Package ‘lba’

January 10, 2018

Title Latent Budget Analysis for Compositional Data
Version 2.4.4
Date 2018-01-02
Author Enio G. Jelihovschi <eniojelihovs@gmail.com> and
Ivan Bezerra Allaman <ivanalaman@gmail.com>
Maintainer Enio G. Jelihovschi <eniojelihovs@gmail.com>
Depends R (>= 3.1.0), MASS, alabama, plotrix, scatterplot3d, rgl
Description Latent budget analysis is a method for the analysis of a two-way
contingency table with an exploratory variable and a response variable. It is
specially designed for compositional data.
Encoding latin1
License GPL (>= 2)
URL https://github.com/ivanalaman/lba
NeedsCompilation no
Repository CRAN
Date/Publication 2018-01-10 18:40:30 UTC

R topics documented:

   goodnessfit .................................................. 2
   housing ...................................................... 6
   Latent Budget Analysis ..................................... 7
   lba ............................................................ 7
   MANHATAN .................................................... 16
   PerfMark ...................................................... 17
   plotcorr ...................................................... 19
   plotlba ...................................................... 24
   postmater ..................................................... 28
   pregnancy ..................................................... 29
   print.goodnessfit .......................................... 30
   print.lba ..................................................... 31
Goodness of Fit results for Latent Budget Analysis

Description

The goodness of fit results assesses how well the model fits the data. It consists of measures of the resemblance between the observed and the expected data, and the parsimony of the model.

Usage

goodnessfit(object,...)

## S3 methods
## Default S3 method:
goodnessfit(object, ...)

## S3 method for class 'lba.ls'
goodnessfit(object, ...)

## S3 method for class 'lba.ls.fe'
goodnessfit(object, ...)

## S3 method for class 'lba.ls.logit'
goodnessfit(object, ...)

## S3 method for class 'lba.mle'
goodnessfit(object, ...)

## S3 method for class 'lba.mle.fe'
goodnessfit(object, ...)

## S3 method for class 'lba.mle.logit'
goodnessfit(object, ...)

Arguments

object An object of one of following classes: lba.ls, lba.ls.fe, lba.ls.logit, lba.mle, lba.mle.fe, lba.mle.logit

... Further arguments (required by generic).
**Value**

The `goodnessfit` function of the method `lba.mle`, `lba.mle.fe` and `lba.mle.logit` returns a list with the slots:

- `dfdb` Degrees of freedom of the base model
- `dfd` Degrees of freedom of the full model
- `G2b` Likelihood ratio statistic of the base model
- `G2` Likelihood ratio statistic of the full model
- `chi2b` Chi-square statistic of the base model
- `chi2` Chi-square statistic of the full model
- `proG1` P-value of likelihood ratio statistic of the base model
- `proG` P-value of likelihood ratio statistic of the full model
- `prochi1` P-value of chi-square statistic of the base model
- `prochi` P-value of chi-square statistic of the full model
- `AICb` AIC criteria of the base model
- `AICC` AIC criteria of the full model
- `BICb` BIC criteria of the base model
- `BICC` BIC criteria of the full model
- `CAICb` CAIC criteria of the base model
- `CAIC` CAIC criteria of the full model
- `delta1` Normed fit index
- `delta2` Normed fit index modified
- `rho1` Bollen index
- `rho2` Tucker-Lewis index
- `RSS1` Residual sum of square of the base model
- `RSS` Residual sum of square of the full model
- `impRSS` Improvement of RSS
- `impPB` Improvement per budget
- `impDF` Average improvement per degree of freedom
- `D1` Index of dissimilarity of the base model
- `D` Index of dissimilarity of the full model
- `pccb` Proportion of correctly classified data of the base model
- `pcc` Proportion of correctly classified data of the full model
- `impD` Improvement of proportion of correctly classified data
- `impPCCB` Improvement of Proportion of correctly classified data per budget
- `AimpPCCDF` Average improvement of Proportion of correctly classified data per degree of freedom
- `mad1` Mean angular deviation of the base model
madk Mean angular deviation of the full model
impMad Improvement mean angular deviation
impPBSat Improvement mean angular deviation per budget
impDFsat Average improvement mean angular deviation per degree of freedom

The goodnessfit function of the method lba.ls, lba.ls.fe and lba.ls.logit returns a list with the slots:
dfdb Degrees of freedom of the base model
dfd Degrees of freedom of the full model
RSS1 Residual sum of square of the base model
RSS Residual sum of square of the full model
impRSS Improvement of RSS
impPB Improvement per budget
impDF Average improvement per degree of freedom
D1 Index of dissimilarity of the base model
D Index of dissimilarity of the full model
pccb Proportion of correctly classified data of the base model
pcc Proportion of correctly classified data of the full model
impD Improvement of proportion of correctly classified data
impPCCB Improvement of Proportion of correctly classified data per budget
AimpPCCDF Average improvement of Proportion of correctly classified data per degree of freedom
mad1 Mean angular deviation of the base model
madk Mean angular deviation of the full model
impMad Improvement mean angular deviation
impPBSat Improvement mean angular deviation per budget
impDFsat Average improvement mean angular deviation per degree of freedom

Note
For a detailed and complete discussion about goodness of fit results for latent budget analysis, see van der Ark 1999.

References

See Also
summary.goodnessfit.lba.ls, summary.goodnessfit.lba.mle, lba
Examples

data('votB')

# Using LS method (default) without constraint
# K = 2
ex1 <- lba(parties ~ city,
         votB,
         K = 2)

gx1 <- goodnessfit(ex1)
gx1

# Using MLE method without constraint
# K = 2
exm <- lba(parties ~ city,
          votB,
          K = 2,
          method='mle')

gxm <- goodnessfit(exm)
gxm

# Using LS method (default) with LOGIT constrain
data('housing')

# Make cross-table to matrix design.
tbh <- xtabs(value ~ Influence + Housing, housing)
Xis <- model.matrix(~ Housing*Influence,
                    tbh,
                    contrasts=list(Housing='contr.sum',
                                    Influence='contr.sum'))

tby <- xtabs(value ~ Satisfaction + Contact, housing)
Yis <- model.matrix(~ Satisfaction*Contact,
                    tby,
                    contrasts=list(Satisfaction='contr.sum',
                                    Contact='contr.sum'))

S <- 12
T <- 5

tabs <- xtabs(value ~ interaction(Housing,
                                  Influence) + interaction(Satisfaction,
                                  Contact),
                                  housing)

# Not run:
ex2 <- lba(tabs,
          K = 2,
          logitA = Xis,
          logitB = Yis,
The Satisfaction with Housing Conditions Study

Description

The housing data frame has 72 rows and 5 columns. The observations were obtained from an investigation of Satisfaction with housing conditions carried out by the Danish Building Research Institute and the Danish Institute of Mental Health Research.

Usage

c data frame containing the following columns:

- Housing: A factor with levels: Apartment; Atrium; Terraced; Tower.
- Influence: A factor with levels: hi; low; med.
- Contact: A factor with levels: high; low.
- Satisfaction: A factor with levels: hi; low; med.
- value: The absolute frequencies of which factor.

Source


References

Latent Budget Analysis

Latent Budget Analysis (LBA) for Compositional Data

Description

Latent budget analysis (LBA) is a method for the analysis of contingency tables, from where the compositional data is derived. It is used to understand the relationship between the table rows and columns, where the rows denote the categories of the explanatory variable and the columns denote the categories of the response variable.

Details

The row vectors of the compositional data are called observed budgets which are approximated by the expected budgets. The LBA allows us to find which categories of the response are related to different groups of the explanatory categories. If the table has a product multinomial distribution we can understand the latent budget model (LBM) as explaining the relationship between the explanatory and the response variables assuming that conditioned on the latent variable they are independent. In that sense, the latent budgets, which are categories of a latent variable, are hidden values which explain the relationship between the explanatory and response variables. LBA reduce the dimensionality of the original problem, thus making it easier to understand its hidden relations.

Author(s)

Enio G. Jelihovschi <eniojelihovs@gmail.com> and Ivan Bezerra Allaman <ivanalaman@gmail.com>. We are grateful to José C. Faria, Andreas L. van der Ark and Fábio M. Correa for suggestions.

Maintainer: Enio G. Jelihovschi <eniojelihovs@gmail.com>

Usage

lba(obj, ...)

## S3 method for class 'matrix'
lba(obj, 
    A = NULL, 
    ...,

Description

Latent budget analysis (LBA) is a method for the analysis of contingency tables, from where the compositional data is derived. It is used to understand the relationship between the table rows and columns, where the rows denote the categories of the explanatory variable and the columns denote the categories of the response variable.

Usage

lba(obj, ...)

## S3 method for class 'matrix'
lba(obj, 
    A = NULL, 
    ...,
B = NULL,
K = 1L,
cA = NULL,
cB = NULL,
logitA = NULL,
logitB = NULL,
omsk = NULL,
psitk = NULL,
S = NULL,
T = NULL,
row.weights = NULL,
col.weights = NULL,
tolG = 1e-10,
tolA = 1e-05,
tolB = 1e-05,
itmax.unide = 1e3,
itmax.ide = 1e3,
trace.lba = TRUE,
toltype = "all",
method = c("ls", "mle"),
what = c("inner","outer"), ...)

## S3 method for class 'table'
lba(obj,
  A = NULL,
  B = NULL,
  K = 1L,
  cA = NULL,
  cB = NULL,
  logitA = NULL,
  logitB = NULL,
  omsk = NULL,
  psitk = NULL,
  S = NULL,
  T = NULL,
  row.weights = NULL,
  col.weights = NULL,
  tolG = 1e-10,
  tolA = 1e-05,
  tolB = 1e-05,
  itmax.unide = 1e3,
  itmax.ide = 1e3,
  trace.lba = TRUE,
  toltype = "all",
  method = c("ls", "mle"),
  what = c("inner","outer"), ...)

## S3 method for class 'formula'
lba(formula, data,
   A = NULL,
   B = NULL,
   K = 1L,
   cA = NULL,
   cB = NULL,
   logitA = NULL,
   logitB = NULL,
   omsk = NULL,
   psitk = NULL,
   S = NULL,
   T = NULL,
   row.weights = NULL,
   col.weights = NULL,
   tolG = 1e-10,
   tolA = 1e-05,
   tolB = 1e-05,
   itmax.unide = 1e3,
   itmax.ide = 1e3,
   trace.lba = TRUE,
   toltype = "all",
   method = c("ls", "mle"),
   what = c("inner","outer"), ...)

## S3 method for class 'ls'
lba(obj,
   A ,
   B ,
   K ,
   row.weights ,
   col.weights ,
   tolA ,
   tolB ,
   itmax.unide ,
   itmax.ide ,
   trace.lba ,
   what , ...)

## S3 method for class 'mle'
lba(obj,
   A ,
   B ,
   K ,
   tolG ,
   tolA ,
   tolB ,
   itmax.unide ,
   itmax.ide ,
trace.lba ,
toltype ,
what , ...

## S3 method for class 'ls.fe'

```r
lba(obj, 
A ,
B ,
K ,
cA ,
cB ,
row.weights ,
col.weights ,
itmax.ide ,
trace.lba , ...
)
```

## S3 method for class 'mle.fe'

```r
lba(obj, 
A ,
B ,
K ,
cA ,
cB ,
tolG ,
tolA ,
tolB ,
itmax.ide ,
trace.lba ,
toltype , ...
)
```

## S3 method for class 'ls.logit'

```r
lba(obj, 
A ,
B ,
K ,
cA ,
cB ,
logitA ,
logitB ,
omsk ,
psitk ,
S ,
T ,
row.weights ,
col.weights ,
itmax.ide ,
trace.lba , ...
)
```
## Arguments

- **obj, formula**
  The function is generic, accepting some forms of the principal argument for specifying a two-way frequency table. Currently accepted forms are matrix, data frame (coerced to frequency tables), objects of class "xtabs" or "table" and one-sided formulae of the form `Col1 + Col2 + ... + Coln ~ Row1 + Row2 + ... + Rown`, where `Rown` and `Coln` are nth row (the mixing parameters) and column variable (the latent components).

- **data**
  A data frame containing variables in `formula`.

- **A**
  The starting value of a (I x K) matrix containing the mixing parameters, if given. The default is `NULL`, producing random starting values.

- **B**
  The starting value of a (J x K) matrix containing the latent components, if given. The default is `NULL`, producing random starting values.

- **K**
  Integer giving the number of latent budgets chosen by the user. The default is 1.

- **cA**
  The value of a (I x K) matrix containing the constraints on the mixing parameters. Fixed constraints are the values themselves which are numbers in the [0,1] interval. The optional equality constraints are indicated by an integer starting from 2, such that parameters that must be equal have the same integer. The default is `NULL`, indicating no constraints.

- **cB**
  The value of a (J x K) matrix containing the constraints on the latent components. Fixed constraints are the values themselves which are numbers in the [0,1] interval. The optional equality constraints are indicated by an integer starting from 2, such that parameters that must be equal have the same integer. The default is `NULL`, indicating no constraints.

- **logitA**
  Design (IxS) matrix for row-covariates. The first column contains 1's, indicating a constant covariate. The entries may be continuous or dummy coded values.

- **logitB**
  Design (JxT) matrix for column-covariates. The entries may be continuous or dummy coded values.

- **omsk**
  A (SxK) matrix giving the starting values for the multinomial logit parameters of the row covariates. The default is `NULL`, producing random starting values.
psitk  A (TxK) matrix giving the starting values for the multinomial logit parameters of the column covariates. The default is NULL, producing random starting values.

S  Number of row-covariates. The default is NULL.

T  Number of column-covariates. The default is NULL.

row.weights  A vector with the same number of rows of the matrix of the weighted least squares method. If is NULL (default), the weights are 
\[ \sqrt{\frac{n_{i+}}{n_{++}}} \]

col.weights  A vector with the same number of columns of the matrix of the weighted least squares method. If is NULL (default), the weights are 
\[ \frac{1}{\sqrt{\frac{n_{+j}}{n_{++}}}} \]

tolG  A tolerance value for judging when convergence has been reached. It is based on the estimated likelihood ratio statistics G2. The default is 1e-10.

tolA  A tolerance value for judging when convergence has been reached. When the one-iteration change in the maximum of the absolute value of the element wise difference of the estimated matrices A is less than tolA. The default is 1e-05.

tolB  A tolerance value for judging when convergence has been reached. When the one-iteration change in the maximum of the absolute value of the element wise difference of the estimated matrices B is less than tolB. The default is 1e-05.

itmax.unide  Maximum number of iterations performed by the mle or ls method, if convergence is not achieved, before identification parameters. The default is 1e3.

itmax.ide  Maximum number of iterations performed by the mle or ls method in the identification process. Is used too when the constrained fixed, equality and logit are required. The default is 1e3.

trace.lba  Logical, indicating whether the base function optim and constrOptim.nl from package alabama, will trace their results. The default is TRUE.

toltype  String indicating which kind of tolerance to be used. That is, the EM algorithm stops updating and considers the maximum log-likelihood to have been found. Their types are: "all" when the one-iteration change in the estimated likelihood ratio statistics G2 is less than tolG, and the one-iteration change in the maximum of the absolute value of the element wise difference of the estimated matrices A is less than tolA and the same for estimated matrices B with respect to tolB; "G2" when the only one-iteration change in the estimated likelihood ratio statistics G2 is less than tolG; "ab" when only the one-iteration change in the maximum of the absolute value of the element wise difference of the estimated matrices A is less than tolA and the same for estimated matrices B with respect to tolB. tolttype works only for method = "mle". The default is "all". The ls method uses only "ab" as tolerance limit.

method  String indicating which kind of estimating method. They are: "ls" when least squares, either weighted or ordinary, method is used; "mle" when maximum likelihood method is used. The default is "ls".
what String indicating which kind identified solutions for mixing parameters and latent budgets matrices. They are: the "inner" extreme solution and the "outer" extreme solution. The default is "inner".

... Further arguments (required by generic).

Value

The method lba.ls and lba.mle returns a list of class lba.ls and lba.mle respectively with the slots:

P The compositional data matrix which is formed by dividing the raw data matrix by their corresponding total, its rows are called observed budgets.
pij Matrix whose rows are the expected budgets.
residual Residual matrix P - pij.
A (I x K) matrix of the unidentified the mixing parameters.
B (J x K) matrix of the unidentified the latent components.
Aoi (I x K) matrix of the identified mixing parameters, they may be either the inner extreme values or the outer extreme values.
Boi (J x K) matrix of the identified latent components, they may be either the inner extreme values or the outer extreme values.
rescb (J x K) matrix of the rescaled latent components.
pk Budget proportions.
val_func Value of least squared or likelihood function achieved.
iter_unide Number of unidentified iterations.
iter_ide Number of identified iterations.

The method lba.ls.fe and lba.mle.fe returns a list of class lba.ls.fe and lba.mle.fe respectively with the slots:

P The compositional data matrix which is formed by dividing the raw data matrix by their corresponding row total, its rows are called observed budgets.
pij Matrix whose rows are the expected budgets.
residual Residual matrix P - pij.
A (I x K) matrix of the unidentified the mixing parameters.
B (J x K) matrix of the unidentified the latent components.
rescb (J x K) matrix of the rescaled latent components.
pk Budget proportions.
val_func Value of least squared or likelihood function achieved.
iter_ide Number of identified iterations.

The method lba.ls.logit and lba.mle.logit returns a list of class lba.ls.logit and lba.mle.logit respectively with the slots:
The compositional data matrix which is formed by dividing the raw data matrix by their corresponding total, its rows are called observed budgets.

\( \text{p} \)  
Matrix whose rows are the expected budgets.

\( \text{residual} \)  
Residual matrix \( \text{p} - \text{pij} \).

\( \text{A} \)  
(\( I \times K \)) matrix of the unidentified the mixing parameters.

\( \text{B} \)  
(\( J \times K \)) matrix of the unidentified the latent componentes.

\( \text{rescB} \)  
(\( J \times K \)) matrix of the rescaled latent components.

\( \text{pk} \)  
Budget proportions.

\( \text{val_func} \)  
Value of least squared or likelihood function achieved.

\( \text{iter_ide} \)  
Number of identified iterations.

\( \text{omsk} \)  
A (\( S \times K \)) matrix giving estimated values of the multinomial logit parameters of the row covariates.

\( \text{psitk} \)  
A (\( T \times K \)) matrix giving the estimated values for the multinomial logit parameters of the column covariates.

**Note**

The user has two options to entry the data: the raw data and the tabulated data. If the raw data is imported, he may indicate which, among the variables, comprises the row and which the column variable and let the \text{lba} function make the tabulation. The user may also tabulate the data with the available functions in R. Recalling that if this second option is used, the object must be of the class \text{xtabs}, \text{table} or \text{matrix}. If the user imports the tabulated data, the class is, in general, \text{data.frame} and so, it is necessary to transform the object data into a \text{matrix}.

The function \text{lba} uses EM algorithm to maximise the latent budget model log-likelihood function; the Active Constraints Methods (ACM) to minimise either the weighted least squares (wls), or ordinary least squares (ols) functions; and "BFGS" variable metric method in \text{constrOptim.nl} function of \text{alabama} package and in \text{optim} function of \text{stats} package used in identification for \( K \geq 3 \), in constraint algorithm for \text{ls} method, in multinomial logit constraints and in some parts of constraining for mle method. Depending on the starting parameters, those algorithms may only locate a local, rather than global, maximum. This becomes more and more of a problem as \( K \), the number of latent budgets, increases. It is therefore highly advisable to run \text{lba} multiple times until you are relatively certain that you have located the global maximum log-likelihood or the global minimum least squares.

**References**


See Also

`goodnessfit`, `summary.lba.ls`, `summary.lba.mle`, `plotlba`, `plotcorr`

Examples

data('votB')

# Using LS method (default) without constraint
# K = 2
ex1 <- lba(parties ~ city,  
           votB,  
           K = 2)
ex1

# Already tabulated data? Ok!
data('PerfMark')

## Not run:
ex2 <- lba(as.matrix(PerfMark),  
           K = 2,  
           what='outer')
ex2

## End(Not run)
# Using LS method (default) with constraint
# Fixed constraint to mixing parameters
cakif1 <- matrix(c(0.2, NA, NA,  
                  NA, NA, 0.2,  
                  NA, NA, 0.2,  
                  0.3, NA, NA,  
                  0.2, NA, NA,  
                  NA, NA, NA),  
                  byrow = TRUE,  
                  ncol = 3)

# K = 3
## Not run:
exf1 <- lba(parties ~ city,  
             votB,  
             cA = cakif1,  
             K = 3)
exf1

## End(Not run)
# Using LS method (default) with LOGIT constrain
data('housing')

# Make cross-table to matrix design.
tbh <- xtabs(value ~ Influence + Housing, housing)

Xis <- model.matrix(~ Housing*Influence,
               tbh,
               contrasts=list(Housing='contr.sum',
               Influence='contr.sum'))

tby <- xtabs(value ~ Satisfaction + Contact, housing)

Yis <- model.matrix(~ Satisfaction*Contact,
               tby,
               contrasts=list(Satisfaction='contr.sum',
               Contact='contr.sum'))[,,-1]

S <- 12
T <- 5

tabs <- xtabs(value ~ interaction(Housing, Influence) + interaction(Satisfaction, Contact),
               housing)

## Not run:
exlogit2 <- lba(tabs,
               K = 2,
               logitA = Xis,
               logitB = Yis,
               S = S,
               T = T,
               trace.lba=FALSE)
exlogit2

## End(Not run)

---

**MANHATAN**

*The Midtown Manhattan Study*

---

**Description**

The **MANHATAN** data frame has 25 rows and 3 columns. The observations were obtained in a study carried out by the sociologist Leo Srole and describe the cross-classification of 1660 adults in Manhattan, ages 20-59, obtained from a sample of midtown residents.

**Usage**

MANHATAN
**Description**

The `PerfMark` data frame has 31 rows and 46 columns. The data set is the result of a survey of 47 beauty salons located at the city of Lavras, Brazil, consisting of two types of questions; the first identifies the profile of the owner manager (explanatory variable), the second are questions referring to the degree of professionalism with respect to planning, market and finances (response variable). The data set is already cross-tabulated.

**Usage**

`PerfMark`

**Format**

This data frame contains the following columns referring the absolute frequencies to each row variable:

- **Planning variables:**
  - **PA14** What is the dependence of the owner to function properly?.
  - **PA20** What are your plans towards next year? only a dream.
  - **PA21** What are your plans towards next year? vague goals. Marketing variables:
  - **MA11** Your business tries to systematically assess the customer satisfaction and use that as a basis for management decisions. Alternative 1.
  - **MA12** Your business tries to systematically assess the customer satisfaction and use that as a basis for management decisions. Alternative 2.
  - **MA20** Your business offers more than the usual services. Alternative 0.
MA21 Your business offers more than the usual services. Alternative 1.
MA30 Your business is focused to further customer loyalty. Alternative 0.
MA31 Your business is focused to further customer loyalty. Alternative 1.
MA32 Your business is focused to further customer loyalty. Alternative 2.
MA42 What is the proportion, among current customers, of those who are customers for more than 6 months. Alternative 2.
MA43 What is the proportion, among current customers, of those who are customers for more than 6 months. Alternative 3.
MB12 Your business offers more services than when it began. Alternative 2.
MB22 How is your business quality perceived as compared to the competition? Alternative 2.
MB23 How is your business quality perceived as compared to the competition? Alternative 3.
MB31 How is your business range of services perceived as compared to the competition? Alternative 1.
MB32 How is your business range of services perceived as compared to the competition? Alternative 2.
MC11 What is your business level of prices perceived as compared to the competition? Alternative 1.
MC12 What is your business level of prices perceived as compared to the competition? Alternative 2.
MD13 Your business location is perceived as appropriate to the target market. Alternative 3.
ME10 Your business uses formal media to advertise itself. Alternative 0.
ME11 Your business uses formal media to advertise itself. Alternative 1.
ME25 Your business uses formal media to advertise itself. Alternative 5. Financial variables:
F10 Your business clearly separates the owner bills from the business bills. Alternative 0.
F14 Your business clearly separates the owner bills from the business bills. Alternative 4.
F20 Your owners withdrawal are planned and controlled in advance. Alternative 0.
F21 Your owners withdrawal are planned and controlled in advance. Alternative 1.
F24 Your owners withdrawal are planned and controlled in advance. Alternative 4.
F31 Your business pays for its purchases in installments. Alternative 1.
F34 Your business pays for its purchases in installments. Alternative 4.
F42 Your business knows today whether it will be able to pay its short-term bills of 60 days. Alternative 2.
F44 Your business knows today whether it will be able to pay its short-term bills of 60 days. Alternative 4.
F50 Your business uses short-term cash-flow analysis to plan for its short-term bills. Alternative 0.
F63 Your business has formal control of the monthly amount it makes from its services. Alternative 3.
F64 Your business has formal control of the monthly amount it makes from its services. Alternative 4.

F70 Your business uses either credit card, checkbook payment or loans, to finance its needs for working capital. Alternative 0.

F74 Your business uses either credit card, checkbook payment or loans, to finance its needs for working capital. Alternative 4.

F80 Your business uses specific credit to finance its needs for capital. Alternative 0.

F91 The company demonstrates knowledge to properly assess the costs of products used in services and costs of renting and taxes. Alternative 1.

F93 The company demonstrates knowledge to properly assess the costs of products used in services and costs of renting and taxes. Alternative 3.

F100 Your business clearly identifies the need for working capital. Alternative 0.

F111 Your business lays down the price of services in a systematic way. Alternative 1.

F113 Your business lays down the price of services in a systematic way. Alternative 3.

F120 The company calculates the interest on contracted loans. Alternative 0.

F125 The company calculates the interest on contracted loans. Alternative 5.

Source


References


---

### plotcorr

Plot `lba` objects using the correspondence analysis approach as suggested by Jelihovschi (2011).

#### Description

S3 methods for `lba` objects.

#### Usage

```r
plotcorr(x, 

## S3 method for class 'lba.1d'
plotcorr(x,
    xlim = NULL,
    ylim = NULL,
)```
### S3 method for class 'lba.2d'

```r
plotcorr(x,
  dim = c(1,2), #only K = 3
  xlim = NULL,
  ylim = NULL,
  xlab = NULL,
  ylab = NULL,
  args.legend = NULL,
  col.points = NULL,
  labels.points = NULL,
  pch.points = NULL,
  pos.points = NULL,
  labels.budget = NULL,
  pch.budget = NULL,
  pos.budget = NULL,
  cex.budget = NULL,
  col.budget = NULL,
  with.ml = c("mix","lat"),
  ...
)
```

### S3 method for class 'lba.3d'

```r
plotcorr(x,
  rgl.use = FALSE,
  dim = c(1,2,3), #only K >= 3
  xlim = NULL,
  ylim = NULL,
  zlim = NULL,
  xlab = NULL,
  ylab = NULL,
  ...
)
```
Arguments

\( x \)  
A object of lba class.

\( \text{dim} \)  
The dimension to be plotted. The default is \( c(1, 2) \) to \( K = 2 \) and \( c(1, 2, 3) \) to \( K = 3 \).

\( \text{xlim} \)  
The x limits (x1, x2) of the plot.

\( \text{ylim} \)  
The y limits of the plot.

\( \text{zlim} \)  
The z limits of the plot.

\( \text{xlab} \)  
A label for the x axis, defaults to a description of "x".

\( \text{ylab} \)  
A label for the y axis, defaults to a description of "y".

\( \text{zlab} \)  
A label for the z axis, defaults to a description of "z".

\( \text{rgl.use} \)  
A logical value. If TRUE the 3d scatter will be done with the rgl environment, in another way the scatterplot3d will be used.

\( \text{metrics} \)  
Logical. If TRUE (default), the radius is plotted.

\( \text{radius} \)  
A arbitrary number to choose the groups. The default is 0.5. See details.

\( \text{col.points} \)  
The color points to be used, possibly vectors. The default is NULL. See datails.

\( \text{height.points} \)  
Points label height in relation to the y-coordinate. The default is NULL.

\( \text{labels.points} \)  
A character vector or expression specifying the _text_ to be written. The default is NULL.

\( \text{pch.points} \)  
A symbols to use. O default is NULL.

\( \text{pos.points} \)  
A position specifier for the text. If specified this overrides any "adj" value given. Values of "1", "2", "3" and "4", respectively indicate positions below, to the left of, above and to the right of the specified coordinates.

\( \text{args.legend} \)  
List of additional arguments to be passed to legend; names of the list are used as argument names. Only used if \( K=2 \). The default is NULL.

\( \text{pch.budget} \)  
A symbols to use. O default is NULL.

\( \text{pos.budget} \)  
A position specifier for the text. If specified this overrides any "adj" value given. Values of "1", "2", "3" and "4", respectively indicate positions below, to the left of, above and to the right of the specified coordinates.
height.budget  Budget label height in relation to the y-coordinate. The default is NULL.
labels.budget  A character vector or expression specifying the _text_ to be written. The default is NULL.
cex.budget     The size of text. The default is NULL.
col.budget     The color budget to be used, possibly vectors. The default is NULL.
with.ml        Vector of two character strings specifying the parameters of the plot. Set "mix" to plot the mixing parameters and "lat" to plot the latent components. The default is "mix".
...             Further graphical parameters.

Details

The `plotcorr` suggested by Jelihovschi et al (2011), has a graphical display which uses the correspondence analysis graphics of the mixing parameters and latent components matrices. In this case, a graphic display is possible for $k \geq 2$.

The argument `radius` was featured in order to help the user as he or she needs do decide which are the points belonging to a certain latent budget. Only the points to the right or left of LB1 and LB2 but always towards the center of the graphic (the zero of x axis) were taken in account, since those in opposite direction automatically belong to the closest latent budget. This argument only works for $K = 2$. It’s should be of size two.

The argument `col.points` takes in account the argument `radius` in order to color the groups which either belong or not to a certain budget, therefore, the size of the vector of this argument must be equal to the number of formed groups.

Author(s)

Enio G. Jelihovschi (<eniojelihovs@gmail.com>)
Ivan Bezerra Allaman (<ivanalaman@gmail.com>)

References


See Also

`plot.default`, `scatterplot3d`, `plot3d`. 
Examples

```r
data('votB')
K = 2
ex1 <- lba(parties ~ city,
    data=votB,
    K = 2)
plotcorr(ex1)

#It's very simple. with colors!
plotcorr(ex1,
    col.points = 3:5,
    col.budget = c(5,3))

#Changing radius!
plotcorr(ex1,
    radius = rep(0.7,2))

#Without metrics!
plotcorr(ex1,
    metrics = FALSE)

#Change legend options!
plotcorr(ex1,
    args.legand = list(ncol=3))

#Change height points!
plotcorr(ex1,
    height.points = rep(-0.1,6))

## Not run:
## K = 3
K = 3
ex2 <- lba(parties ~ city,
    data=votB,
    K = 3)

plotcorr(ex2)

#Change budget options
plotcorr(ex2,
    pch.budget = 5,
    col.budget = 2,
    labels.budget = c('lba1','lba2','lba3'))

#Change points options
plotcorr(ex2,
    pch.points = 20,
    col.points = 4,
    labels.points = rownames(ex2$Aoi),
    args.legand = list(plot=FALSE))
```
# Coloring the groups
plotcorr(ex2,
    col.points = c(1,2,3,3,3,2),
    col.budget = c(3,1,2),
    args.legend = list(ncol=3))

#K = 4
K = 4
data(postmater)
new_post <- as.matrix(postmater[, -1])
row.names(new_post) <- postmater[, 1]

ex3 <- lba(new_post,
    K = K)

plotcorr(ex3)

# A bit didactic!
plotcorr(ex3,
    args.legend = list(x = -2.5,
                        y = 5.5,
                        xpd=TRUE,
                        ncol=5))

# Dynamic? Yes, you can!
plotcorr(ex3, rgl.use = TRUE)

## End(Not run)

---

**plotlba**

*Plotlba objects using the approach suggested by van der Ark (1999).*

**Description**

S3 methods for lba objects.

**Usage**

```r
## S3 method for class 'lba.1d'
plotlba(x,
    height.line = NULL,
    xlab = NULL,
    ylab = NULL,
    ylim = NULL,
    args.legend = NULL,
```
plotlba

labels.points = NULL,
col.points = par('col'),
col.lines = par('col'),
lty.lines = par('lty'),
lwd.lines = par('lwd'),
pch.budget = par('pch'),
col.budget = par('fg'),
lty.budget = par('lty'),
lwd.budget = par('lwd'),
colline.budget = NULL,
with.ml = c("mix","lat"),
...

## S3 method for class 'lba.2d'
plotlba(x,
    axis.labels = NULL,
    labels.points = NULL,
    col.points = par('fg'),
pch.budget = par('pch'),
col.budget = par('fg'),
lty.budget = par('lty'),
lwd.budget = par('lwd'),
colline.budget = par('fg'),
args.legend = NULL,
    with.ml = c("mix","lat"),
    ...
)

## Arguments

x
A object of lba class.

height.line
Is a vector with the lines height when \( K = 2 \).

xlab
A title for the x axis.

ylab
A title for the y axis.

ylim
The y limits of the plot.

args.legend
List of additional arguments to be passed to legend; names of the list are used as argument names. The default is NULL.

axis.labels
Labels for the three axes in the order left, right, bottom. Defaults to the column names.

labels.points
A character vector or expression specifying the text to be written. The default is NULL.

col.points
A vector of colour representing the points of the mixing parameters. The default is par('fg').

col.lines
A vector of colour representing the lines of the mixing parameters. The default is par('fg').

lty.lines
A vector of line types representing the mixing parameters. The default is par('lty').
lwd.lines  A vector of line width representing the mixing parameters. The default is `par('lwd')`.

pch.budget  A vector of plotting characters or symbols representing the budget proportion. The default is `par('pch')`.

col.budget  A vector of colour representing the budget proportion. The default is `par('fg')`.

type.budget  A vector of line types representing the budget proportion. The default is `par('lty')`.

lwd.budget  A vector of line width representing the budget proportion. The default is `par('lwd')`.

colline.budget  The colors for line budget. The default is `par('fg')`.

with.ml  What's parameters do you like to plot? The default is mixing parameters (`'mix'`).

...  Other graphical parameters may also be passed as arguments to these functions.

Details

The `plotlba` function, was suggested at de Leeuw et all (1990) and at van der Ark (1999) thesis. Those types of plots have only graphical views for \( K = 2 \) and \( K = 3 \). When \( K = 2 \), either the latent budgets or the mixing parameters are displayed on a (one dimensional) line segment. When \( K = 3 \), either the latent budgets or the mixing parameters are displayed in an equilateral triangle using a barycentric coordinate system where the budgets are represented by the vertices and the plot is made with help of `triax.plot` and `triax.points` function of `plotrix` package.

Author(s)

Enio G. Jelihovschi (<eniojelihovs@gmail.com>)
Ivan Bezerra Allaman (<ivanalaman@gmail.com>)

References


See Also

`triax.plot`, `triax.points`.

Examples

data('votB')

# K = 2
ex1 <- lba(parties ~ city,
            data=votB,
            K = 2)

plotlba(ex1)
#It's very simple. With colors!
plotlba(ex1,
   col.points = 1:6,
   col.lines = 1:6)

#Add title in plot!
plotlba(ex1,
   main='Mixing parameters')

#Change budget proportion!
plotlba(ex1,
   pch.budget = 23,
   col.budget = 9,
   colline.budget = 8,
   lwd.budget = 2,
   lty.budget = 2)

#A little more!
plotlba(ex1,
   xlab = 'Lb2 -> Lb1',
   height.line = rep(0.5,6),
   lty.lines = 2,
   args.legend = list(ncol=3))

## Not run:
#K = 3

data(MANHATAN)

tbm <- xtabs(value ~ socecon+health,
   MANHATAN)

ex2 <- lba(tbm,
   K = 3)

plotlba(ex2)

#A little more!
plotlba(ex2,
   labels.points = rownames(tbm),
   col.points = 2:7,
   args.legend = list(plot=F))

plotlba(ex2,
   col.points = 3:8,
   col.budget = 2,
   pch.budget = 20,
   lty.budget = 2,
   lwd.budget = 3,
   colline.budget = 3,
   axis.labels = c('Lba1','Lba2','Lba3'))
### Description

The `postmater` data frame has 13 rows and 8 columns. The raw data refers to a political and social survey across Europe that is conducted twice a year.

### Usage

`postmater`

### Format

This data frame contains the following columns:

- **country**: A factor with levels: `F` France; `B` Belgium; `NL` Netherlands; `D` Germany; `I` Italy; `L` Luxembourg; `DK` Denmark; `IRL` Ireland; `GB` Great Britain; `NIRL` Northern Ireland; `GR` Greece; `E` Spain; `P` Portugal.
- **m.**: The absolute frequencies of materialist factor in the respect country. The degree of ranking of this index is ++.
- **m.**: The absolute frequencies of materialist factor in the respect country. The degree of ranking of this index is +.
- **m**: The absolute frequencies of materialist factor in the respect country. The degree of ranking of this index is below of the "m."
- **m_pm**: The absolute frequencies of materialist/post-materialist factor in the respect country. The degree of ranking of this index is below of the "m".
- **pm**: The absolute frequencies of post-materialist factor in the respect country. The degree of ranking of this index is below of the "pm".
- **pm.**: The absolute frequencies of post-materialist factor in the respect country. The degree of ranking of this index is below of the "pm."
- **pm..**: The absolute frequencies of post-materialist factor in the respect country. The degree of ranking of this index is below of the "pm.".

### Source


### References

Description

The pregnancy matrix has 16 rows and 5 columns. The raw data refers to California pregnancy-related deaths from 2002-2005.

Usage

pregnancy

Format

This matrix contains the following columns:

Pre.E  Preeclampsia/eclampsia
OH     Obstetric hemorrhage
CVD    Cardiovascular diseases
DVTPE  Deep vein thrombosis - pulmonary embolism
AFE    Amniotic fluid embolism

The rows refers to:

Hfob  Hispanic, foreign-born
Husb  Hispanic, us-born
Wnh   White, non-hispanic
Bnh   Black, non-hispanic
$<30$b Maternal age
$30-40$b Maternal age
$>40$b Maternal age
$1$    Parity
$2-4$  Parity
$5+$   Parity
$<30$a Maternal age
$30-40$a Maternal age
$>40$a Maternal age
$<32$w Gestational age at delivery
$32-36$w Gestational age at delivery
$>37$w Gestational age at delivery
Source

References

---

**print.goodnessfit**

*Print Method for goodnessfit objects.*

**Description**

Returns (and prints) a summary list for goodnessfit objects.

**Usage**

```r
## S3 method for class 'goodnessfit.lba.ls'
print(x, digits=3L, ...)

## S3 method for class 'goodnessfit.lba.ls.fe'
print(x, digits=3L, ...)

## S3 method for class 'goodnessfit.lba.ls.logit'
print(x, digits=3L, ...)

## S3 method for class 'goodnessfit.lba.mle'
print(x, digits=3L, ...)

## S3 method for class 'goodnessfit.lba.mle.fe'
print(x, digits=3L, ...)

## S3 method for class 'goodnessfit.lba.mle.logit'
print(x, digits=3L, ...)
```

**Arguments**

- `digits` A non-null value for `digits` specifies the minimum number of significant digits to be printed in values. The default is 3.
- `...` Further arguments (require by generic).
Author(s)
Enio G. Jelihovschi (<eniojelihovs@gmail.com>)
Ivan Bezerra Allaman (<ivanalaman@gmail.com>)

See Also
summary.goodnessfit.lba.ls, summary.goodnessfit.lba.mle

Examples

data('votB')

# Using LS method (default) without constraint
# K = 2
ex1 <- lba(city ~ parties,
          votB,
          K = 2)
exm <- goodnessfit(ex1)
exm

print.lba

Print Method for lba objects.

Description

Returns (and prints) a summary list for objects of class lba.ls, lba.ls.fe, lba.ls.logit, lba.mle,
 lba.mle.fe, and lba.mle.logit.

Usage

## S3 method for class 'lba.ls'
print(x, digits = 3L, ...)

## S3 method for class 'lba.ls.fe'
print(x, digits = 3L, ...)

## S3 method for class 'lba.ls.logit'
print(x, digits = 3L, ...)

## S3 method for class 'lba.mle'
print(x, digits = 3L, ...)

## S3 method for class 'lba.mle.fe'
print(x, digits = 3L, ...)

## S3 method for class 'lba.mle.logit'
print(x, digits = 3L, ...)
Arguments

x  A given object of the class lba, lba.ls.fe, lba.mle.fe, lba.ls.logit and lba.mle.logit.
digits  Number of decimal digits in the results. The default is 3.
...

Further arguments (require by generic).

Author(s)

Enio G. Jelihovschi (<eniojelihovs@gmail.com>)
Ivan Bezerra Allaman (<ivanalaman@gmail.com>)

See Also

lba

Examples

data('votB')

# Using LS method (default) without constraint
# K = 2
ex1 <- lba(city ~ parties,
  votB,
  K = 2)

ex1

summary.goodnessfit  Summary Method for goodnessfit objects.

Description

Returns (and prints) a summary list for goodnessfit objects.

Usage

## S3 method for class 'goodnessfit.lba.ls'
summary(object, digits = 2L, ...)

## S3 method for class 'goodnessfit.lba.ls.fe'
summary(object, digits = 2L, ...)

## S3 method for class 'goodnessfit.lba.ls.logit'
summary(object, digits = 2L, ...)


## S3 method for class 'goodnessfit.lba.mle'
summary(object, digits = 2L, ...)

## S3 method for class 'goodnessfit.lba.mle.fe'
summary(object, digits = 2L, ...)

## S3 method for class 'goodnessfit.lba.mle.logit'
summary(object, digits = 2L, ...)

### Arguments
- **object**: A given object of the class `goodnessfit.lba.ls` and `goodnessfit.lba.mle`.
- **digits**: Number of decimal digits in the results. The default is 2.
- **...**: Further arguments (require by generic).

### Author(s)
Enio G. Jelihovschi (<eniojelihovs@gmail.com>)
Ivan Bezerra Allaman (<ivanalaman@gmail.com>)

### See Also
- `goodnessfit`

### Examples
```r
data('votB')

# Using LS method (default) without constraint
# K = 2
ex1 <- lba(city ~ parties,
           votB,
           K = 2)
exm <- goodnessfit(ex1)
summary(exm)
```

---

### summary.lba

**Summary Method for lba objects.**

### Description

Returns (and prints) a summary list for objects of class `lba`, `lba.ls`, `lba.ls.logit`, `lba.mle`, `lba.mle.fe`, and `lba.mle.logit`.
Usage

```r
## S3 method for class 'lba.ls'
summary(object, digits = 2L, ...)

## S3 method for class 'lba.ls.fe'
summary(object, digits = 2L, ...)

## S3 method for class 'lba.ls.logit'
summary(object, digits = 2L, ...)

## S3 method for class 'lba.mle'
summary(object, digits = 2L, ...)

## S3 method for class 'lba.mle.fe'
summary(object, digits = 2L, ...)

## S3 method for class 'lba.mle.logit'
summary(object, digits = 2L, ...)
```

Arguments

- `object`: A given object of the class `lba`, `lba.ls.fe`, `lba.mle.fe`, `lba.ls.logit` and `lba.mle.logit`.
- `digits`: Number of decimal digits in the results. The default is 2.
- `...`: Further arguments (require by generic).

Author(s)

Enio G. Jelihovschi (<eniojelihovs@gmail.com>)
Ivan Bezerra Allaman (<ivanalaman@gmail.com>)

See Also

- `lba`

Examples

```r
data('votB')

# Using LS method (default) without constraint
# K = 2
ex1 <- lba(city ~ parties,
          votB,
          K = 2)
summary(ex1)
```
**Description**

The `votB` data frame has 8971 rows and 2 columns. The raw data refers to the type of the city and the political party which each participant voted for in the 1986 general elections in the Netherlands.

**Usage**

`votB`

**Format**

This data frame contains the following columns:

- **city** A factor with levels: `co` Commuter; `lx` Large city; `mc` Middle large city; `ri` Rural industrialized; `ru` Rural; `sc` Small city.
- **parties** A factor with levels: `cda` Christian democrats; `d66` Democrats; `left` Other left-wing parties; `pvda` Labor party; `right` Other right-wing parties; `vvd` Liberals.

**Source**


**References**

Index

*Topic **correspondence analysis**
  plotcorr, 19
*Topic **datasets**
  housing, 6
  MANHATAN, 16
  PerfMark, 17
*Topic **dataset**
  postmater, 28
  pregnancy, 29
  votB, 35
*Topic **goodnessfit**
  print.goodnessfit, 30
  summary.goodnessfit, 32
*Topic **lba**
  plotcorr, 19
  plotlba, 24
  print.lba, 31
  summary.lba, 33
*Topic **package**
  Latent Budget Analysis, 7
  print.goodnessfit, 30
  print.lba, 31
  summary.goodnessfit, 32
  summary.lba, 33
*Topic **plot**
  plotcorr, 19
  plotlba, 24

goodnessfit, 2, 15, 33
housing, 6

Latent Budget Analysis, 7
lba, 4, 7, 32, 34
lba-package (Latent Budget Analysis), 7

MANHATAN, 16

PerfMark, 17
plot.default, 22
plot3d, 22

plotcorr, 15, 19
plotlba, 15, 24
postmater, 28
pregnancy, 29
print.goodnessfit, 30
print.lba, 31
scatterplot3d, 22
summary.goodnessfit, 32
summary.goodnessfit.lba.ls, 4, 31
summary.goodnessfit.lba.mle, 4, 31
summary.lba, 33
summary.lba.ls, 15
summary.lba.mle, 15

triax.plot, 26
triax.points, 26

votB, 35