Package 'VecStatGraphs3D'

October 7, 2014

Type Package

Title Vector analysis using graphical and analytical methods in 3D

Version 1.6

Date 2014-10-03

Author Angel Felicisimo, Juan Carlos Ruiz Cuetos, Maria Eugenia Polo Garcia, Aurora Cuartero,Pablo Garcia Rodriguez

Maintainer Pablo Garcia Rodriguez <pablogr@unex.es>

Depends R (>= 2.10.1), rgl, misc3d, tcltk, MASS

Description This package performs a 3D statistical analysis, both numerical and graphical, of a set of vectors. Since a vector has three components (a module and two angles) vectorial analysis is performed in two stages: modules are analyzed by means of linear statistics and orientations are analyzed by spherical statistics. Tests and spherical statistic parameters are accompanied by a graphs as: density maps, distribution modules and angles. The tests, spherical statistic parameters and graphs allow us detecting another distribution properties (i.e. anisotropy) and outliers.

License GPL-3

URL http://gim.unex.es/VecStatGraphs3D/

Repository CRAN

NeedsCompilation no

Date/Publication 2014-10-07 09:38:46

R topics documented:

`VecStatGraphs3D-package`

Vector analysis using graphical and analytical methods in 3D

Description

This package performs a 3D statistical analysis, both numerical and graphic, of a set of vectors. Since a vector has three components (a module and two angles) vector analysis is performed in two stages: modules are analyzed by means of linear statistics and orientations are analyzed by spherical statistics. Tests and spherical statistic parameters are accompanied by a full range of graphing: vector maps, density maps,distribution modules and angles. The tests, spherical statistic parameters and graphs allow us detecting another distribution properties (I.e. anisotropy) and outliers.

Details

Package:	VecStatGraphs3D
Type:	Package
Version:	1.2
Date:	2012-08-09
License:	GPL-3
LazyLoad:	yes

Author(s)

Juan Carlos Ruiz Cuetos, Maria Eugenia Polo Garcia, Pablo Garcia Rodriguez.

Maintainer: Pablo Garcia Rodriguez <pablogr@unex.es>

References

Fisher N.I. , Lewis T. , Embleton, B.J.J. (1987) Statistical analysis of spherical data. Cambridge. Cambridge University Press.

Website http://gim.unex.es/VecStatGraphs3D/

See Also

AllAngleStatistics, AllModuleStatistics3D, DrawModuleAndAngleDistribution3D, DrawDensity3D, RayleighTest3D.

`AllAngleStatistics` *Calculation of All Statistics of the Angle.*

Description

This function calculates several statistics for unit and non-unit vectors from a set of input coordinates.

Usage

```
AllAngleStatistics(incr, ndig=4)
```

Arguments

`incr`	Matrix containing the values of the coordinates
`ndig`	Integer value that indicates the decimal places. The default value is 4

Details

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

The statistics calculated are: `Spherical Standar Error, Mean Module, Concentration Parameter, Mean Direction.`

Value

The value All Angle Statistics of all input X, Y and Z coordinates.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Felicisimo, A.M., <amfeli@unex.es>, Cuartero, A., <acuartero@unex.es>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Fisher N.I. , Lewis T. , Embleton, B.J.J. (1987) Statistical analysis of spherical data. Cambridge. Cambridge University Press.

Website http://gim.unex.es/VecStatGraphs3D/

See Also

MeanModule3D,\ MeanDirection3D, SphericalStandardError3D, ConcentrationParameter3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
coordinates<-dat[,4:6]
AllAngleStatistics(coordinates, ndig=4)
```

AllModuleStatistics3D *Calculation of All Statistics of the Modules.*

Description

This function calculates several statistics from a set of input modules.

Usage

```
AllModuleStatistics3D(modules, ndig=4)
```

Arguments

modules	Vector containing the values of the modules
ndig	Integer value that indicates the decimal places. The default value is 4

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

The statistics calculated are: Number Of Elements, Max Value, Min Value, Range, Module Sum, Mean Arithmetic, S

Value

The value All Module Statistics of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Felicisimo, A.M., <amfeli@unex.es>, Cuartero, A., <acuartero@unex.es>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
AllModuleStatistics3D(modules, ndig=4)
```

ArithmeticMean3D *Calculation of Mean Arithmetic modules.*

Description

This function calculates the arithmetic mean of a set of modules

Usage

```
ArithmeticMean3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The arithmetic mean value of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, Range3D, ModuleSum3D, StandardError3D, ModuleStandardDeviation
ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
ArithmeticMean3D(modules)
```

ConcentrationParameter3D

Calculation of concentration parameter of the angles.

Description

This function calculates the concentration parameter a set of input coordinates.

Usage

```
ConcentrationParameter3D(coord)
```

Arguments

coord	Matrix containing the values of the coordinates

Details

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

The concentration parameter measures the departure of the distribution from the uniform distribution. If this parameter tends to 0, the distribution converges to the uniform distribution, if it tends to infinity, the distribution is concentrated around the mean angle.

Value

The concentration parameter all input coordinates.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Felicisimo, A.M., <amfeli@unex.es>, Cuartero, A., <acuartero@unex.es>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Fisher N.I. , Lewis T. , Embleton, B.J.J. (1987) Statistical analysis of spherical data. Cambridge. Cambridge University Press.

Website http://gim.unex.es/VecStatGraphs3D/

See Also

MeanModule3D, MeanDirection3D, SphericalStandardError3D, AllAngleStatistics.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
coordinates<-dat[,4:6]
ConcentrationParameter3D(coordinates)
```

DrawDensity3D *Graphic representation of a point Density Map.*

Description

The density map was built as follows: a) vectors are moved to a common origin without changing its angle and module, b) end position perform a point cloud of varying density that depends on the accumulation of vectors with similar properties, c) end positions are used for to calculate the density map as a surface with values depending on the end-position accumulation per area unit. Areas where there is a greater number of points (end position of the vectors) will have a deeper color (red), while the areas a lesser density will have a less intense color (white).

Usage

```
DrawDensity3D(vectors, Div = 40, Layers = 3, DrawAxes = FALSE)
```

Arguments

vectors	Matrix containing the values of the coordinates
Div	Integer value that indicates the number of divisions that will have the density map. The default value is 40
Layers	Integer value that indicates the number of layers that will have the density map. The default value is 3
DrawAxes	Logical value, if DrawAxes=TRUE draw axes, if DrawAxes=FALSE draw not axes. The default value is FALSE

Details

Kernel descriptors are used to create the density map. To perform these calculations the MASS package is required.

The parameter Div is very important because a very large value will cause the creation of the slow density map, and a very small value would create a ineffective density map.

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

Typical usages are

```
    DrawDensity3D(vectors, ...)
DrawDensity3D(vectors, Div = 60, ...)
DrawDensity3D(vectors, Layers = 4, DrawAxes = TRUE, ...)
    ......
```

Value

This function returns no value, creates a 3D Graph that represents a density map of the input values.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Felicisimo, A.M., <amfeli@unex.es>, Cuartero, A., <acuartero@unex.es>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs2D/

See Also

DrawModuleAndAngleDistribution3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
coordinates<-dat[,4:6]
DrawDensity3D(coordinates, Layers=3, Div=50)
```

DrawModuleAndAngleDistribution3D

Graphic representation of Module and Angle Distribution.

Description

This function creates a 3D graph, that allows us to check the angular and linear magnitudes simultaneously.

Usage

```
DrawModuleAndAngleDistribution3D(dat, Long=FALSE, hW=0.5, hL=0.3, plane, BarSlider=FALSE)
```

Arguments

dat	Matrix containing the values of the LoadData3D
Long	Logical value, if Long = TRUE we can select an area of points for to obtain its length.The default value is Long = FALSE.
hW	Integer value, representing the width of the head of the arrows. The default value is 0.5
hL	Integer value, representing the lenght of the head of the arrows. The default value is 0.3
plane	Possible values are XY, XZ and YZ represents the plane where they will draw the orientacion the heads of the arrows.
BarSlider	Logical value, if BarSlider = TRUE we may change the size of the heads of the arrows through a slider.The default value is BarSlider = FALSE.

Details

The 3D graph draws an arrow for each of the samples. The lines will have origin in (0,0) and end at coordinates (X, Y, Z).

Typical usages are

```
   DrawModuleAndAngleDistribution3D(dat, Long = FALSE, plane = "XY", ...)
   DrawModuleAndAngleDistribution3D(dat, hW = 0.6, hL = 0.4, ...)
DrawModuleAndAngleDistribution3D(dat, Long = FALSE, plane = "XY", BarSlider = TRUE, ...)
```

Value

This function returns no value, creates a 3D graph that represents module and angle distribution

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

DrawDensity3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
DrawModuleAndAngleDistribution3D(dat, plane="XY");
```

KurtosisModuleCoefficient3D

Calculation of the kurtosis coefficient of the modules in 3D

Description

This function calculates the kurtosis coefficient of the modules in 3D, which characterizes the shape of the distribution from a set of input modules.

Usage

```
KurtosisModuleCoefficient3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to obtain a set of modules from X, Y and Z coordinates of the vectors is to use the LoadData3D function.

The kurtosis coefficient is a measure of whether the data are flat or peaked relative to the normal distribution.

Value

The value of the kurtosis coefficient of all input modules

Author(s)

Felicisimo, A.M., <amfeli@unex.es>, Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Cuartero, A., <acuartero@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance
ModulePopulationStandardDeviation3D, SkewnessModuleCoefficient3D, AllModuleStatistics3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
KurtosisModuleCoefficient3D(modules)
```

`LoadData3D` *Data loading and conversion of cartesian coordinates, incremental data, and polar coordinates.*

Description

This function reads data from a txt file containing data of coordinates. Data can be read as cartesian coordinates, incremental data or polar coordinates. This functions returns a matrix with all possible data conversions.

Usage

```
LoadData3D(FileName, Type = 2)
```

Arguments

`FileName`	File name to load data
`Type`	Integer value that indicates the type of data contained in the file. Type 1 = Cartesian, Type 2= Incremental, Type 3 = Polar

Details

Input files must be in a certain format for the data type. By default the type is incremental data (Type = 2).

The cartesian coordinate type contains six columns, defining the coordinates of a vector. The first three columns represent the X, Y and Z coordinates of the intial data and the last three columns represents the coordinates of the final data, and each row corresponds to one of the data. If necessary, the inverse order can be used to adapt the direction to our data.

The incremental data type contains three columns, the first column is the increase of X coordinate, the second column is the increase of Y coordinate and the third column is the increase of Z coordinates, and each row corresponds to one of the data. The X coordinate, Y coordinate and Z coordinate are obtained from the difference of the final node with respect to the initial node.

The polar type contains two or three columns. If it contains two columns, the first column represents the colatitude and the second column represents the longitude, in this case the module by default is equal to 1. If it contains three columns, the first column represents the module, the second column represents the colatitude and the third column represents the longitude. Each row corresponds to one of the data.

Typical usages are

```
LoadData3D(FileName, ...)
LoadData3D(FileName, Type=2)
......
```

Value

The function returns a nx13 matrix size , where n is the number of data. The column 1 represents the module, the column 2 represents the colatitude, the column 3 represents the longitude, the column 4 represents the increase of X coordinate, the column 5 represents the increase of Y coordinate, the column 6 represents the increase of Z coordinate, the column 7 represents the type of input data, the column 8, 9 and 10 represent the X, Y and Z coordinates of the initial data, the column 11, 12 and 13 represent the X, Y and Z coordinates of the final data.

The columns 8, 9, 10, 11, 12 and 13 will only have values if the input data are of Type 1 (cartesian coordinates).

Author(s)

Felicisimo, A.M., <amfeli@unex.es>, Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Cuartero, A., <acuartero@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

DrawDensity3D, DrawModuleAndAngleDistribution3D, AllAngleStatistics, AllModuleStatistics3D, RayleighTest3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
LoadData3D(FileName, Type=1)
```

MaxValue3D *Calculation of the Maximum Value of the modules.*

Description

This function calculates the maximum value of a set of modules.

Usage

```
MaxValue3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The max value of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MinValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviat

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
MaxValue3D(modules)
```

MeanDirection3D *Calculation of the Mean Direction of the angles.*

Description

This function calculates the Mean Direction of a set of input coordinates.

Usage

```
MeanDirection3D(coord)
```

Arguments

coord	Matrix containing the values of the coordinates

Details

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

Value

The mean direction value of all input coordinates. The value is expressed in sexagesimal.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Fisher N.I. , Lewis T. , Embleton, B.J.J. (1987) Statistical analysis of spherical data. Cambridge. Cambridge University Press.

Website http://gim.unex.es/VecStatGraphs3D/

See Also

AllAngleStatistics, MeanModule3D, SphericalStandardError3D, ConcentrationParameter3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
coordinates<-dat[,4:6]
MeanDirection3D(coordinates)
```

MeanModule3D *Calculation of the Mean Module of the angles.*

Description

This function calculates the mean module from a set of input coordinates.

Usage

```
MeanModule3D(coord)
```

Arguments

coord	Matrix containing the values of the coordinates

Details

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

Value

The mean module value of all input coordinates.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Fisher N.I. , Lewis T. , Embleton, B.J.J. (1987) Statistical analysis of spherical data. Cambridge. Cambridge University Press.

Website http://gim.unex.es/VecStatGraphs3D/

See Also

AllAngleStatistics, MeanDirection3D, SphericalStandardError3D, ConcentrationParameter3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
coordinates<-dat[,4:6]
MeanModule3D(coordinates)
```

MinValue3D *Calculation of the Minimum Value of the modules.*

Description

This function calculates the minimum value of a set of modules.

Usage

```
MinValue3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The min value of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, AllModuleStatistics3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
MinValue3D(modules)
```

ModulePopulationStandardDeviation3D

Calculation of the Population Standard Deviation of the modules.

Description

This function calculates the population standard deviation of a set of input modules.

Usage

```
ModulePopulationStandardDeviation3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The population standard deviation of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, AllModuleStatistics3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
ModulePopulationStandardDeviation3D(modules)
```

ModulePopulationVariance3D

Calculation of the Population Variance of the modules.

Description

This function calculates the population variance of a set of input modules.

Usage

```
ModulePopulationVariance3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The population variance of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, AllModuleStatistics3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
ModulePopulationVariance3D(modules)
```

ModuleStandardDeviation3D

Calculation of the Standard Deviation of the modules.

Description

This function calculates the standard deviation of a set of input modules.

Usage

```
ModuleStandardDeviation3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The standard deviation of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, AllModuleStatistics3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
ModuleStandardDeviation3D(modules)
```

ModuleSum3D *Calculation of the Sum of the Modules.*

Description

This function calculates the sum of all modules.

Usage

```
ModuleSum3D(modules)
```

Arguments

modules	Vector containing the values of the module

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The sum of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, Range3D, AllModuleStatistics3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
ModuleSum3D(modules)
```

ModuleVariance3D *Calculation of the Variance of the modules.*

Description

This function calculates the variance of a set of input modules.

Usage

```
ModuleVariance3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The variance of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, AllModuleStatistics3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
ModuleVariance3D(modules)
```

NumberOfElements3D *Calculation of the Number of Elements of the modules.*

Description

This function calculates the Number of Elements of a set of modules

Usage

```
NumberOfElements3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The number of elements of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

AllModuleStatistics3D, MaxValue3D, MinValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
NumberOfElements3D(modules)
```

Range3D *Calculation of the Range of the modules.*

Description

This function calculates the range (difference between maximum and minimum value of the module) of a set of modules

Usage

```
Range3D(modules)
```

Arguments

modules	Vector containing the values of the module

Details

One way to get a set of modules from the X, Y and Z in the original position and end position (Coordinates X, Y and Z vector) or the colatitude and longitude, using the LoadData3D function.

Value

The range value of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, AllModuleStatistics3D, ModuleSum3D, ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
Range3D(modules)
```

RayleighTest3D *Rayleigh Test. Formal test of uniformity.*

Description

This function performs the Rayleigh test of uniformity.

Usage

```
RayleighTest3D(coord, Alpha = 0.05)
```

Arguments

coord	Matrix containing the values of the coordinates
Alpha	Value used to obtain the Rayleigh Value from the chi-square table. The values can be 0.05, 0.025, 0.01, 0.005, 0.001, 0.0005. The default is 0.05.

Details

This test detects a single modal direction in a sample of angles when the mean angles is unspecified. The hypothesis of uniformity is rejected if the mean module is very large. This test assumes that a larger mean module implies a more concentration around the mean, and therefore less probability that the data is uniformly distributed.

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

Value

Returns the probability value, and indicates whether or not to accept the hypothesis of uniformity.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Fisher N.I. , Lewis T. , Embleton, B.J.J. (1987) Statistical analysis of spherical data. Cambridge. Cambridge University Press.

Website http://gim.unex.es/VecStatGraphs3D/

See Also

AllAngleStatistics, AllModuleStatistics3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
coordinates<-dat[,4:6]
RayleighTest3D(coordinates, Alpha = 0.05)
```

SkewnessModuleCoefficient3D

Calculation of the skewness coefficient of the modules in 3D

Description

This function calculates the skewness coefficient and indicates the symmetry of the distribution from a set of input modules.

Usage

```
SkewnessModuleCoefficient3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to obtain a set of modules from X and Y coordinates of the vectors is to use the LoadData3D function.

A normal data distribution presents a skewness coefficient equal to 0.

Value

The value skewness coefficient of all input modules

Author(s)

Felicisimo, A.M., <amfeli@unex.es>, Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Cuartero, A., <acuartero@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

ArithmeticMean3D, StandardError3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance
ModulePopulationStandardDeviation3D, SkewnessModuleCoefficient3D, AllModuleStatistics3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
SkewnessModuleCoefficient3D(modules)
```

SphericalStandardError3D

Calculation of the Standard Error of the angles.

Description

This function calculates the standard error of the arithmetic mean of a set of input coordinates.

Usage

```
SphericalStandardError3D(coord)
```

Arguments

coord	Matrix containing the values of the coordinates

Details

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

Value

The standard error of all input coordinates.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Fisher N.I. , Lewis T. , Embleton, B.J.J. (1987) Statistical analysis of spherical data. Cambridge. Cambridge University Press.

Website http://gim.unex.es/VecStatGraphs3D/

See Also

MeanModule3D, MeanDirection3D, AllAngleStatistics, ConcentrationParameter3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
coordinates<-dat[,4:6]
SphericalStandardError3D(coordinates)
```

StandardError3D *Calculation of the Standard Error of the modules.*

Description

This function calculates the standard error of the arithmetic mean of a set of input modules.

Usage

```
StandardError3D(modules)
```

Arguments

modules	Vector containing the values of the modules

Details

One way to get a set of coordinates X, Y and Z of the origin position and end position (coordinates X, Y and Z of the vector) or of the colatitude and longitude, it is using the LoadData3D function.

Value

The standard error of all input modules.

Author(s)

Ruiz-Cuetos, J.C., <bilba_t@hotmail.com>, Polo, M.E., <mepolo@unex.es>, Rodriguez, P.G. <pablogr@unex.es>

References

Website http://gim.unex.es/VecStatGraphs3D/

See Also

NumberOfElements3D, MaxValue3D, MinValue3D, Range3D, ModuleSum3D, ArithmeticMean3D, AllModuleStatistics3D, ModuleStandardDeviation3D, ModuleVariance3D, ModulePopulationVariance3D, ModulePopulationStandardDeviation3D.

Examples

```
FileName<-system.file("data/XYZcoor.txt", package="VecStatGraphs3D")
dat<-LoadData3D(FileName, Type=1)
modules<-dat[,1]
StandardError3D(modules)
```

XYZcoor *GPS Data Set*

Description

Dataset obtained with a GPS, UNEX Merida.

Usage

```
XYZcoor
```

Format

A matrix containing 53 observations.

Source

UNEX Merida

References

Escuela Universitaria Politecnica, Caceres.

Index