Package ‘RTaxometrics’

September 9, 2017

Type Package
Title Taxometric Analysis
Version 2.3
Date 2017-09-09
Author John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu>
Maintainer John Ruscio <ruscio@tcnj.edu>
Description We provide functions to perform taxometric analyses. This package contains 44 functions, but only 5 should be called directly by users. CheckData() should be run prior to any taxometric analysis to ensure that the data are appropriate for taxometric analysis. RunTaxometrics() performs taxometric analyses for a sample of data. RunCCFIPer() performs a series of taxometric analyses to generate a CCFI profile. CreateData() generates a sample of categorical or dimensional data. ClassifyCases() assigns cases to groups using the base-rate classification method.
License MIT + file LICENSE
RoxygenNote 6.0.1
NeedsCompilation no
Repository CRAN
Date/Publication 2017-09-09 20:39:30 UTC

R topics documented:

AddVariance ........................................... 2
AssignMAMBAC ........................................ 3
AssignMAXEIG ......................................... 4
CalculateCCFI .......................................... 4
CalculateCCFIs ........................................ 5
CalculateFitDensities ................................. 6
CalculateKurtosis .................................... 6
CalculateLModeCCFI .................................. 7
CalculateMAMBAC ..................................... 8
CalculateMAXEIG ..................................... 8
CalculateMAXSLOPE .................................. 9
AddVariance

AddVariance(x, k, parameters)

Description

This function adds variance if necessary

Usage

AddVariance(x, k, parameters)
AssignMAMBAC

Arguments

- **x**  The supplied data matrix.
- **k**  The number of variables.
- **parameters**  The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Data with necessary variance added

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

Description

This function assigns variables to input/output configurations for MAMBAC analysis.

Usage

AssignMAMBAC(parameters)

Arguments

- **parameters**  The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Input/output variables per curve

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
AssignMAXEIG

Assigns variables to input/output for MAXEIG procedure

Description

This function assigns variables to input/output configurations for MAXEIG analysis.

Usage

AssignMAXEIG(parameters)

Arguments

parameters The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Input/output variables per curve

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

CalculateCCFI

Calculates CCFIs

Description

This function calculates CCFIs for a MAMBAC, MAXEIG, or MAXSLOPE curve

Usage

CalculateCCFI(curve, curve.dim, curve.cat)

Arguments

curve Empirical data curve
curve.dim Average curve for dimensional comparison data
curve.cat Average curve for categorical comparison data
Calculate CCFIs

Details
Called by higher-order functions; users do not need to call this function directly.

Value
CCFI value

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

Description
This function calculates CCFI results for CCFI profiles

Usage
CalculateCCFIs(x.results, x.dim.results, x.cat.results, parameters)

Arguments
- x.results: Empirical data results
- x.dim.results: Dimensional comparison data results
- x.cat.results: Categorical comparison data results
- parameters: The data and program parameters

Details
Called by higher-order functions; users do not need to call this function directly.

Value
CCFI values

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
CalculateFitDensities  Calculates fit

Description
This function calculates fit for L-Mode curves

Usage
CalculateFitDensities(shift, data)

Arguments

- shift  Horizontal shift
- data  Curves for empirical and comparison data

Details
Called by higher-order functions; users do not need to call this function directly.

Value
Fit value

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

CalculateKurtosis  Calculates kurtosis

Description
This function calculates the sample kurtosis of a distribution

Usage
CalculateKurtosis(x)

Arguments

- x  The data vector

Details
Called by higher-order functions; users do not need to call this function directly.
CalculateLModeCCFI

Value

The sample kurtosis of x

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

**CalculateLModeCCFI**  *Calculates L-Mode CCFI*

**Description**

This function calculates CCFI for an L-Mode curve

**Usage**

`CalculateLModeCCFI(curve.x, curve.y, curve.dim.x, curve.dim.y, curve.cat.x, curve.cat.y)`

**Arguments**

- `curve.x`  
  Empirical data curve, x
- `curve.y`  
  Empirical data curve, y
- `curve.dim.x`  
  Average curve for dimensional comparison data, x
- `curve.dim.y`  
  Average curve for dimensional comparison data, y
- `curve.cat.x`  
  Average curve for categorical comparison data, x
- `curve.cat.y`  
  Average curve for categorical comparison data, y

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

CCFI value

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
CalculateMAMBAC  \(\text{Calculates MAMBAC curve}\)

Description

This function calculates one MAMBAC curve

Usage

\[\text{CalculateMAMBAC}(\text{input}, \text{output}, \text{parameters})\]

Arguments

- input: Input indicator
- output: Output indicator
- parameters: The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

One MAMBAC curve

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

------------------------------

CalculateMAXEIG  \(\text{Calculates MAXEIG curve}\)

Description

This function calculates one MAXEIG curve

Usage

\[\text{CalculateMAXEIG}(\text{input}, \text{outputs}, \text{parameters})\]

Arguments

- input: Input indicator
- outputs: Output indicators
- parameters: The data and program parameters
CalculateMAXSLOPE

Details
Called by higher-order functions; users do not need to call this function directly.

Value
List object with one MAXEIG curve:

curve.x  x values
curve.y  y values

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

Description
This function calculates one MAXSLOPE curve

Usage
CalculateMAXSLOPE(x, curve)

Arguments

x  The data matrix
curve  Curve number

Details
Called by higher-order functions; users do not need to call this function directly.

Value
List object with one MAXSLOPE curve:

curve.x  x values
curve.y  y values

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
### CalculateSkew

**Description**

This function calculates the sample skewness of a distribution.

**Usage**

```r
CalculateSkew(x)
```

**Arguments**

- `x`: The data vector

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

The sample skewness of `x`.

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu>

Maintainer: John Ruscio <ruscio@tcnj.edu>

---

### CalculateValidity

**Description**

This function calculates the standardized mean difference between two groups (Cohen’s D).

**Usage**

```r
CalculateValidity(x.1, x.2)
```

**Arguments**

- `x.1`: Data for the first group
- `x.2`: Data for the second group

**Details**

Called by higher-order functions; users do not need to call this function directly.
CheckClassification

Value

The standardized mean difference between groups

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

Description

This function checks classification for problems, and terminates the program if necessary

Usage

CheckClassification(group, n)

Arguments

  group : Classification of cases
  n : Sample size

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Nothing; text output if problem occurs

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
CheckData

Description

This function checks whether the supplied empirical data set is appropriate for taxometric analysis, and provides descriptive statistics about the data set. If data do not meet certain requirements, the program prints warnings in the output, with details about which specific criteria are not met.

Usage

CheckData(x)

Arguments

x The supplied data matrix. Cases missing any data will be removed prior to analysis.

Details

This function should be called directly by users before performing any taxometric procedures.

Value

This program returns nothing, and provides text output only.

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangsi7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

Examples

# create or import data set
# creates a categorical data set
test.cat <- CreateData("cat")
# Checks data
CheckData(test.cat)

# creates a dimensional data set
test.dim <- CreateData("dim")
# Checks data
CheckData(test.dim)
CheckParameters  

**Description**

This function checks the parameter specifications for problems, and adjusts these parameters as needed.

**Usage**

```
CheckParameters(x, parameters)
```

**Arguments**

- `x`  
  The data matrix  
- `parameters`  
  The data and program parameters

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

Data parameters, adjusted as needed

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

ClassifyCases  

**Description**

This function assigns cases to groups using the base-rate classification technique. Cases are sorted according to their total scores on all indicators, and the highest-scoring cases are assigned to the taxon such that the proportion of taxon members equals the specified base rate estimate.

**Usage**

```
ClassifyCases(x, p, cols = Ø)
```
Arguments

- x: The supplied data matrix.
- p: The base rate estimate that will be used to classify cases.
- cols: The column numbers that contain variables.

Details

Users should call this function directly if they wish to assign cases to groups.

Value

Data matrix with a new classification variable.

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

References

Correlation among variables. The default value is 0.

Correlation among variables within the taxon. The default value is 0.

Correlation among variables within the complement. The default value is 0.

Parameter used to control asymmetry (scalar); sign indicates direction and absolute value indicates magnitude of skew (e.g., +/- .30 yields substantial asymmetry).

Parameter used to control tail weight (scalar); positive values yield tails that are longer/thinner than a standard normal curve, negative values do the reverse (e.g., +/- .15 is a substantial departure from normality).

Parameter used to create ordered categories, if nonzero (scalar); number of categories will be cuts + 1.

Whether to generate random values (the program default) or use uniformly distributed quantiles (T/F).

Random number seed; specifying the same seed enables users to generate and analyze identical data sets. The default value is 1.

Users should call this function directly if they wish to create an artificial data set.

Data matrix; k columns contain data, final column contains classification.

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

# creates a categorical data set
test.cat <- CreateData("cat")

# creates a dimensional data set
test.dim <- CreateData("dim")

Generates sample of correlated data with univariate g-and-h distributions.

CreateSample(n, k, r, g, h, uniform)
Arguments

- **n**: Sample size
- **k**: Number of variables
- **r**: Correlation among variables
- **g**: Parameter used to control asymmetry (scalar); sign indicates direction and absolute value indicates magnitude of skew (e.g., +/- .30 yields substantial asymmetry).
- **h**: Parameter used to control tail weight (scalar); positive values yield tails that are longer/thinner than a standard normal curve, negative values do the reverse (e.g., +/- .15 is a substantial departure from normality).
- **uniform**: Whether to generate random values (the program default) or use uniformly distributed quantiles (T/F).

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Sample of data

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

CreateVariable

*Creates a variable*

Description

Generates variable with g-and-h distribution.

Usage

CreateVariable(n, g, h, uniform)

Arguments

- **n**: Size of sample to create
- **g**: Parameter used to control asymmetry (scalar); sign indicates direction and absolute value indicates magnitude of skew (e.g., +/- .30 yields substantial asymmetry).
- **h**: Parameter used to control tail weight (scalar); positive values yield tails that are longer/thinner than a standard normal curve, negative values do the reverse (e.g., +/- .15 is a substantial departure from normality).
uniform  Whether to generate random values (the program default) or use uniformly distributed quantiles (T/F).

Details
Called by higher-order functions; users do not need to call this function directly.

Value
Single variable

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintain: John Ruscio <ruscio@tcnj.edu>

DisplayBaseRates  Displays base rates

Description
This function calculates and reports base rate estimates for taxometric analysis.

Usage
DisplayBaseRates(x.results, parameters)

Arguments
x.results  Empirical data results
parameters  The data and program parameters

Details
Called by higher-order functions; users do not need to call this function directly.

Value
This function returns nothing, and provides text output only

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintain: John Ruscio <ruscio@tcnj.edu>
DisplayCCFIs

Calculates and displays CCFIs

Description

This function calculates and reports CCFI results for taxometric analysis.

Usage

DisplayCCFIs(x.results, x.dim.results, x.cat.results, parameters)

Arguments

- `x.results`: Empirical data results
- `x.dim.results`: Dimensional comparison data results
- `x.cat.results`: Categorical comparison data results
- `parameters`: The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

This function returns nothing, and provides text output only.

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

DisplayPanels

Displays panels of graphs

Description

This function provides panels of graphs for taxometric analysis.

Usage

DisplayPanels(x.results, x.dim.results, x.cat.results, parameters)
DisplayProfiles

Arguments

- `x.results`: Empirical data results
- `x.dim.results`: Dimensional comparison data results
- `x.cat.results`: Categorical comparison data results
- `parameters`: The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

This function returns nothing, and provides graphical output only

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

DisplayProfiles  Plots CCFI Profiles

Description

This function plots CCFI profiles

Usage

DisplayProfiles(CCFIs, parameters)

Arguments

- `CCFIs`: CCFI values across base rates and procedures
- `parameters`: The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

This function returns nothing, and provides graphical output only

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
DisplaySpecifications  Displays analytic specifications

Description
This function provides text output for analytic specifications

Usage
DisplaySpecifications(parameters)

Arguments
parameters  The data and program parameters

Details
Called by higher-order functions; users do not need to call this function directly.

Value
This function returns nothing, and provides text output only

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

DisplayTextOutput  Displays CCFI profile output

Description
This function displays aggregated CCFIs and base rate estimates for CCFI profile

Usage
DisplayTextOutput(CCFIs, parameters)

Arguments
CCFIs  CCFI values across base rates and procedures
parameters  The data and program parameters

Details
Called by higher-order functions; users do not need to call this function directly.
EstimateLMode

Value

This function returns nothing, and provides text output only.

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wang7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

EstimateLMode

Estimates L-Mode base rate

Description

This function estimates the taxon base rate for an L-Mode curve.

Usage

EstimateLMode(curve.x, curve.y, parameters)

Arguments

curve.x X values of density
curve.y Y values of density
parameters The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

List of base rate estimates:
p.r Based on location of left mode
p.l Based on location of right mode
p.estimate mean

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wang7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
EstimateMAMBAC

Estimates MAMBAC base rate

Description

This function estimates the taxon base rate for a MAMBAC curve

Usage

EstimateMAMBAC(curve)

Arguments

curve MAMBAC curve

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Base rate estimate

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wang57@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

EstimateMAXEIG

Estimates MAXEIG base rate

Description

This function estimates the taxon base rate for a MAXEIG curve

Usage

EstimateMAXEIG(curve)

Arguments

curve MAXEIG curve

Details

Called by higher-order functions; users do not need to call this function directly.
EstimateMAXSLOPE

Value

Base rate estimate

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

EstimateMAXSLOPE

Estimates MAXSLOPE base rate

Description

This function estimates the taxon base rate for a MAXSLOPE curve

Usage

EstimateMAXSLOPE(curve.x, curve.y)

Arguments

curve.x X values of curve
curve.y Y values of curve

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Base rate estimate

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
### GenerateData

**Generates comparison data**

**Description**

This function generates a population of comparison data.

**Usage**

```
GenerateData(x, n, n.factors = 0, max.trials = 5, initial.multiplier = 1)
```

**Arguments**

- `x`: The data matrix
- `n`: Size of population to create
- `n.factors`: The number of factors used to reproduce correlations. The default value is 0.
- `max.trials`: Maximum number of trials. The default value is 5.
- `initial.multiplier`: Size of multiplier to adjust target correlations. The default value is 1.

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

Population of comparison data

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> 
Maintainer: John Ruscio <ruscio@tcnj.edu>

### PlotPanel

**Plots a panel of curves**

**Description**

This function plots a two-panel graph with results for empirical and comparison data.

**Usage**

```
PlotPanel(x.results, x.dim.results, x.cat.results, parameters, procedure)
```
**Arguments**

<table>
<thead>
<tr>
<th>arguments</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>x.results</code></td>
<td>Empirical data results</td>
</tr>
<tr>
<td><code>x.dim.results</code></td>
<td>Dimensional comparison data results</td>
</tr>
<tr>
<td><code>x.cat.results</code></td>
<td>Categorical comparison data results</td>
</tr>
<tr>
<td><code>parameters</code></td>
<td>The data and program parameters</td>
</tr>
<tr>
<td><code>procedure</code></td>
<td>Name of taxometric procedure</td>
</tr>
</tbody>
</table>

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

This function returns nothing, and provides graphical output only.

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

**ProcessProfile**

**Calculates CCFIs and base rates for CCFI profile**

**Description**

This function calculates the aggregated CCFI and base rate estimate for one CCFI profile

**Usage**

`ProcessProfile(CCFIs, parameters)`

**Arguments**

<table>
<thead>
<tr>
<th>arguments</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CCFIs</code></td>
<td>CCFI values across base rates for a single procedure</td>
</tr>
<tr>
<td><code>parameters</code></td>
<td>The data and program parameters</td>
</tr>
</tbody>
</table>

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

List of aggregated CCFI and base rate estimate

<table>
<thead>
<tr>
<th>column</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCFI</td>
<td>Aggregated CCFI</td>
</tr>
<tr>
<td><code>p.est</code></td>
<td>Base rate estimate</td>
</tr>
</tbody>
</table>
**RemoveMissingData**  
*Removes missing data*

**Description**

This function performs listwise deletion of missing data.

**Usage**

```
RemoveMissingData(x)
```

**Arguments**

- `x` The data matrix

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

Data after listwise deletion of missing data

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

**RunCCFIProfile**  
*Performs taxometric analyses to generate a CCFI profile*

**Description**

This function performs a series of taxometric analysis using categorical comparison data sets that vary in taxon base rates, and plots a profile of CCFI values across this range of base rates.

**Usage**

```
RunCCFIProfile(x, seed = 1, min.p = 0.025, max.p = 0.975, num.p = 39, 
n.pop = 1e+05, n.samples = 100, reps = 1, MAMBAC = TRUE, assign.MAMBAC = 1, 
n.cuts = 50, n.end = 25, MAXEIG = TRUE, assign.MAXEIG = 1, windows = 50, 
overlap = 0.9, LMode = TRUE, mode.l = -0.001, mode.r = 0.001, MAXSLOPE = FALSE, 
profile = TRUE)
```
RunCCFIProfile

Arguments

x
Supplied data matrix. Cases missing any data will be removed prior to analysis.

seed
Random number seed provided prior to analysis of empirical data as well as prior to generating each population of comparison data. The default value is 1.

min.p
Minimum base rate for CCFI profile. The default value is .025.

max.p
Maximum base rate for CCFI profile. The default value is .975.

num.p
Number of base rates for CCFI profile. The default value is 39.

n.pop
Size of the finite populations of categorical and dimensional comparison data. The default value is 100,000.

n.samples
Number of comparison data sets of each structure to generate and analyze. The default value is 100.

reps
Number of times to resort cases along the input indicator at random and redo the calculations (if tied scores are found), averaging to obtain final results. The default value is 1 if no tied scores are found, and 10 if tied scores are found.

MAMBAC
Whether the MAMBAC procedure is performed. The default value is TRUE.

assign.MAMBAC
How variables are assigned as input and output variables in the MAMBAC procedure. Variables may be used in all possible input-output pairings (assign.MAMBAC = 1), or variables may be summed to form the input variable (assign.MAMBAC = 2). The default value is 1.

n.cuts
The total number of cuts to make along the input variable when performing the MAMBAC procedure. The default value is 25.

n.end
The number of cases to set aside at each extreme along the input variable before making the first and last cuts when performing the MAMBAC procedure. The default value is 25.

MAXEIG
Whether the MAXEIG procedure is performed. The default value is TRUE if k is >= 3, and FALSE if k < 3.

assign.MAXEIG
How variables are assigned as input and output variables in the MAXEIG procedure. Variables may be used in all input-output triplets (assign.MAXEIG = 1), each variable may serve as input once (assign.MAXEIG = 2), or variables may be summed to form the input (assign.MAXEIG = 3). The default value is 1.

windows
The number of overlapping windows to use when performing the MAXEIG procedure. The default value is 50.

overlap
The amount of overlap between windows when performing the MAXEIG procedure. The default value is .90.

LMode
Whether the L-Mode procedure is performed. The default value is TRUE if k is >= 3, and FALSE if k < 3.

mode.l
Position beyond which to search for the left mode when performing the L-Mode procedure. The default value is -.001.

mode.r
Position beyond which to search for the right mode when performing the L-Mode procedure. The default value is .001.

MAXSLOPE
Whether the MAXSLOPE procedure is performed. The default value is FALSE if k >= 3, and TRUE if k < 3.

profile
Whether a CCFI profile is generated. The default value is TRUE.
This function should be called directly by users who wish to perform taxometric analyses to generate a CCFI profile.

This program returns nothing, and provides text and graphical output only.

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

### Details

This function performs factor analysis.

### Usage

```r
code
RunFactorAnalysis(x, cor.matrix = FALSE, n.factors = 0, max.iter = 50, criterion = 0.01)
```

### Arguments

- `x`: The data or correlation matrix.
- `cor.matrix`: Whether `x` is a correlation matrix. The default is `FALSE`.
- `n.factors`: The number of factors to use. The default value is `0`.
- `max.iter`: The maximum number of iterations. The default value is `50`.
- `criterion`: Acceptably small change in h2 between interations. The default value is `.01`.

### Details

Called by higher-order functions; users do not need to call this function directly.

### Value

List of factor loadings and number of factors:

- `loadings`: The factor loadings
- `factors`: The number of factors
**RunLMode**

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

**RunLMode**

*Performs L-Mode*

---

**Description**

This function performs the L-Mode analysis

**Usage**

```r
runlmode(x)
```

**Arguments**

- `x` The data matrix

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

L-Mode curve:

- `curve.x` X values of curve
- `curve.y` Y values of curve

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

**References**

**RunMAMBAC**

**Performs MAMBAC**

**Description**

This function performs the MAMBAC analysis

**Usage**

`RunMAMBAC(x, parameters)`

**Arguments**

- `x` The data matrix
- `parameters` The data and program parameters

**Details**

Called by higher-order functions; users do not need to call this function directly.

**Value**

Panel of MAMBAC curves

**Author(s)**

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

**References**


---

**Run MAXEIG**

**Performs MAXEIG**

**Description**

This function performs the MAXEIG analysis

**Usage**

`RunMAXEIG(x, parameters)`
Arguments

x The data matrix
parameters The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

Panel of MAXEIG curves:

curve.x X values of curve
curve.y Y values of curve

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

References


Description

This function performs the MAXSLOPE analysis

Usage

RunMAXSLOPE(x)

Arguments

x The data matrix

Details

Called by higher-order functions; users do not need to call this function directly.
RunProcedures

Value

Panel of MAXSLOPE curves
curve.x X values of curve
curve.y Y values of curve

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

References


RunProcedures runs taxometric procedures for empirical data

Description

This function runs the MAMBAC, MAXEIG, L-Mode, and MAXSLOPE analyses for empirical data

Usage

RunProcedures(x, parameters)

Arguments

x The data matrix
parameters The data and program parameters

Details

Called by higher-order functions; users do not need to call this function directly.

Value

A list of curve-level data for each procedure performed:
MAMBAC MAMBAC curve
MAXEIG.x X values of MAXEIG curve
MAXEIG.y Y values of MAXEIG curve
LMode.x X values of LMode curve
LMode.y Y values of LMode curve
MAXSLOPE.x X values of MAXSLOPE curve
MAXSLOPE.y Y values of MAXSLOPE curve
RunProceduresComp

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

---

RunProceduresComp  Runs taxometric procedures for comparison data

---

Description
This function runs the MAMBAC, MAXEIG, L-Mode, and MAXSLOPE analyses for comparison data

Usage
RunProceduresComp(x, parameters)

Arguments
- x The data matrix
- parameters The data and program parameters.

Details
Called by higher-order functions; users do not need to call this function directly.

Value
A list of averaged curves for each procedure performed:

- `MAMBAC` MAMBAC curve
- `MAXEIG.x` X values of MAXEIG curve
- `MAXEIG.y` Y values of MAXEIG curve
- `LMode.x` X values of LMode curve
- `LMode.y` Y values of LMode curve
- `MAXSLOPE.x` X values of MAXSLOPE curve
- `MAXSLOPE.y` Y values of MAXSLOPE curve

Author(s)
John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
RunTaxometrics  

**Taxometric analysis for a sample of data**

**Description**

Performs taxometric analysis for a sample of data and provides text (analytic specifications, CCFI values, base rate estimates) and graphical (panels of empirical data curves superimposed above comparison data curves) output. By default, the function will run MAMBAC, MAXEIG, and L-Mode, unless only 2 variables are provided, in which case the program will run MAMBAC and MAXSLOPE.

**Usage**

```r
RunTaxometrics(x, seed = 1, n.pop = 1e+05, n.samples = 100, reps = 1,
               MAMBAC = TRUE, assign.MAMBAC = 1, n.cuts = 50, n.end = 25, MAXEIG =
               TRUE, assign.MAXEIG = 1, windows = 50, overlap = 0.9, LMode = TRUE, mode.l =
               -0.001, mode.r = 0.001, MAXSLOPE = FALSE)
```

**Arguments**

- **x**: The supplied data matrix. Cases missing any data will be removed prior to analysis.
- **seed**: Random number seed provided prior to analysis of empirical data as well as prior to generating each population of comparison data. The default value is 1.
- **n.pop**: Size of the finite populations of categorical and dimensional comparison data. The default value is 100,000.
- **n.samples**: Number of comparison data sets of each structure to generate and analyze. The default value is 100.
- **reps**: Number of times to resort cases along the input indicator at random and redo the calculations (if tied scores are found), averaging to obtain final results. The default value is 1 if no tied scores are found, and 10 if tied scores are found.
- **MAMBAC**: Whether the MAMBAC procedure is performed. The default value is TRUE.
- **assign.MAMBAC**: How variables are assigned as input and output variables in the MAMBAC procedure. Variables may be used in all possible input-output pairings (assign.MAMBAC = 1), or variables may be summed to form the input variable (assign.MAMBAC = 2). The default value is 1.
- **n.cuts**: The total number of cuts to make along the input variable when performing the MAMBAC procedure. The default value is 25.
- **n.end**: The number of cases to set aside at each extreme along the input variable before making the first and last cuts when performing the MAMBAC procedure. The default value is 25.
- **MAXEIG**: Whether the MAXEIG procedure is performed. The default value is TRUE if k is >= 3, and FALSE if k < 3.
SummarizeDist

assign.MAXEIG  How variables are assigned as input and output variables in the MAXEIG procedure. Variables may be used in all input-output triplets (assign.MAXEIG = 1), each variable may serve as input once (assign.MAXEIG = 2), or variables may be summed to form the input (assign.MAXEIG = 3). The default value is 1.

windows  The number of overlapping windows to use when performing the MAXEIG procedure. The default value is 50.

overlap  The amount of overlap between windows when performing the MAXEIG procedure. The default value is .90.

LMode  Whether the L-Mode procedure is performed. The default value is TRUE if k is >= 3, and FALSE if k < 3.

mode.l  Position beyond which to search for the left mode when performing the L-Mode procedure. The default value is -.001.

mode.r  Position beyond which to search for the right mode when performing the L-Mode procedure. The default value is .001.

MAXSLOPE  Whether the MAXSLOPE procedure is performed. The default value is FALSE if k >= 3, and TRUE if k < 3.

Details

This function should be called directly by users who wish to perform taxometric analyses for a sample of data.

Value

This program returns nothing, and provides text and graphical output only.

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wangs7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>

| SummarizeDist | Summarizes distribution |

Description

This function calculates the sample mean, standard deviation, skewness, and kurtosis.

Usage

SummarizeDist(x)

Arguments

x  The data vector
Details

Called by higher-order functions; users do not need to call this function directly.

Value

The sample mean, standard deviation, skewness, and kurtosis of x.

Author(s)

John Ruscio <ruscio@tcnj.edu> and Shirley Wang <wang7@tcnj.edu> Maintainer: John Ruscio <ruscio@tcnj.edu>
Index

AddVariance, 2
AssignMAMBC, 3
AssignMAXEIG, 4

CalculateCCFI, 4
CalculateCCFIs, 5
CalculateFitDensities, 6
CalculateKurtosis, 6
CalculateLModeCCFI, 7
CalculateMAMBC, 8
CalculateMAXEIG, 8
CalculateMAXSLOPE, 9
CalculateSkew, 10
CalculateValidity, 10
CheckClassification, 11
CheckData, 12
CheckParameters, 13
ClassifyCases, 13
CreateData, 14
CreateSample, 15
CreateVariable, 16

DisplayBaseRates, 17
DisplayCCFIs, 18
DisplayPanels, 18
DisplayProfiles, 19
DisplaySpecifications, 20
DisplayTextOutput, 20

EstimateLMode, 21
EstimateMAMBC, 22
EstimateMAXEIG, 22
EstimateMAXSLOPE, 23

GenerateData, 24

PlotPanel, 24
ProcessProfile, 25

RemoveMissingData, 26
RunCCFIProfile, 26

RunFactorAnalysis, 28
RunLMode, 29
RunMAMBC, 30
RunMAXEIG, 30
RunMAXSLOPE, 31
RunProcedures, 32
RunProceduresComp, 33
RunTaxometrics, 34

SummarizeDist, 35