Package ‘ABCanalysis’

March 13, 2017

Type Package

Title Computed ABC Analysis

Version 1.2.1

Date 2017-03-13

Author Michael Thrun, Jorn Lotsch, Alfred Ultsch

Maintainer Florian Lerch <lerch@mathematik.uni-marburg.de>

Description For a given data set, the package provides a novel method of computing precise limits to acquire subsets which are easily interpreted. Closely related to the Lorenz curve, the ABC curve visualizes the data by graphically representing the cumulative distribution function. Based on an ABC analysis the algorithm calculates, with the help of the ABC curve, the optimal limits by exploiting the mathematical properties pertaining to distribution of analyzed items. The data containing positive values is divided into three disjoint subsets A, B and C, with subset A comprising very profitable values, i.e. largest data values (``the important few''), subset B comprising values where the yield equals to the effort required to obtain it, and the subset C comprising of non-profitable values, i.e., the smallest data sets (``the trivial many''). Package is based on ``Computed ABC Analysis for rational Selection of most informative Variables in multivariate Data'', PLoS One. Ultsch. A., Lotsch J. (2015) <DOI:10.1371/journal.pone.0129767>.

Imports plotrix

Depends R (>= 2.10)

License GPL-3

LazyLoad yes

URL https://www.uni-marburg.de/fb12/datenbionik/software-en

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2017-03-13 14:31:38

R topics documented:

ABCanalysis-package . . . 2
ABCanalysis . . . 3
ABCanalysis4curve . . . 4
ABCanalysisPlot . . . 5
ABCcleanData . . . 7
ABCcurve . . . 8
ABCplot . . . 9
ABCRemoveSmallYields . . . 10
calculatedABCanalysis . . . 11
Gini4ABC . . . 12
GiniIndex . . . 13
SwissInhabitants . . . 13

Index **15**

`ABCanalysis-package` *Computed ABC analysis*

Description

Computed ABC Analysis allows the optimal calculation of three disjoint subsets A,B,C in data sets containing positive values:

subset A containing few most profitable values, i.e. largest data values ("the important few"), subset B containing data, where the profit gain equals effort required to obtain this gain, and the subset C of non-profitable values, i.e. the smallest data sets ("the trivial many").

This package calculates the three subsets A, B and C by means of an algorithm based on statistically valid definitions of thresholds for the three sets A,B and C.

Note

Check out our new Umatrix package for visualisation and clustering of high-dimensional data on our Webpage.

Author(s)

Michael Thrun, Jorn Lotsch, Alfred Ultsch

http://www.uni-marburg.de/fb12/datenbionik

<mthrun@mathematik.uni-marburg.de>

References

Ultsch. A ., Lotsch J.: Computed ABC Analysis for Rational Selection of Most Informative Variables in Multivariate Data, PloS one, Vol. 10(6), pp. e0129767. doi 10.1371/journal.pone.0129767, 2015.

Examples

```
  data("SwissInhabitants")
abc=ABCanalysis(SwissInhabitants,PlotIt=TRUE)
SetA=SwissInhabitants[abc$Aind]
SetB=SwissInhabitants[abc$Bind]
SetC=SwissInhabitants[abc$Cind]
```

`ABCanalysis` *Computed ABC analysis: calculates a division of the data in 3 classes A, B and C*

Description

divide the Data in 3 classes A, B and C such that

A=Data[Aind] : with low effort much yield

B=Data[Bind] : yield and effort are about equal

C=Data[Cind] : with much effort low yield

Usage

```
ABCanalysis(Data,ABCcurvedata,PlotIt=FALSE)
```

Arguments

`Data`	vector(1:n) describes an array of data: n cases in rows of one variable, if matrix or dataframe then first column will be used.
`ABCcurvedata`	only for internal usage, list from ABCcurve
`PlotIt`	default(FALSE), if variable is used, a plot is made, set with arbitrary value

Details

Pareto point: Minimum distance to (0,1) = minimal unrealized potential

BreakEven Point: `B_x` is the x value of the point, where the slope of ABCcurve equals one.

For further description to `p` in variable `AlimitIndInInterpolation` see ABCcurve

Value

Output is of type list which parts are described in the following

`Aind`	vector [1:j], A==Data(Aind) : with little effort much Yield
`Bind`	vector [1:l], B==Data(Bind) : effort and Yield are balanced
`Cind`	(vector [1:m], C==Data(Cind) : much effort for little Yield
`ABexchanged`	Boolean, TRUE if Point A is the Break Even and point B is the Pareto Point, FALSE otherwise

`A`	c(Ax,Ay), Pareto point or BreakEven Point indicated by ABexchanged
`B`	c(Bx,By), Pareto point or BreakEven Point indicated by ABexchanged
`C`	Submarginal point: minimum distance to `[B_x,1]`
`smallestAData`	Boundary AB, defined by point A or B with ABexchanged
`smallestBData`	Boundary BC, defined by point C
`AlimitIndInInterpolation`	index of AB Boundary in [p, ABC], the interpolation of the ABC plot
`BlimitIndInInterpolation`	index of BC Boundary in [p, ABC], the interpolation of the ABC plot

Author(s)

Michael Thrun

http://www.uni-marburg.de/fb12/datenbionik

References

Ultsch. A ., Lotsch J.: Computed ABC Analysis for Rational Selection of Most Informative Variables in Multivariate Data, PloS one, Vol. 10(6), pp. e0129767. doi 10.1371/journal.pone.0129767, 2015.

See Also

ABCplot

Examples

```
  data("SwissInhabitants")
abc=ABCanalysis(SwissInhabitants,PlotIt=TRUE)
A=abc$Aind
B=abc$Bind
C=abc$Cind
Agroup=SwissInhabitants[A]
Bgroup=SwissInhabitants[B]
Cgroup=SwissInhabitants[C]
```

ABCanalysis4curve *calculate ABC Analysis from a given curve.*

Description

calculate points A B C of the ABC Analysis from a given curve.

Arguments

`p[1:m]`	a vector of values specifying where interpolation took place
`ABC[1:m]`	given values of the curve at positions from p

Value

BreakEvenPunktIndex = BreakEvenPunktIndex, ParetoPunktIndex = ParetoPunktIndex, SubmarginalPunktIndex = SubmarginalPunktIndex, ABx = Effort[AB], ABy = Yield[AB], BCx = Effort[BC], BCy = Yield[BC], Bx = Effort[B], By = Yield[B]))

`BreakEvenPunktIndex`	Index of breakeven point
`ParetoPunktIndex`	Index of pareto point
`SubmarginalPunktIndex`	Index of submarginal point
`ABx`	Position of AB point on x axis
`ABy`	Position of AB point on y axis
`BCx`	Position of BC point on x axis
`BCy`	Position of BC point on y axis
`Bx`	Position of the unused point (breakeven or pareto) on the x axis
`By`	Position of the unused point (breakeven or pareto) on the y axis

Author(s)

Florian Lerch

`ABCanalysisPlot` *Displays ABC plot with ABCanalysis*

Description

Displays ABC Curve : cumulative percentage of largest Data (effort) vs cumlative percentage of sum of largest data (yield) with set limits generated by an calculated ABCanalysis.

Usage

```
ABCanalysisPlot(Data, LineType = 0, LineWidth = 3,
ShowUniform = TRUE,title, limits = TRUE, MarkPoints = TRUE,
ABCcurvedata,ResetPlotDefaults=TRUE)
```

Arguments

`Data`	vector[1:n] describes an array of data: n cases in rows of one variable
`LineType`	integer, optional, for plot default: LineType=0 for solid line; for other line codes see documentation about pch
`LineWidth`	integer, optional, width of Line, see `lwd` in par
`ShowUniform`	boolean, optional, the ABC curve of the uniform distribution is shown in plot if TRUE (default)
`title`	string, optional, see parameter `main` in plot
`limits`	boolean, = TRUE, lines of division in A, B and C are drawn, default = FALSE
`MarkPoints`	boolean, optional, default= TRUE, Mark the three points of interest
`ABCcurvedata`	optional, see ABCcurve
`ResetPlotDefaults`	optional, default =TRUE. If ResetPlotDefaults=FALSE, multiple plots in one window possible, but no resetting of plot to default parameters.

Value

object is a list of items with

`ABC`	Output of ABCplot
`ABCanalysis`	Output of ABCanalysis

Note

The Break Even point is always marked with a green star.

The diagonal from (0,1) to (1,0) is the equilibrium, where effort equals yield.

Author(s)

Michael Thrun

http://www.uni-marburg.de/fb12/datenbionik

See Also

`ABCanalysis`

Examples

```
## Standard Example
data("SwissInhabitants")
abc=ABCanalysisPlot(SwissInhabitants)
## Multiple plots in one Window:
m=runif(4,100,200)
s=runif(4,1,10)
Data=sapply(1:4,FUN=function(x,m,s) rnorm(1000,m,s),m,s)
# windows() #screen devices should not be used in examples etc
par(mfrow=c(2,2))
```

```
for (i in 1:4)
{
ABCanalysisPlot(Data[,i],ResetPlotDefaults=FALSE)
}
```

`ABCcleanData` *Data cleaning for ABC analysis*

Description

Only the first column of Data is used, anything not beeinh positive numerical value is set to zero

Usage

```
ABCcleanData(Data)
```

Arguments

`Data`	vector[1:n] describes an array of data: n cases in rows of one variable

Details

Data <0 are set to zero, non-numeric values (NA,NaN,etc.) in Data are set to zero strings and chars are set to zero infinitive numbers are set to max(Data)

Value

Output is of type list which's parts are described in the following

`CleanedData`	vector [1:m], columnvector containing Data>=0 and zeros for all NA, NaN and negative values in Data(1:n)
`Data2CleanInd`	vector [1:k], Index such that CleanedData = nantozero(Data(Data2CleanInd))
`RemovedInd`	vector [1:l], Index such that Data(RemovedInd) is the data that has been removed if RemoveSmallYields==1

Author(s)

http://www.uni-marburg.de/fb12/datenbionik

Michael Thrun

ABCcurve *calculates ABC Curve*

Description

Calculates cumulative percentage of largest data (effort) and cumulative percentages of sum of largest Data (yield) with spline interpolation (second order, piecewise) of values in-between.

Usage

```
ABCcurve(Data, p)
```

Arguments

`Data`	vector[1:n] describes an array of data: n cases in rows of one variable
`p`	optional, an vector of values specifying where interpolation takes place, created by `seq` of package base

Value

Output is of type list which parts are described in the following

`Curve`	A list with `Effort`:vector [1:k], cumulative population in percent `Yield`: vector [1:k], cumulative high data in percent
`CleanedData`	vector [1:m], columnvector containing Data>=0 and zeros for all NA, NaN and negative values in Data(1:n)
`Slope`	A list with `p`: X-values for spline interpolation, defualt: p = (0:0.01:1) `dABC`: first deviation of the functio ABC(p)=Effort(Yield

Author(s)

Michael Thrun

http://www.uni-marburg.de/fb12/datenbionik

References

Ultsch. A ., Lotsch J.: Computed ABC Analysis for Rational Selection of Most Informative Variables in Multivariate Data, PloS one, Vol. 10(6), pp. e0129767. doi 10.1371/journal.pone.0129767, 2015.

ABCplot — *displays an ABC Curve as an alternative to an Lorenz curve*

Description

Plots cumulative percentage of largest data (effort) vs. cumulative percentage of sum of largest data (yield)

Usage

```
ABCplot(Data, LineType = 0, LineWidth = 3, ShowUniform = TRUE,
 title, ABCcurvedata,defaultAxes = TRUE)
```

Arguments

`Data`	vector[1:n], describes an array of data: n cases in rows of one variable
`LineType`	for plot default: LineType=0 for a line, other line codes see documentation about `pch` in par
`LineWidth`	integer, width of Line, see `lwd` in par
`ShowUniform`	bool, =TRUE: the ABC curve of the uniform distribution is shown in plot
`title`	string, optional, see parameter `main` in plot
`ABCcurvedata`	optional, see ABCcurve
`defaultAxes`	optional, boolean, see parameter `axes` in plot

Value

Output is of type list which parts are described in the following

`ABCx`	vector [1:k], cumulative population in percent
`ABCy`	vector [1:k], cumulative high Data in percent

Note

The diagonal from (1,0) to (0,1) is the Equilibrium, where effort equals yield

Author(s)

Michael Thrun

http://www.uni-marburg.de/fb12/datenbionik

Examples

```
data("SwissInhabitants")
vec=ABCplot(SwissInhabitants)
```

ABCRemoveSmallYields *Extended Data cleaning for ABC analysis*

Description

Only the first column of Data is used, anything not beeing positive numerical value is set to zero

Usage

```
ABCRemoveSmallYields(Data,CumSumSmallestPercentage)
```

Arguments

`Data`	vector[1:n] describes an array of data: n cases in rows of one variable
`CumSumSmallestPercentage`	(default =0.5),the smallest data up to a cumulated sum of less than CumSumSmallestPercentage

Details

Data <0 are set to zero, non-numeric values (NA,NaN,etc.) in Data are set to zero strings and chars are set to zero infinitive numbers are set to max(Data) the smallest data up to a cumulated sum of less than CumSumSmallestPercentage of the total sum (yield) is removed

Value

Output is of type list which's parts are described in the following

`SubstantialData`	columnvector containing Data>=0 and zeros for all NaN and negative values in Data(1:n)
`Data2CleanInd`	Index such that SubstantialData = nantozero(Data(Data2SubstantialInd))
`RemovedInd`	Data(RemovedInd) is the data that has been removed

Author(s)

http://www.uni-marburg.de/fb12/datenbionik

Michael Thrun

calculatedABCanalysis *Computed ABC analysis: calculates a division of the data in 3 classes A, B and C*

Description

divide the Data in 3 classes A, B and C such that

A=Data[Aind] : with low effort much yield

B=Data[Bind] : yield and effort are about equal

C=Data[Cind] : with much effort low yield

Usage

```
calculatedABCanalysis(Data)
```

Arguments

`Data`	vector(1:n) describes an array of data: n cases in rows of one variable, if matrix or dataframe then first column will be used.

Details

Pareto point: Minimum distance to (0,1) = minimal unrealized potential

BreakEven Point: `B_x` is the x value of the point, where the slope of ABCcurve equals one.

For further description to `p` in variable `AlimitIndInInterpolation` see ABCcurve

Value

Output is of type list which parts are described in the following

`Aind`	vector [1:j], A==Data(Aind) : with little effort much Yield
`Bind`	vector [1:l], B==Data(Bind) : effort and Yield are balanced
`Cind`	(vector [1:m], C==Data(Cind) : much effort for little Yield
`smallestAData`	Boundary AB, defined by point A or B with ABexchanged
`smallestBData`	Boundary BC, defined by point C

Author(s)

Michael Thrun

http://www.uni-marburg.de/fb12/datenbionik

References

Ultsch. A ., Lotsch J.: Computed ABC Analysis for Rational Selection of Most Informative Variables in Multivariate Data, PloS one, Vol. 10(6), pp. e0129767. doi 10.1371/journal.pone.0129767, 2015.

See Also

ABCanalysis

Examples

```
  data("SwissInhabitants")
abc=calculatedABCanalysis(SwissInhabitants)
A=abc$Aind
B=abc$Bind
C=abc$Cind
Agroup=SwissInhabitants[A]
Bgroup=SwissInhabitants[B]
Cgroup=SwissInhabitants[C]
```

Gini4ABC *Gini index*

Description

Gini index for an ABC curve

Usage

```
Gini4ABC(p, ABC)
```

Arguments

p	vector [1:k], cumulative population in percent
ABC	vector [1:k], cumulative high data in percent

Value

Gini gini index i.e. the integral over ABC(p) / 0.5 *100

given in percent i.e in [0..100]

Author(s)

FL?MT?

GiniIndex *Gini-Index*

Description

calculation of the Gini-Index from Data

Usage

```
GiniIndex(Data,p)
```

Arguments

`Data`	vector[1:n] describes an array of data: n cases in rows of one variable
`p`	optional, an vector of values specifying where interpolation takes place, created by `seq` of package base

Details

uses ABCcurve and Gini4ABC

Value

`Gini`	gini index i.e. the integral over Area *200 -100 given in percent i.e in [0..100]
`p`	vector [1:k], cumulative population in percent
`ABC`	vector [1:k], cumulative high data in percent
`CleanedData`	vector [1:m], columnvector containing Data>=0 and zeros for all NA, NaN and negative values in Data(1:n)

Author(s)

Michael Thrun

SwissInhabitants *SwissInhabitants in 1900*

Description

Number of inhabitants in the 2896 villages of Switzerland in the year 1900.

Usage

```
data("SwissInhabitants")
```

Details

This data set consists of the number of inhabitants in the 2896 communes, i.e. cities and villages, in the year 1900. The individual count is the total number of persons living in the particular commune. The data set is unordered for anonymity reasons. The data set has been used as part of a larger data set to identify patterns of concentration in Switzerland (see reference).

Source

Schuler,M., Ullmann, D. Eidgenossische Volkszahlung:Bevoelkerungsentwicklung der Gemeinden, Bundesamt fur Statistik, Neuchatel, Switzerland, 2002

References

Behnisch, M., Ultsch, A.: Population Patterns in Switzerland 1850-2000, in: Gaul, W. et al (Eds), Advances in Data Analysis, Data Handling and Business Intelligence, Springer, Heidelberg, pp. 163-173, 2010.

Examples

```
data(SwissInhabitants)
## maybe str(SwissInhabitants) ; plot(SwissInhabitants) ...
```

Index

∗Topic **ABC analysis**
ABCanalysis, 3
ABCanalysisPlot, 5
ABCplot, 9
calculatedABCanalysis, 11
∗Topic **ABC curve**
ABCcurve, 8
∗Topic **ABCanalysis**
ABCanalysis, 3
ABCanalysisPlot, 5
calculatedABCanalysis, 11
∗Topic **ABCcurve**
ABCcurve, 8
∗Topic **ABC**
ABCanalysis, 3
ABCplot, 9
calculatedABCanalysis, 11
∗Topic **Computed ABC analysis**
calculatedABCanalysis, 11
∗Topic **Lorenz curve**
ABCanalysis, 3
ABCcurve, 8
ABCplot, 9
calculatedABCanalysis, 11
∗Topic **Lorenz**
ABCanalysis, 3
ABCcurve, 8
ABCplot, 9
calculatedABCanalysis, 11
∗Topic **datasets,SwissInhabitants,SwissInhabitants1900**
SwissInhabitants, 13
∗Topic **package**
ABCanalysis-package, 2

ABCanalyse *(*ABCanalysis-package*)*, 2
ABCanalysis, 3, *6*, *12*
ABCanalysis-package, 2
ABCanalysis4curve, 4
ABCanalysisPlot, 5
ABCcleanData, 7
ABCcurve, *3*, *6*, 8, *9*, *11*
ABCplot, *4*, *6*, 9
ABCRemoveSmallYields, 10

calculatedABCanalysis, 11

dbt.ABC *(*ABCanalysis-package*)*, 2
dbt.ABCanalyse *(*ABCanalysis-package*)*, 2
dbt.ABCanalysis *(*ABCanalysis-package*)*, 2

Gini4ABC, 12
GiniIndex, 13

par, *6*, *9*
plot, *6*, *9*

seq, *8*, *13*
SwissInhabitants, 13
SwissInhabitants1900 *(*SwissInhabitants*)*, 13