```

Arguments

`buffer`	A vector of elements

Value

There isn't return value, prints on screen

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5 or c(true,false,false) creates a vector with the booleans: true, false, true

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
  #data creation
  data <- c(1:12)
  drawVector(data)
  }
```

---

explain.absolute_acum_frecuency

Absolute Accumulated Frecuency Calculus Explained

Description

Step by step demonstration of the absolute accumulated frecuency calculus

Usage

```
explain.absolute_acum_frecuency(v,x)
```

Arguments

`v`	Should be a vector
`x`	Should be a number

Details

To calculate the absolute accumulated frecuency, the user should give a vector and a number. We can saw the absolute accumulated frecuency formule in the frecuency_acum_absolute help document.

Value

A demonstration of the calculus process

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)
    value = 2
    #function execution
    explain.absolute_acum_frecuency(data, value)
}
```

---

explain.absolute_frecuency
Absolute Frecuency Calculus Explained

Description

Step by step demonstration of the absolute frecuency calculus

Usage

```
explain.absolute_frecuency(v,x)
```

Arguments

v	Should be a vector
x	Should be a number

Details

To calculate the absolute frecuency, the user should give a vector and a number. We can saw the absolute frecuency formule in the frecuency_abs help document.

Value

A demonstration of the calculus process

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)
    value = 2
    #function execution
    explain.absolute_frecuency(data, value)
}
```

---

explain.averageDeviation

Average Absolute Deviation Function Explained

Description

Step by step demonstration of the average absolute deviation calculus.

Usage

```
explain.averageDeviation(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the average absolute deviation, the user should give a numbers vector. The result is the explained process to calculate the average absolute deviation, with the data of the dataset provided like argument. We can saw the average absolute deviation formule in the averageDeviation_ help document.

Value

Numeric, the average absolute deviation of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
#data creation
data <- c(7,2,5,7,1,4,12)

explain.averageDeviation(data)
```

---

explain.geometricMean *Geometric Mean Function Explained*

Description

Step by step demonstration of the geometric mean calculus.

Usage

```
explain.geometricMean(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the geometric mean of a dataset, the user should give a vector. The result is the explained process to calculate the geometric mean, with the data of the dataset provided like argument. We can saw the geometric mean formule in the geometricMean_ help document.

Value

Numeric result and the process of this calculus explained.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(5,21,12,7,3,9,1)

    explain.geometricMean(data)
}
```

explain.mean *Mean Function Explained*

Description

Step by step demonstration of the arithmetic mean calculus.

Usage

```
explain.mean(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the arithmetic mean of a dataset, the user should give a vector. The result is the explained process to calculate the arithmetic mean, with the data of the dataset provided like argument. We can saw the arithmetic mean formule in the mean_ help document.

Value

Numeric result and the process of this calculus explained.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)

    explain.mean(data)
}
```

---

explain.median *Median Function Explained*

Description

Step by step demonstration of the median calculus.

Usage

```
explain.median(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the median, the user should give a numbers vector. The result is the explained process to calculate the median, with the data of the dataset provided like argument. We can saw the median formule in the median_ help document.

Value

Numeric result and the process of this calculus explained.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)

    explain.median(data)
}
```

explain.mode *Mode Function Explained*

Description

Step by step demonstration of the mode calculus.

Usage

```
explain.mode(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the mode, the user should give a numbers vector. The result is the explained process to calculate the mode, with the data of the dataset provided like argument. We can saw the mode formule in the mode_ help document.

Value

Numeric result and the process of this calculus explained.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
  #data creation
  data <- c(1,1,2,5,2,3,1,4,1)

  explain.mode(data)

}
```

---

explain.percentile *Percentiles Calculus Explained*

Description

Step by step demonstration of the percentiles calculus

Usage

```
explain.percentile(x)
```

Arguments

x	Should be a vector

Details

To calculate the percentiles, the user should give a vector. We can saw the percentile formule in the percentile_ help document.

Value

A demonstration of the calculus process

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)

    explain.percentile(data)
}
```

`explain.quartile` *Quartiles Calculus Explained*

Description

Step by step demonstration of the quartiles calculus

Usage

```
explain.quartile(x)
```

Arguments

`x`	Should be a vector

Details

To calculate the quartiles, the user should give a vector. We can saw the quartile formule in the quartile_ help document.

Value

A demonstration of the calculus process

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <`dennis.monhemimius@edu.uah.es`>
Eduardo Benito, <`eduardo.benito@edu.uah.es`>
Juan Jose Cuadrado, <`jjcg@uah.es`>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)

    explain.quartile(data)
}
```

---

explain.relative_acum_frecuency *Relative Accumulated Frecuency Calculus Explained*

Description

Step by step demonstration of the relative accumulated frecuency calculus

Usage

```
explain.relative_acum_frecuency(v,x)
```

Arguments

v	Should be a vector
x	Should be a numebr of the vector

Details

To calculate the relative accumulated frecuency, the user should give a vector and a number. We can saw the relative accumulated frecuency formule in the frecuency_acum_relative help document.

Value

A demonstration of the calculus process

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)
    value = 2
    #function execution
    explain.relative_acum_frecuency(data, value)
}
```

---

explain.relative_frecuency

Relative Frecuency Calculus Explained

Description

Step by step demonstration of the relative frecuency calculus

Usage

```
explain.relative_frecuency(v,x)
```

Arguments

v	Should be a vector
x	Should be a number

Details

To calculate the relative frecuency, the user should give a vector and a number. We can saw the relative frecuency formule in the frecuency_relative help document.

Value

A demonstration of the calculus process

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data <- c(1,2,2,5,10,4,2)
    value = 2
    #function execution
    explain.relative_frecuency(data, value)
}
```

---

explain.standardDeviation

Standard Deviation Function Explained

Description

Step by step demonstration of the standard deviation calculus.

Usage

```
explain.standardDeviation(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the standard deviation, the user should give a numbers vector. The result is the explained process to calculate the standard deviation, with the data of the dataset provided like argument. We can saw the standard deviation formule in the standardDeviation_ help document.

Value

Numeric result and the process of this calculus explained.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
#data creation
data <- c(1,5,3,7,10,4,2)

explain.standardDeviation(data)
```

explain.variance *Variance Function Explained*

Description

Step by step demonstration of the variance calculus.

Usage

```
explain.variance(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the variance, the user should give a numbers vector. The result is the explained process to calculate the variance, with the data of the dataset provided like argument. We can saw the variance formule in the variance_ help document.

Value

Numeric result and the process of this calculus explained.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
#data creation
data <- c(10,4,5,7,3,4,1)

explain.averageDeviation(data)
```

`frecuency_abs` *Absolute Frecuency Calculus*

Description

This function calculate the number of times that a specific number appears in the data set.

Usage

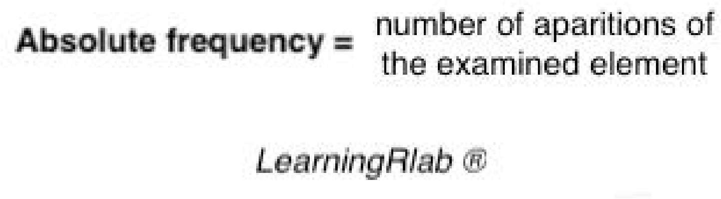
```
frecuency_abs(v,x)
```

Arguments

`v`	Should be a vector
`x`	Should be a number

Details

The absolute frecuency formula is the following:



Absolute frequency = number of aparitions of the examined element

LearningRlab ®

Value

An integer that represents the number of times that the value appears in the vector

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
  #data creation
  data = c(1,4,3,3,2,5,7,12,1,2,3,12)
  value = 12
  #function execution
  frecuency_abs(data, value)
}
```

---

frecuency_absolute_acum
Accumulated Absolute Frecuency Calculus

Description

This function calculate the number of times that a specific number appears in the data set. The value depends on the elements that are lower than itself

Usage

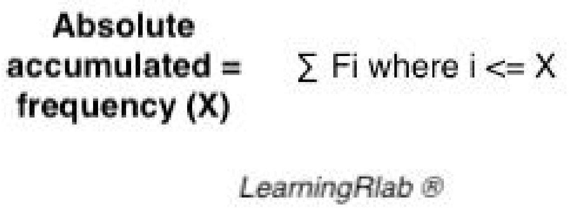
```
frecuency_absolute_acum(v,x)
```

Arguments

v	Should be a vector
x	Should be a number

Details

The accumulated absolute frecuency formula is the following:



Absolute accumulated = frequency (X) ∑ Fi where i <= X

LearningRlab ®

Value

A double that represents the number of times that the value appears in the vector regarding the total of elements

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data = c(1,4,3,3,2,5,7,12,1,2,3,12)
    value = 12
    #function execution
    frecuency_absolute_acum(data, value)

    }
```

`frecuency_relative` *Relative Frecuency Calculus*

Description

This function calculate the number of times that a specific number appears in the data set divided by the total length of the vector.

Usage

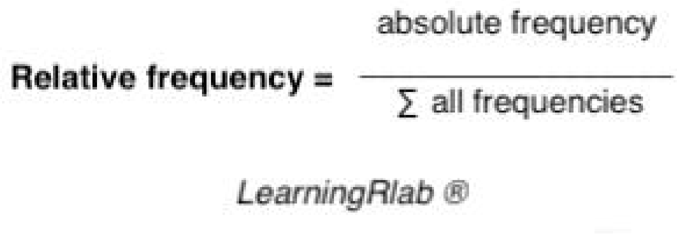
```
frecuency_relative(v,x)
```

Arguments

v	Should be a vector
x	Should be a number

Details

The relative frecuency formula is the following:



Value

A double that represents the number of times that the value appears in the vector regarding the total of elements

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
#data creation
data <- c(1,4,3,3,2,5,7,12,1,2,3,12)
value = 12
frecuency_relative(data, value)
}
```

---

frecuency_relative_acum

Accumulated Relative Frecuency Calculus

Description

This function calculate the number of times that a specific number appears in the data set divided by the total length of the vector. The value depends on the elements that are lower than itself

Usage

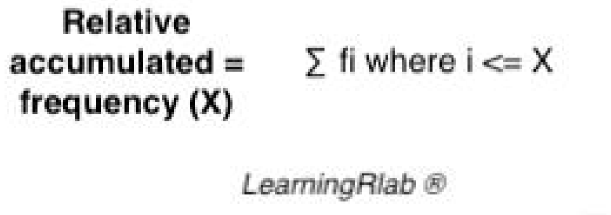
```
frecuency_relative_acum(v,x)
```

Arguments

`v`	Should be a vector
`x`	Should be a number

Details

The accumulated relative frecuency formula is the following:



Relative accumulated = frequency (X) ∑ fi where i <= X

LearningRlab ®

Value

A double that represents the number of times that the value appears in the vector regarding the total of elements

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <`dennis.monhemimius@edu.uah.es`>
Eduardo Benito, <`eduardo.benito@edu.uah.es`>
Juan Jose Cuadrado, <`jjcg@uah.es`>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data = c(1,4,3,3,2,5,7,12,1,2,3,12)
    value = 12
    #function execution
    frecuency_relative_acum(data, value)

    }
```

geometricMean_ *Geometric Mean Function*

Description

This function calculates the geometric mean of a numbers vector.

Usage

```
geometricMean_(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the geometric mean of a dataset, the user should give a numbers vector. The result is the product of all vector elements raise to 1 divided by the number of elements. The arithmetic mean formule is the following:

Value

A numeric, the geometric mean of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
#data creation
data = c(1:20)

geometricMean_(data)
```

getUserAction *Get User Action Funcion*

Description

This function get the buffer introduced by the user. Typically a numerical vector.

Usage

```
getUserAction()
```

Arguments

void, not argument

Value

A vector

Note

The process is interactive with the user

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
  ## Not run:
  vector <- getUserAction()

## End(Not run)
  }
```

initImages *Init Images Function*

Description

This function is used to display an image.

Usage

```
initImages(path)
```

Arguments

path	An url of an image

Value

There isn't return value

Note

The path shold be toward an image

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{

  ## Not run:
    path = "https://i.imgur.com/8237YhzJ.png"
  initImages(path)


## End(Not run)
  }
```

`interactive.absolute_acum_frecuency`

User Interactive Absolute Accumulated Frecuency Calculus

Description

Interactive function for absolute accumulated frecuency calculus.

Usage

```
interactive.absolute_acum_frecuency()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the formula.

Value

An interactive process to calculate the absolute accumulated frecuency

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.absolute_acum_frecuency()

## End(Not run)
```

interactive.absolute_frecuency

User Interactive Absolute Frecuency Calculus

Description

Interactive function for absolute frecuency calculus.

Usage

```
interactive.absolute_frecuency()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the formula.

Value

An interactive process to calculate the absolute frecuency

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.absolute_frecuency()

## End(Not run)
```

`interactive.averageDeviation` *User Interactive Average Absolute Deviation Calculus*

Description

Interactive function for average absolute deviation calculus.

Usage

```
interactive.averageDeviation()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the average absolute deviation formule, apart from the averageDeviation_ help document.

Value

An interactive process to calculate the average absolute deviation

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>

References

https://en.wikipedia.org/wiki/Average_absolute_deviation

Examples

```
   ## Not run:
  interactive.averageDeviation()

## End(Not run)
```

interactive.geometricMean

User Interactive Geometric Mean Calculus

Description

Interactive function for geometric mean calculus.

Usage

```
interactive.geometricMean()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the geometric mean formule, apart from the geometricMean_ help document.

Value

An interactive process to calculate the geometric mean.

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.geometricMean()

## End(Not run)
```

`interactive.mean` *User Interactive Mean Calculus*

Description

Interactive function for arithmetic mean calculus.

Usage

```
interactive.mean()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the arithmetic mean formule, apart from the mean_ help document.

Value

An interactive process to calculate the arithmetic mean.

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{

  ## Not run:
  interactive.mean()

## End(Not run)
}
```

`interactive.median` *User Interactive Median Calculus*

Description

Interactive function for median calculus.

Usage

```
interactive.median()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the median formule, apart from the median_ help document.

Value

An interactive process to calculate the median

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.median()

## End(Not run)
```

`interactive.mode` *User Interactive Mode Calculus*

Description

Interactive function for mode calculus.

Usage

```
interactive.mode()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset.

Value

An interactive process to calculate the mode.

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.mode()

## End(Not run)
```

interactive.percentile *User Interactive Percentile Calculus*

Description

Interactive function for percentiles calculus.

Usage

```
interactive.percentile()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the formula.

Value

An interactive process to calculate the percentiles

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.percentile()

## End(Not run)
```

`interactive.quartile` *User Interactive Quartiles Calculus*

Description

Interactive function for quartiles calculus.

Usage

```
interactive.quartile()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the formula.

Value

An interactive process to calculate the quartiles

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.quartile()

## End(Not run)
```

interactive.relative_acum_frecuency *User Interactive Relative Accumulated Frecuency Calculus*

Description

Interactive function for relative accumulated frecuency calculus.

Usage

```
interactive.relative_acum_frecuency()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the formula.

Value

An interactive process to calculate the relative accumulated frecuency

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.relative_acum_frecuency()

## End(Not run)
```

interactive.relative_frecuency

User Interactive Relative Frecuency Calculus

Description

Interactive function for relative frecuency calculus.

Usage

```
interactive.relative_frecuency()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the formula.

Value

An interactive process to calculate the relative frecuency

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
  ## Not run:
  interactive.relative_frecuency()

## End(Not run)
```

`interactive.standardDeviation` *User Interactive Standard Deviation Calculus*

Description

Interactive function for standard deviation calculus.

Usage

```
interactive.standardDeviation()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the standard deviation formule, apart from the standardDeviation_ help document.

Value

An interactive process to calculate the standard deviation

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>

References

https://en.wikipedia.org/wiki/Standard_deviation

Examples

```
   ## Not run:
  interactive.standardDeviation()

## End(Not run)
```

`interactive.variance` *User Interactive Variance Calculus*

Description

Interactive function for variance calculus.

Usage

```
interactive.variance()
```

Arguments

void, not argument

Details

The user provides the dataset when the function needs it. After that, the function will ask what is the correct result for this dataset. The function itself will provide the variance formule, apart from the variance_ help document.

Value

An interactive process to calculate the average absolute deviation

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```

  ## Not run:
  interactive.variance()

## End(Not run)
```

`LearningRlab` *Statistical Learning Functions*

Description

Package used to teach basic statistics to students.

Details

This package pretends to serve the user as a method of learning basic statistical functions at secondary and baccalaureate courses. The content of the package incorporate a serie of statistical functions like the calculus of the arithmetic mean or the calculus of the frequencies. There is no only calculus functions, further more, there are incorporated interactive and explicative functions to help and guide the user in the learning process.

Package:	LearningRlab
Type:	Package
Version:	1.2
Date:	2018-09-19
License:	Unlimited

Author(s)

Dennis Monheimius, `<dennis.monhemimius@edu.uah.es>`
Eduardo Benito, `<eduardo.benito@edu.uah.es>`
Juan Jose Cuadrado, `<jjcg@uah.es>`
Universidad de Alcala de Henares
Maintainer: Eduardo Benito `<eduardo.benito@edu.uah.com>`

`meanC` *Mean Function Developed in C*

Description

This function calculates the arithmetic mean of a numbers vector.

Usage

```
meanC(x)
```

Arguments

`x`	Should be a numbers vector

Details

To calculate the arithmetic mean of a dataset, the user should give a numbers vector. The result is the addition of all vector elements divided by the number of elements. The arithmetic mean formule is the following:

MEAN FORMULA

$$\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n} = \frac{X1 + X2 + \ldots + Xn}{n}$$

@LearningRlab

Value

A numeric, the arithmetic mean of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
#data creation
vector = c(1:10)
meanC(vector)
```

mean_ *Mean Calculus Function*

Description

This function calculates the arithmetic mean of a numbers vector.

Usage

```
mean_(x)
```

Arguments

x Should be a numbers vector

Details

To calculate the arithmetic mean of a dataset, the user should give a numbers vector. The result is the addition of all vector elements divided by the number of elements. The arithmetic mean formule is the following:

MEAN FORMULA

$$\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n} = \frac{X1 + X2 + \ldots + Xn}{n}$$

®LearningRlab

Value

A numeric, the arithmetic mean of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
vector <- c(2,4,6,8,10,12,14,16,18)
result = mean_(vector)
result
```

---

median_ *Median Calculus Function*

Description

This function calculates the median of a numbers vector.

Usage

```
median_(x)
```

Arguments

`x`	Should be a numbers vector

Details

To calculate the median, the user should give a numbers vector. The result is the value separating the higher half from the lower half of the dataset, it may be thought of as the middle value. The median formule is the following:

$$\text{Median} = \frac{1}{2}(n + 1)\text{th value}$$

® LearningRlab

Value

A numeric, the median of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <`dennis.monhemimius@edu.uah.es`>
Eduardo Benito, <`eduardo.benito@edu.uah.es`>
Juan Jose Cuadrado, <`jjcg@uah.es`>
Universidad de Alcala de Henares

Examples

```
{

result = median_(c(1,3,2,5,12,4,4,2,9))
result
}
```

mode_ *Mode Calculus Function*

Description

This function calculates the mode of a numbers vector.

Usage

```
mode_(x)
```

Arguments

x	Should be a numbers vector

Details

To calculate the mode of a dataset, the user should give a numbers vector. The result is the numeric value that appears most often. In other words, it's the value that is most likely to be sampled. The mode formule is the following:

Value

Numeric, the mode of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{

    #data creation
    data = c(1,2,2,3,4)

    mode_(data)

}
```

percentile_ *Percentile Calculus Function*

Description

This function calculate the percentiles of a vector of numbers

Usage

```
percentile_(x)
```

Arguments

x	Should be a vector

Details

To calculate the percentiles, the user should give a vector. This function divide the dataset in 100 parts as equal as possible. The formula is the following:

Value

A vector sorted with the elements divided by 100 parts

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data = c(1,4,3,3,2,5,7,12,1,2,3,12)

    percentile_(data)

}
```

quartile_ *Quartiles Calculus*

Description

Calculates the 3 Quartiles of a vector of data

Usage

```
quartile_(x)
```

Arguments

x	Should be a vector

Details

To calculate the quartiles, the user should give a vector. This function divide the dataset in 4 parts as equal as possible. The formula is the following:

Value

A vector sorted with the elements divided by 4 parts

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
{
    #data creation
    data = c(1:20)

    quartile_(data)

    }
```

`standardDeviation_` *Standard Deviation Calculus Function*

Description

This function calculates the standard deviation of a numbers vector.

Usage

```
standardDeviation_(x)
```

Arguments

`x`	Should be a numbers vector

Details

To calculate the standard deviation, the user should give a numbers vector. The result is the square root of the sum of the differences between each vector element and the mean squared divided by the number of elemets. The standard deviation formule is the following:

Value

Numeric, the standard deviation of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```
#data creation
data = c(1,4,3,3,2,5,7,12,1,2,3,12)

standardDeviation_(data)
```

`variance_` *Variance Calculus Function*

Description

This function calculates the variance of a numbers vector.

Usage

```
variance_(x)
```

Arguments

`x`	Should be a numbers vector

Details

To calculate the variance, the user should give a numbers vector. The result is the expectation of the squared deviation of all numbers vector from its mean. The variance formule is the following:

Value

Numeric, the variance of the numbers vector.

Note

A vector is created by c(), like c(1,2,3,4,5) creates a vector with the numbers: 1,2,3,4,5

Author(s)

Dennis Monheimius, <dennis.monhemimius@edu.uah.es>
Eduardo Benito, <eduardo.benito@edu.uah.es>
Juan Jose Cuadrado, <jjcg@uah.es>
Universidad de Alcala de Henares

Examples

```

#data creation
data = c(1,4,3,3,2,5,7,12,1,2,3,12)

variance_(data)
```

Index