Package 'conjoint'

June 10, 2025

Version 1.42

Title An Implementation of Conjoint Analysis Method

Description This is a simple R package that allows to measure the stated preferences using traditional conjoint analysis method.

License GPL

URL https://github.com/packagesR/conjoint

Imports AlgDesign, stats, grDevices, graphics, fpc, broom, ggplot2, cluster, ggfortify

Encoding UTF-8

NeedsCompilation no

Author Andrzej Bak [aut], Tomasz Bartlomowicz [aut, cre]

Maintainer Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Repository CRAN

Date 2025-06-22

Date/Publication 2025-06-09 22:20:02 UTC

Contents

`caBTL` *Function caBTL estimates participation (market share) of simulation profiles*

Description

Function caBTL estimates participation of simulation profiles using probabilistic model BTL (Bradley-Terry-Luce). Function returns vector of percentage participations. The sum of participation should be 100%.

Usage

```
caBTL(sym, y, x)
```

Arguments

`sym`	matrix of simulation profiles
`y`	matrix of preferences
`x`	matrix of profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caLogit, caMaxUtility and ShowAllSimulations

Examples

```
#Example 1
library(conjoint)
data(tea)
simutil<-caBTL(tsimp,tpref,tprof)
print("Percentage participation of profiles: ", quote=FALSE)
print(simutil)

#Example 2
library(conjoint)
data(chocolate)
simutil<-caBTL(csimp,cpref,cprof)
print("Percentage participation of profiles:", quote=FALSE)
print(simutil)

#Example 3
library(conjoint)
data(chocolate)
ShowAllSimulations(csimp,cpref,cprof)

#Example 4
#library(conjoint)
#data(journey)
#ShowAllSimulations(jsimp,jpref,jprof)
```

---

caEncodedDesign *Function caEncodedDesign encodes full or fractional factorial design*

Description

Function caEncodedDesign encodes full or fractional factorial design. Function converts design of experiment to matrix of profiles.

Usage

```
caEncodedDesign(design)
```

Arguments

design	design of experiment returned by caFactorialDesign function

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caFactorialDesign and caRecreatedDesign

Examples

```
#Example 1
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
design=caFactorialDesign(data=experiment,type="orthogonal")
print(design)
code=caEncodedDesign(design)
print(code)
print(cor(code))
```

caFactorialDesign *Function caFactorialDesign creates full or fractional factorial design*

Description

Function caFactorialDesign creates full or fractional factorial design. Function can return orthogonal factorial design.

Usage

```
caFactorialDesign(data, type="null", cards=NA, seed=123)
```

Arguments

data	experiment whose design consists of two or more factors, each with with 2 or more discrete levels
type	type of factorial design (possible values: "full", "fractional", "ca", "aca", "orthogonal"; default value: type="null")
cards	number of experimental runs
seed	seed settings (default value: seed=123)

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caEncodedDesign and caRecreatedDesign

Examples

```
#Example 1
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
design=caFactorialDesign(data=experiment,type="full")
print(design)
print(cor(caEncodedDesign(design)))

#Example 2
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
design=caFactorialDesign(data=experiment)
print(design)
print(cor(caEncodedDesign(design)))

#Example 3
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
design=caFactorialDesign(data=experiment,type="orthogonal")
print(design)
print(cor(caEncodedDesign(design)))

#Example 4
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
design=caFactorialDesign(data=experiment,type="fractional",cards=16)
print(design)
print(cor(caEncodedDesign(design)))

#Example 5
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
```

```
design=caFactorialDesign(data=experiment,type="fractional")
print(design)
print(cor(caEncodedDesign(design)))

#Example 6
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
design=caFactorialDesign(data=experiment,type="ca")
print(design)
print(cor(caEncodedDesign(design)))

#Example 7
library(conjoint)
experiment<-expand.grid(
price=c("low","medium","high"),
variety=c("black","green","red"),
kind=c("bags","granulated","leafy"),
aroma=c("yes","no"))
design=caFactorialDesign(data=experiment,type="aca")
print(design)
print(cor(caEncodedDesign(design)))
```

`caImportance` *Function caImportance calculates importance of all attributes*

Description

Function caImportance calculates importance of all attributes. Function returns vector of percentage attributes' importance and corresponding chart (barplot). The sum of importance should be 100%.

Usage

```
caImportance(y, x)
```

Arguments

y	matrix of preferences
x	matrix of profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

Conjoint

Examples

```
#Example 1
library(conjoint)
data(tea)
imp<-caImportance(tprefm,tprof)
print("Importance summary: ", quote=FALSE)
print(imp)
print(paste("Sum: ", sum(imp)), quote=FALSE)

#Example 2
library(conjoint)
data(chocolate)
imp<-caImportance(cprefm,cprof)
print("Importance summary: ", quote=FALSE)
print(imp)
print(paste("Sum: ", sum(imp)), quote=FALSE)

#Example 3
library(conjoint)
data(journey)
imp<-caImportance(jpref[1,],jprof)
print("Importance summary of first respondent: ", quote=FALSE)
print(imp)
print(paste("Sum: ", sum(imp)), quote=FALSE)

#Example 4
library(conjoint)
data(journey)
imp<-caImportance(jpref[1:5,],jprof)
print("Importance summary of group of 5 respondents: ", quote=FALSE)
print(imp)
print(paste("Sum: ", sum(imp)), quote=FALSE)
```

caLogit *Function caLogit estimates participation (market share) of the simulation profiles*

Description

Function caLogit estimates participation of simulation profiles using logit model. Function returns vector of percentage participations. The sum of participation should be 100%.

Usage

```
caLogit(sym, y, x)
```

Arguments

sym	matrix of simulation profiles
y	matrix of preferences
x	matrix of profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caBTL, caMaxUtility and ShowAllSimulations

Examples

```
#Example 1
library(conjoint)
data(tea)
simutil<-caLogit(tsimp,tpref,tprof)
print("Percentage participation of profiles: ", quote=FALSE)
print(simutil)

#Example 2
library(conjoint)
data(chocolate)
simutil<-caLogit(csimp,cpref,cprof)
print("Percentage participation of profiles:", quote=FALSE)
print(simutil)

#Example 3
library(conjoint)
data(chocolate)
ShowAllSimulations(csimp,cpref,cprof)

#Example 4
#library(conjoint)
#data(journey)
#ShowAllSimulations(jsimp,jpref,jprof)
```

`caMaxUtility` *Function caMaxUtility estimates participation (market share) of simulation profiles*

Description

Function caMaxUtility estimates participation of simulation profiles using model of maximum utility ("first position"). Function returns vector of percentage participations. The sum of participation should be 100%.

Usage

```
caMaxUtility(sym, y, x)
```

Arguments

`sym`	matrix of simulation profiles
`y`	matrix of preferences
`x`	matrix of profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caBTL, caLogit and ShowAllSimulations

Examples

```
#Example 1
library(conjoint)
data(tea)
simutil<-caMaxUtility(tsimp,tpref,tprof)
print("Percentage participation of profiles: ", quote=FALSE)
print(simutil)

#Example 2
library(conjoint)
data(chocolate)
simutil<-caMaxUtility(csimp,cpref,cprof)
print("Percentage participation of profiles:", quote=FALSE)
print(simutil)

#Example 3
library(conjoint)
data(chocolate)
ShowAllSimulations(csimp,cpref,cprof)

#Example 4
#library(conjoint)
#data(journey)
#ShowAllSimulations(jsimp,jpref,jprof)
```

caModel *Function caModel estimates parameters of conjoint analysis model*

Description

Function caModel estimates parameters of conjoint analysis model for one respondent. Function caModel returns vector of estimated parameters of traditional conjoint analysis model.

Usage

```
caModel(y, x)
```

Arguments

y	vector of preferences, vector should be like single profil of preferences
x	matrix of profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

Conjoint

Examples

```
#Example 1
library(conjoint)
data(tea)
model=caModel(tprefm[1,], tprof)
print(model)
```

```
#Example 2
library(conjoint)
data(chocolate)
model=caModel(cprefm[1,], cprof)
print(model)

#Example 3
library(conjoint)
data(journey)
model=caModel(jpref[306,], jprof)
print(model)
```

`caPartUtilities` *Function caPartUtilities calculates matrix of individual utilities*

Description

Function caPartUtilities calculates matrix of individual utilities for respondents. Function returns matrix of partial utilities (parameters of conjoint model regresion) for all artificial variables including parameters for reference levels for respondents (with intercept on first place).

Usage

```
caPartUtilities(y, x, z)
```

Arguments

`y`	matrix of preferences
`x`	matrix of profiles
`z`	vector of levels names

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caUtilities, caTotalUtilities and ShowAllUtilities

Examples

```
#Example 1
library(conjoint)
data(tea)
uslall<-caPartUtilities(tprefm,tprof,tlevn)
print(uslall)

#Example 2
library(conjoint)
data(chocolate)
uslall<-caPartUtilities(cprefm,cprof,clevn)
print(head(uslall))

#Example 3
library(conjoint)
data(journey)
usl<-caPartUtilities(jpref[1,],jprof,jlevn)
print("Individual (partial) utilities for first respondent:")
print(usl)
```

---

caRankToScore — *Function caRankToScore transforms ranking data into rating data design*

Description

Function caRankToScore transforms ranking data into rating data design necessary for conjoint model.

Usage

```
caRankToScore(y.rank)
```

Arguments

y.rank	matrix of preferences in ranking format

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

Examples

```
#Example 1
library(conjoint)
data(ice)
print(ilevn)
print(iprof)
print(ipref)
preferences<-caRankToScore(ipref)
print(preferences)
Conjoint(preferences, iprof, ilevn)
```

`caRecreatedDesign` *Function caRecreatedDesign reconstructs factorial design*

Description

Function caRecreatedDesign reconstructs the factorial design on the basis of arguments in the form of: a vector of variables (attributes) names, a vector of the number of variables' levels, a vector of variable level names and the list of numbers of the reconstructed profiles.

Usage

```
caRecreatedDesign(attr.names,lev.numbers,z,prof.numbers)
```

Arguments

attr.names	a vector of variables (attributes) names
lev.numbers	a vector of the number of variables' levels
z	a vector of variable level names
prof.numbers	list of numbers of the reconstructed profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caFactorialDesign and caEncodedDesign

Examples

```
#Example 1
library(conjoint)
attrNames<-c("price","variety","kind","aroma")
levNumbers<-c(3,3,3,2)
z<-c("low","medium","high","black","green","red","bags","granulated","leafy","yes","no")
profNumbers<-c(3,4,14,20,27,29,33,35,39,43,46,50,51)
design<-caRecreatedDesign(attrNames,levNumbers,z,profNumbers)
print(design)
print(design$dnumbers)
print(design$dnames)
```

caSegmentation *Function caSegmentation divides respondents on clusters*

Description

Function caSegmentation divides respondents on n clusters (segments) using *k*-means method (function kmeans, package stats). There are two data sets used - matrix or vector of preferences and matrix of profiles.

Usage

```
caSegmentation(y, x, c)
```

Arguments

y	matrix of preferences
x	matrix of profiles
c	number of clusters (optional), default value c=2

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

Examples

```
#Example 1
library(conjoint)
require(fpc)
data(tea)
segments<-caSegmentation(tprefm,tprof)
print(segments$seg)
plotcluster(segments$util,segments$sclu)
```

```
#Example 2
library(conjoint)
require(fpc)
data(tea)
segments<-caSegmentation(tpref,tprof,3)
print(segments$seg)
plotcluster(segments$util,segments$sclu)

#example 3
library(conjoint)
require(fpc)
require(broom)
require(ggplot2)
data(tea)
segments<-caSegmentation(tprefm,tprof,3)
dcf<-discrcoord(segments$util,segments$sclu)
assignments<-augment(segments$segm,dcf$proj[,1:2])
ggplot(assignments)+geom_point(aes(x=X1,y=X2,color= .cluster))+labs(color="Cluster Assignment",
title="K-Means Clustering Results")

#Example 4
library(conjoint)
require(ggfortify)
data(tea)
segments<-caSegmentation(tpref,tprof,3)
print(segments$seg)
util<-as.data.frame(segments$util)
set.seed(123)
ggplot2::autoplot(kmeans(util,3),data=util,label=TRUE,label.size=4,frame=TRUE)

#Example 5
#library(conjoint)
#require(ggfortify)
#require(cluster)
#data(tea)
#segments<-caSegmentation(tpref,tprof,3)
#print(segments$seg)
#util<-as.data.frame(segments$util)
#ggplot2::autoplot(pam(util,3),label=TRUE,label.size=4,frame=TRUE,frame.type='norm')
```

`caTotalUtilities` *Function caTotalUtilities calculates matrix of theoreticall total utilities*

Description

Function caTotalUtilities calculates matrix of theoreticall total utilities for respondents. Function returns matrix of total utilities for all profiles.

Usage

```
caTotalUtilities(y, x)
```

Arguments

y	matrix of preferences
x	matrix of profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caUtilities, caPartUtilities and ShowAllUtilities

Examples

```
#Example 1
library(conjoint)
data(tea)
uslall<-caTotalUtilities(tprefm,tprof)
print(uslall)

#Example 2
library(conjoint)
data(chocolate)
uslall<-caTotalUtilities(cprefm,cprof)
print(uslall)

#Example 3
library(conjoint)
data(journey)
usl<-caTotalUtilities(jpref[1,],jprof)
```

```
print("Individual (total) utilities for first respondent:")
print(usl)
```

`caUtilities` *Function caUtilities calculates utilities of levels of atrtributes*

Description

Function caUtilities calculates utilities of attribute's levels. Function returns vector of utilities.

Usage

```
caUtilities(y,x,z)
```

Arguments

`y`	matrix of preferences
`x`	matrix of profiles
`z`	matrix of levels names

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

`caPartUtilities` and `caTotalUtilities`

Examples

```
#Example 1
library(conjoint)
data(tea)
uslall<-caUtilities(tprefm,tprof,tlevn)
print(uslall)

#Example 2
library(conjoint)
data(chocolate)
uslall<-caUtilities(cprefm,cprof,clevn)
print(uslall)

#Example 3
library(conjoint)
data(journey)
usl<-caUtilities(jpref[1,],jprof,jlevn)
print("Individual utilities for first respondent:")
print(usl)
```

---

chocolate *Sample data for conjoint analysis*

Description

Sample data in score mode. Rating (score) data does not need any conversion. Data collected in the survey conducted by W. Nowak in 2000.

Usage

```
data(chocolate)
```

Format

`cpref` Vector of preferences (length 1392).

`cprefm` Matrix of preferences (87 respondents and 16 profiles).

`cprof` Matrix of profiles (5 attributes and 16 profiles).

`clevn` Character vector of names for the attributes' levels.

`csimp` Matrix of simulation profiles.

Examples

```
library(conjoint)
data(chocolate)
print(cprefm)
print(cprof)
print(clevn)
print(csimp)
```

`Conjoint` *Function Conjoint sums up the main results of conjoint analysis*

Description

Function Conjoint is a combination of following conjoint pakage's functions: `caPartUtilities`, `caUtilities` and `caImportance`. Therefore it sums up the main results of conjoint analysis. Function Conjoint returns matrix of partial utilities for levels of variables for respondents, vector of utilities for attribute's levels and vector of percentage attributes' importance with corresponding chart (barplot). The sum of importance should be 100

Usage

```
Conjoint(y, x, z, y.type)
```

Arguments

`y`	matrix of preferences
`x`	matrix of profiles
`z`	matrix of levels names
`y.type`	type of data preferences (possible values: "score" for preferences as rating data, "rank" for preferences as ranking data; default value: y.type="score")

Author(s)

Andrzej Bak `<andrzej.bak@ue.wroc.pl>`,

Tomasz Bartlomowicz `<tomasz.bartlomowicz@ue.wroc.pl>`

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

`caImportance`, `caPartUtilities` and `caUtilities`

Examples

```
#Example 1
library(conjoint)
data(ice)
print("Preferences of all respondents (preferences as ranking data):")
Conjoint(ipref,iprof,ilevn,y.type="rank")

#Example 2
library(conjoint)
data(ice)
ipref=caRankToScore(ipref)
print("Preferences of all respondents (preferences converted into rating data):")
Conjoint(ipref,iprof,ilevn,y.type="score")

#Example 3
library(conjoint)
data(journey)
print("Preferences of all respondents (preferences as default - rating data):")
Conjoint(jpref,jprof,jlevn)

#Example 4
library(conjoint)
data(tea)
print("Preferences of all respondents (preferences as rating data):")
Conjoint(tprefm,tprof,tlevn,y.type="score")

#Example 5
library(conjoint)
data(tea)
print("Preferences of first respondent (preferences as default - rating data):")
Conjoint(tprefm[1,],tprof,tlevn)

#Example 6
library(conjoint)
data(tea)
print("Preferences of group of 5 respondents (preferences as rating data):")
Conjoint(tprefm[11:15,],tprof,tlevn,y.type="score")
```

`czekolada` *Sample data for conjoint analysis*

Description

Sample data in score mode. Rating (score) data does not need any conversion. Data collected in the survey conducted by W. Nowak in 2000.

Usage

```
data(czekolada)
```

Format

`czpref` Vector of preferences (length 1392).

`czprefm` Matrix of preferences (87 respondents and 16 profiles).

`czprof` Matrix of profiles (5 attributes and 16 profiles).

`czlevn` Character vector of names for the attributes' levels.

`czsimp` Matrix of simulation profiles.

Examples

```
library(conjoint)
data(czekolada)
print(czprefm)
print(czprof)
print(czlevn)
print(czsimp)
```

---

herbata *Sample data for conjoint analysis*

Description

Sample data in score mode. Rating (score) data does not need any conversion. Data collected in the survey conducted by M. Baran in 2007.

Usage

```
data(herbata)
```

Format

`hpref` Vector of preferences (length 1300).

`hprefm` Matrix of preferences (100 respondents and 13 profiles).

`hprof` Matrix of profiles (4 attributes and 13 profiles).

`hlevn` Character vector of names for the attributes' levels.

`hsimp` Matrix of simulation profiles.

Examples

```
library(conjoint)
data(herbata)
print(hprefm)
print(hprof)
print(hlevn)
print(hsimp)
```

ice *Sample data for conjoint analysis*

Description

Sample artificial data in rank mode. Ranking (rank) data needs conversion into rating (score) data.

Usage

```
data(ice)
```

Format

`ipref` Matrix of preferences (6 respondents and 9 profiles).

`iprof` Matrix of profiles (4 attributes and 9 profiles).

`ilevn` Character vector of names for the attributes' levels.

Examples

```
library(conjoint)
data(ice)
print(ipref)
print(iprof)
print(ilevn)
```

journey *Sample data for conjoint analysis*

Description

Sample data in score mode. Rating (score) data does not need any conversion. Data collected in the survey conducted by M. Gordzicz in 2015/2016.

Usage

```
data(journey)
```

Format

`jpref` Matrix of preferences (306 respondents and 14 profiles).

`jprof` Matrix of profiles (4 attributes and 14 profiles).

`jlevn` Character vector of names for the attributes' levels.

`jsimp` Matrix of simulation profiles.

Examples

```
library(conjoint)
data(journey)
print(jpref)
print(jprof)
print(jlevn)
print(jsimp)
```

---

lody *Sample data for conjoint analysis*

Description

Sample artificial data in rank mode. Ranking (rank) data needs conversion into rating (score) data.

Usage

```
data(lody)
```

Format

`lpref` Matrix of preferences (6 respondents and 9 profiles).

`lprof` Matrix of profiles (4 attributes and 9 profiles).

`llevn` Character vector of names for the attributes' levels.

Examples

```
library(conjoint)
data(lody)
print(lpref)
print(lprof)
print(llevn)
```

---

plyty *Sample data for conjoint analysis*

Description

Sample artificial data in score mode. Rating (score) data does not need any conversion.

Usage

```
data(plyty)
```

Format

`ppref` Matrix of preferences (6 respondents and 8 profiles).

`pprof` Matrix of profiles (3 attributes and 8 profiles).

`plevn` Character vector of names for the attributes' levels.

Examples

```
library(conjoint)
data(plyty)
print(ppref)
print(pprof)
print(plevn)
```

ShowAllSimulations *Function ShowAllSimulations sums up the results of all simulation functions*

Description

Function ShowAllSimulations sums up the results of all simulation functions. It's a combination of following conjoint pakage's functions: `caMaxUtility`, `caBTL` and `caLogit`. Therefore it sums up the main results of simulation using conjoint analysis method. Function ShowAllSimulations returns three vectors of percentage participations using maximum utility, BTL and logit models. The sum of importance for every vector should be 100%.

Usage

```
ShowAllSimulations(sym, y, x)
```

Arguments

`sym`	matrix of simulation profiles
`y`	matrix of preferences
`x`	matrix of profiles

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A., Bartlomowicz T. (2012), *Conjoint analysis method and its implementation in conjoint R package*, [In:] Pociecha J., Decker R. (Eds.), *Data analysis methods and its applications*, C.H.Beck, Warszawa, p.239-248.

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa, p. 283-317.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, p. 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caBTL, caLogit and caMaxUtility

Examples

```
#Example 1
library(conjoint)
data(tea)
ShowAllSimulations(tsimp,tpref,tprof)

#Example 2
library(conjoint)
data(chocolate)
ShowAllSimulations(csimp,cpref,cprof)

#Example 3
#library(conjoint)
#data(journey)
#ShowAllSimulations(jsimp,jpref,jprof)
```

ShowAllUtilities *Function ShowAllUtilities sums up all results of utility measures*

Description

Function ShowAllUtilities is a combination of following conjoint pakage's functions: caPartUtilities, caTotalUtilities, caUtilities and caImportance. Function ShowAllUtilities returns: matrix of partial utilities (basic matrix of utilities with the intercept), matrix of total utilities for n profiles and all respondents, vector of utilities for attribute's levels and vector of percentage attributes' importance, with sum of importance. The sum of importance should be 100%.

Usage

```
ShowAllUtilities(y, x, z)
```

Arguments

y	matrix of preferences
x	matrix of profiles
z	matrix of levles names

Author(s)

Andrzej Bak <andrzej.bak@ue.wroc.pl>,

Tomasz Bartlomowicz <tomasz.bartlomowicz@ue.wroc.pl>

Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland

References

Bak A. (2009), *Analiza Conjoint [Conjoint Analysis]*, [In:] Walesiak M., Gatnar E. (Eds.), *Statystyczna analiza danych z wykorzystaniem programu R [Statistical Data Analysis using R]*, Wydawnictwo Naukowe PWN, Warszawa.

Green P.E., Srinivasan V. (1978), *Conjoint Analysis in Consumer Research: Issues and Outlook*, "Journal of Consumer Research", September, 5, 103-123.

SPSS 6.1 Categories (1994), SPSS Inc., Chicago.

See Also

caImportance, caPartUtilities, caTotalUtilities and caUtilities

Examples

```
#Example 1
library(conjoint)
data(tea)
ShowAllUtilities(tprefm,tprof,tlevn)

#Example 2
library(conjoint)
data(chocolate)
ShowAllUtilities(cprefm,cprof,clevn)
```

---

tea *Sample data for conjoint analysis*

Description

Sample data in score mode. Rating (score) data does not need any conversion. Data collected in the survey conducted by M. Baran in 2007.

Usage

```
data(tea)
```

Format

`tpref` Vector of preferences (length 1300).

`tprefm` Matrix of preferences (100 respondents and 13 profiles).

`tprof` Matrix of profiles (4 attributes and 13 profiles).

`tlevn` Character vector of names for the attributes' levels.

`tsimp` Matrix of simulation profiles.

Examples

```
library(conjoint)
data(tea)
print(tprefm)
print(tprof)
print(tlevn)
print(tsimp)
```

---

wycieczka *Sample data for conjoint analysis*

Description

Sample data in score mode. Rating (score) data does not need any conversion. Data collected in the survey conducted by M. Gordzicz in 2015/2016.

Usage

```
data(wycieczka)
```

Format

`wpref` Matrix of preferences (306 respondents and 14 profiles).

`wprof` Matrix of profiles (4 attributes and 14 profiles).

`wlevn` Character vector of names for the attributes' levels.

`wsimp` Matrix of simulation profiles.

Examples

```
library(conjoint)
data(wycieczka)
print(wpref)
print(wprof)
print(wlevn)
print(wsimp)
```

Index