Package ‘OrdinalLogisticBiplot’

November 26, 2013

Type Package

Title Biplot representations of ordinal variables

Version 0.2

Date 2013-11-25

Author Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon

Maintainer Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

Description Analysis of a matrix of polytomous items using Ordinal Logistic Biplots (OLB)
   The OLB procedure extends the binary logistic biplot to ordinal (polytomous) data.
   The individuals are represented as points on a plane and the variables are represented
   as lines rather than vectors as in a classical or binary biplot, specifying the points
   for each of the categories of the variable.
   The set of prediction regions is established by stripes perpendicular to the line
   between the category points, in such a way that the prediction for each individual is given
   by its projection into the line of the variable.

License GPL (>= 2)

Encoding latin1

Repository CRAN

Depends R (>= 2.15.1),mirt,MASS,NominalLogisticBiplot

LazyData yes

Archs i386, x64

NeedsCompilation no

Date/Publication 2013-11-26 18:40:30
Ordinal Logistic Biplot representations for polytomous ordered data.

Description

Analysis of a matrix of polytomous ordered items using Ordinal Logistic Biplots (OLB). The OLB procedure extends the binary logistic biplot to ordinal (polytomous) data.

Details

Package: OrdinalLogisticBiplot
Type: Package
Version: 0.1
Date: 2013-10-29
License: GPL (>=2)

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>
**BiplotDensity**

**Density plot of a variable with contours overlaid.**

**Description**

This function draws for a set of points a density contour lines biplot. This tool uses the `kde2d` function of library MASS.

**Usage**

```r
BiplotDensity(X, y = NULL, nlevels = max(y), grouplabels = 1:nlevels, ncontours = 6, groupcols = 1:nlevels, img = TRUE, separate = FALSE, ncolors = 20, ColorType = 4, xlim = -1, ylim = 1, ylimu = -1, ylim1 = 1, margin = 0.2, plotind = FALSE)
```

**Arguments**

- `X` A matrix with the items coordinates for the plane in which the the contour lines will be plotted.
- `y` Categorical variable that the user wants to study.
- `nlevels` Maximum value of the variable specified in the second parameter.
- `grouplabels` Set of labels for the centers of each category. It should be a vector.
- `ncontours` Number of contours that will be used in the representation.
- `groupcols` Set of colors for each contour lines group. It should be a vector.
- `img` Boolean parameter to specify if an image will be plotted behind the contour lines indicating with diferente colorations the density of the points. Default value is TRUE.
- `separate` Boolean parameter to specify for each category of the variable a separate biplot representation. Default value is FALSE.
- `ncolors` Number of colors that will be used in the image.
- `ColorType` Type of coloration for the image in the biplot. It should be a number between 1 and 5.

**Examples**

```r
data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd,sFormula=NULL,numFactors=2, coordinates="EM",penalization=0.2,show=FALSE)
summary(olbo)
plot(olbo,PlotInd=TRUE,xlim=-1,xlimu=1,ylim=-1,ylimu=1,margin = 0.2,
ColorVar = c("red","green","black","blue","yellow"),CexVar = c(0.7),showIC=FALSE)
```
CheckDataSet

Check the data set.

Description

This function checks if the data is a data frame or a matrix and saves the data as a matrix of integers, and stores the names of rows, columns and levels for each variable as vectors to use them later.

Usage

CheckDataSet(datanom)

Arguments

datanom It can be a data frame or a matrix.

Details

The function checks if some variable has NA values and deletes the corresponding row. Also checks for missing categories and recodifies the variable keeping the original labels for levels.

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon

Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

Examples

data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd[, 1:4])
x = olbo$ItemsCoords[, 1]
y = olbo$ItemsCoords[, 2]
plot(x, y, cex = 0, xlim = c(-1, 1), ylim = c(-1, 1))
X = olbo$ItemsCoords
y = as.matrix(as.numeric(LevelSatPhd[, 4]))
gcols = c("midnightblue", "black", "red", "gray87")
BiplotDensity(X, y, groupcols = gcols)
Value

An object of class "data.ordinal". This has components:

datanom Matrix of integers with the values of the variables
RowNames Vector with the names of the rows
ColumnNames Vector with the names of the variables
LevelNames Levels of each variable

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon

Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

Examples

data(LevelSatPhd)
dataChecked = CheckDataSet(LevelSatPhd)

Description

This function uses mirt and fscores functions of "mirt" package to estimate the coordinates for each row and the parameters for each variable.

Usage

EstimationRowsMIRT(dataFile,numFactors=2,metfsco="EAP",rotation="varimax",maxiter=100)

Arguments

dataFile Data set with ordinal variables.
numFactors Number of dimensions of the solution. It should be lower than the number of variables. It has a default value of 2.
metfsco Calculation method for the fscores.
rotation Rotation method.
maxiter Maximum number of iterations executed by the algorithm.
Value

An object of class "EstimationRowsMIRT". This has some components:

- `estimRows`: Coordinates for the rows or the individuals
- `dataFactor`: Data set of ordinal variables
- `rotation`: Rotation method
- `metfsco`: Calculation method for the fscores
- `numFactors`: Number of dimensions of the solution
- `coefMirt`: List with the estimated coefficients for all the variables
- `sepCoefMirt`: Object with two matrices with the estimated coefficients for the independent variables and for the thresholds
- `summ`: Object with the information provided by summary function for mirt one

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

Examples

data(LevelSatPhd)
erm = EstimationRowsMIRT(LevelSatPhd)

GetOrdinalBiplotObjectMIRT

Calculation of Mirt ordinal biplot object.

Description

This function uses the estimation from the MIRT method to calculate all the information needed for representing the ordinal biplot: the biplot axis for each variable with the points separating the visible categories.

Usage

GetOrdinalBiplotObjectMIRT(modelMirt, planex = 1, planey = 2)

Arguments

- `modelMirt`: Object of class "EstimationRowsMIRT" with all the estimated parameters with Mirt method.
- `planex`: Dimension for X axis.
- `planey`: Dimension for Y axis.
Details

This function should only be used if the user selects "MIRT" coordinates as the estimation method in OrdinalLogisticBiplot function.

Value

An object of class "CategOrdBiplot". This has components:

- coefMirt: Estimated coefficients from Mirt method
- sepCoeffMirt: Separated coefficients for the independent variables and for the thresholds
- numFactors: Number of dimensions of the solution
- rotation: Rotation method used with "MIRT" option
- metFscro: Calculation method for the fscores with "MIRT" option
- planex: Dimension for X axis.
- planey: Dimension for Y axis.
- scores: Coordinates for the rows in the reduced space
- matBiplot: This item keeps a matrix with the same structure than in GetOrdinalBiplotObjectPenal function
- summaryMirt: Summary parameters for the mirt procedure

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

See Also

OrdinalLogisticBiplot, GetOrdinalBiplotObjectPenal, EstimationRowsMIRT

Examples

data(LevelSatPhd)
olboM = OrdinalLogisticBiplot(LevelSatPhd,coordinates="MIRT",metFscro="EAP",rotation="varimax")
catOrdBiplot = GetOrdinalBiplotObjectMIRT(olboM$estimObject)

GetOrdinalBiplotObjectPenal

Calculation of the ordinal biplot object.

Description

This function uses the estimation from the EM alternated algorithm to calculate all the information needed for representing the ordinal biplot, as the biplot axis for each variable and the points separating the visible categories.
GetOrdinalBiplotObjectPenal

Usage

GetOrdinalBiplotObjectPenal(ColumnNames, olb, planex = 1, planey = 2)

Arguments

ColumnNames Vector with the names of the studied variables.
olb Object of class "ordinal.logistic.biplotEM" with the alternated algorithm estimation.
planex Dimension for X axis.
planey Dimension for Y axis.

Details

This function should only be used if the user selects "EM" coordinates as the estimation method in OrdinalLogisticBiplot function.

Value

An object of class "CategOrdBiplotPenal". This has components:

models Estimated object with the EM alternated algorithm of the class "ordinal.logistic.biplotEM". This object has the following items: RowCoordinates(Coordinates for the rows in the reduced space),ColumnParameters(estimated values for the coefficients and the thresholds and indicators of the fitting, like logLik, Deviance, p-values and pseudo R-squared values),loadings(factor loadings values),LogLikelihood(total logarithm of the likelihood),r2(communalities) and Ncats(number of categories of each variable)

matBiplot Matrix with so many columns as variables. Each column keeps a list of variables describing the biplot elements. The variables are: var(name of the variable), cosines(cosines of the angles between the straight line and the axes), numcat(Number of categories of the variable), coef(estimated coefficients for this variable), slope(slope of the biplot axis), order(boolean variable that is TRUE if the coordinates of the curves intersections [1-2, 2-3, ..., (numcat-1)-numcat] with the biplot axis are ordered), pointsc(matrix with the intersection points between curves calculated as before), pointprob(matrix with the intersection points of each curve with the rest of them and with the biplot axis. It has 5 columns: the first one is the probability for the points in witch the curves are equal, the second an third are the coordinates x and y of the intersection of the two curves with the biplot axis, and the last two are the categories compared. This matrix has zeros if the variable order is TRUE, because it has not been calculated because it is not necessary.)

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon

Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>
**LevelSatPhd**

**See Also**

*OrdinalLogBiplotEM, OrdinalLogisticBiplot*

**Examples**

```r
data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd)
ColumnNames = olbo$dataSet$ColumnNames
olb = olbo$estimObject
catOrdBiplotPenal = GetOrdinalBiplotObjectPenal(ColumnNames, olb)
```

---

**LevelSatPhd**

<table>
<thead>
<tr>
<th>Questionnaire Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Salary</td>
</tr>
<tr>
<td>Benefits</td>
</tr>
<tr>
<td>Job Security</td>
</tr>
<tr>
<td>Job Location</td>
</tr>
<tr>
<td>Working conditions</td>
</tr>
</tbody>
</table>

**Description**

The sample data, as part of a large survey, corresponds to 10 people who have the PhD degree and shows the level of satisfaction about some issues.

**Usage**

```r
data(LevelSatPhd)
```

**Format**

This data frame contains 10 observation for the following 5 ordinal variables, with four categories each: (1= "Very Satisfied", 2= "Somewhat Satisfied", 3= "Somewhat dissatisfied", 4= "Very dissatisfied")

**Salary**

**Benefits**

**Job Security**

**Job Location**

**Working conditions**

**Source**


**Examples**

```r
data(LevelSatPhd)
```
Alternated EM algorithm for Ordinal Logistic Biplots

Description

This function computes, with an alternated algorithm, the row and column parameters of an Ordinal Logistic Biplot for ordered polytomous data. The row coordinates (E-step) are computed using multidimensional Gauss-Hermite quadratures and Expected a posteriori (EAP) scores and parameters for each variable or items (M-step) using Ridge Ordinal Logistic Regression to solve the separation problem present when the points for different categories of a variable are completely separated on the representation plane and the usual fitting methods do not converge. The separation problem is present in almost every data set for which the goodness of fit is high.

Usage

OrdinalLogBiplotEM(x, dim = 2, nnodos = 15, tol = 0.001, maxiter = 100, penalization = 0.2, show = TRUE, initial = 2, alfa = 1)

Arguments

x                  Matrix with the ordinal data. The matrix must be in numerical form.
dim                Dimension of the solution
nnodos             Number of nodes for the multidimensional Gauss-Hermite quadrature
tol                Value to stop the process of iterations.
maxiter            Maximum number of iterations in the process of solving the regression coefficients.
penalization       Penalization used in the diagonal matrix to avoid singularities.
show               Boolean parameter to specify if the user wants to see every iteration.
initial            Method used to choose the initial ability in the algorithm.
alfa               Optional parameter to calculate row and column coordinates in Simple correspondence analysis if the initial parameter is equal to 1.

Value

An object of class "ordinal.logistic.biplot.EM". This has components:

RowCoordinates  Coordinates for the rows or the individuals
ColumnParameters List with information about the Ordinal Logistic Models calculated for each variable including: estimated parameters with thresholds, percents of correct classifications, and pseudo-R squared
loadings        factor loadings
LogLikelihood    Logarithm of the likelihood
r2               R squared coefficient
Ncats            Number of the categories of each variable
**Author(s)**

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

**References**


**See Also**

`pordlogist`

**Examples**

```r
data(LevelSatPhd)
dataSet = CheckDataSet(LevelSatPhd)
datanom = dataSet$data$datanom
olb = OrdinalLogBiplotEM(datanom,dim = 2, nnodos = 10,
tol = 0.001, maxiter = 100, penalization = 0.2)
olb
```

---

**OrdinalLogisticBiplot** *Ordinal Logistic Biplot for ordered polytomous data*

**Description**

Function that calculates the parameters of the Ordinal Logistic Biplot.

**Usage**

```r
OrdinalLogisticBiplot(datanom,sFormula=NULL,numFactors=2,
coordinates="EM",rotation="varimax",metfsco="EAP",
nodos = 10, tol = 0e-04, maxiter = 100,
penalization = 0.1,cte=TRUE, show=TRUE,ItemCurves = FALSE,initial=2,alfa=1)
```

**Arguments**

- `datanom`: The data set, it can be a matrix with integers or a data frame with factors. All variables have to be ordinal.
- `sFormula`: This parameter follows the unifying interface for selecting variables from a data frame for a plot, test or model. The most common formula es of type y ~ x1+x2+x3. It has a default value of NULL if not specified.
- `numFactors`: Number of dimensions of the solution. It should be lower than the number of variables. It has a default value of 2.
- `coordinates`: This parameter can be: "EM" or "MIRT". Method to compute the row coordinates.
rotation

Rotation method to used with "MIRT" option in "coordinates". No effect for other options.

metfsco

Calculation method for the fscores with "MIRT" option in "coordinates". No effect for other options.

nnodos

Number of nodes for gauss quadrature in the EM algorithm.

tol

Tolerance for the EM algorithm.

maxiter

Maximum number of iterations in the EM algorithm.

penalization

Penalization for the ridge regression for each variable.

cte

Include constant in the logistic regression model. Default is TRUE.

show

Show intermediate computations. Default is TRUE.

ItemCurves

Show item information curves. Default is FALSE.

initial

Method used to choose the initial ability in the EM algorithm. Default value is 2.

alfa

Optional parameter to calculate row and column coordinates in Simple correspondence analysis if the initial parameter is equal to 1. Default value is 1.

Details

The general algorithm used is essentially an alternated procedure in which parameters for rows and columns are computed in alternated steps repeated until convergence. Parameters for the rows are calculated by expectation (E-step) and parameters for the columns are computed by maximization (M-step), i.e., by Ordinal Logistic Regression.

There are several options for the computation:

1.- Using the package mirt to obtain the row scores, i.e., using a solution obtained from a latent trait model. The column (item) parameters should be directly used by our biplot procedure but, because of the characteristics of the package that performs a default rotation after parameter estimation, we have to reestimate the item parameters to be coherent to the scores.

2.- Using our implementation of the EM algorithm alternating expected a posteriori scores and Ridge Nominal Logistic Regression for each variable. We use here a Cumulative link model, that is, a logistic regression model for cumulative logits.

Equations defining the set of probability response surfaces for the cumulative probabilities are sigmoidal as in the binary case (Vicente-Villardon et al. 2006) and then share its geometry. All categories have a different constant but the same slopes, that means that the prediction direction is common to all categories and just the prediction markers are different. The representation subspace can be divided into prediction regions, for each category, delimited by parallel straight lines.

Value

An object of class "ordinal.logistic.biplot". This has some components:

dataSet

Data set of study with all the information about the name of the levels and names of the variables and individuals

ItemsCoords

Coordinates for the rows in the reduced space

NCats

Number of categories of each variable from the data set
estimObject: Object with all the estimated information using EM alternated algorithm or MIRT procedure.

Fitting: Matrix with the percentage of correct classifications and pseudo R squared values for each variable.

coops: Matrix with the estimated coefficients.

thresholds: Matrix with the estimated intercept limits.

NumFactors: Number of dimensions selected for the study.

Coordinates: Type of coordinates to calculate the row positions.

Rotation: Type of rotation if we have chosen mirt coordinates.

Methodfscores: Method of calculation of the fscores in mirt process.

NumNodos: Number of nodes for the gauss quadrature in EM algorithm.

tol: Cut point to stop the EM-algorithm.

maxiter: Maximum number of iterations in the EM-algorithm.

penalization: Value for the correction of the ridge regression.

cute: Boolean value to choose if the model for each variable will have independent term.

show: Boolean value to indicate if we want to see the results of our analysis.

ItemCurves: Boolean value to specify if item information curves will be plotted.

LogLik: Logarithm of the likelihood.

FactorLoadingsComm: Factor loadings and communalities.

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon

Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

References


plot.ordinal.logistic.biplot


See Also

OrdinalLogBiplotEM

Examples

data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd)
summary(olbo)

plot.ordinal.logistic.biplot

*Graphical representation of an Ordinal Logistic Biplot.*

Description

Plotting an Ordinal Logistic Biplot. There are parameters related to the way in which the biplot is plotted. All the possible parameters have default values.

Usage

```r
## S3 method for class 'ordinal.logistic.biplot'
plot(x, planex = 1, planey = 2,
     AtLeastR2 = 0.01, xlim = -1.5, xlimu = 1.5, ylim = -1.5,
     ylimu = 1.5, margin = 0, ShowAxis = TRUE, PlotVars = TRUE,
     PlotInd = TRUE, LabelVar = TRUE, LabelInd = TRUE, CexInd = NULL,
     CexVar = NULL, ColorInd = NULL, ColorVar = NULL, PchInd = NULL,
     PchVar = NULL, showIIC = FALSE, iiçx1 = -1.5, iiçxu = 1.5,
     legendPlot = FALSE, PlotClus = FALSE, Clusters=NULL,
     chulls = TRUE, centers = TRUE, colorCluster = NULL,
     ConfidentLevel=NULL, addToExistingPlot=FALSE,...)
```

Arguments

- **x** An object of the class ordinal.logistic.biplot.
- **planex** Dimension for X axis.
- **planey** Dimension for Y axis.
AtLeastR2: It establishes the cutting value to plot a variable attending to its Nagelkerke pseudo R squared value. A variable is plotted if its pseudo R squared is higher than this value.

xlimi: Minimum value on the x-axis.

xlimu: Maximum value on the x-axis.

ylimi: Minimum value on the y-axis.

ylimu: Maximum value on the y-axis.

margin: This value establishes the space between the plotted items and the border of the window.

ShowAxis: Should the axis be shown?

PlotVars: Should the variables (items) be plotted?

PlotInd: Should the individuals be plotted?

LabelVar: Should the variable labels be shown?

LabelInd: Should the individual labels be shown?

CexInd: Size of the individual points. It can be an array with the cex information for each row.

CexVar: Size of the category points. It can be an array with the cex information for each variable.

ColorInd: Color of the individual points. It can be an array with the color information for each row.

ColorVar: Color for the variables. It can be an array with the color information for each variable.

PchInd: Symbol for the individuals. It can be an array with the pch information for each row.

PchVar: Symbol for the variables. It could be an array with the pch information for each variable.

showIIC: Boolean parameter to decide if the user wants to see the item information curves for each variable. Default value is FALSE.

iicxi: Lower limit for the X-axis when plotting item information curves.

iicxu: Upper limit for the X-axis when plotting item information curves.

legendPlot: Boolean parameter to show the legend of the plot. Default value is FALSE.

PlotClus: Boolean parameter to show the clusters studied. Default value is FALSE.

Clusters: Variable with the cluster associated for each item. Default value is NULL.

chulls: Boolean parameter to specify if it will be plotted convex hulls figures. Default value is FALSE.

centers: Boolean parameter to plot the centers of each cluster. Default value is NULL.

colorCluster: Color for every cluster. It can be an array with the color information for each cluster. Default value is NULL.

ConfidentLevel: Value between 0 and 1 to avoid extreme values for the plot. Default value is NULL.

addToExistingPlot: Boolean parameter to decide if the plotted items will be added to an existing plot or not. Default value is FALSE.

...: Additional parameters to plot.
Details

The function without parameters plots the ordinal.logistic.biplot object with labels in the original
data and default values for colors, symbols and sizes for points and lines. Other values of colors,
symbols and sizes can be supplied. A single value applies to all the points but an array with different
values can be used to improve the undestanding of the plot.

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

See Also

OrdinalLogisticBiplot

Examples

data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd,penalization=0.2)
plot(olbo,PlotInd=TRUE,xlim1=-1.5,xlimu=1.5,ylim1=-1.5,ylimu=1.5,
margin = 0.2, ColorVar = c("red","green","black","blue","yellow"),
CexVar = c(0.7),showIIC=FALSE)

---

PlotClusterVariable

Graphical representation of clustering variables.

Description

This function uses a clustering variable to represent the centers or convex hulls regions of the points
that share each cluster. This variable could be the result of the application of one of the main
clustering methods to the data set.

Usage

PlotClusterVariable(A, Groups = ones(c(nrow(A), 1)),
colors = NULL, chulls = TRUE, centers = TRUE, ConfidentLevel = 0.95)

Arguments

A A matrix with the items coordinates for the plane of study. It should have only
two columns.
Groups Clustering variable: it keeps the cluster for each observation.
colors It is a vector used to specify the colors of the centers and convex hulls regions
for each cluster.
chulls Should convex hulls regions be plotted?
centers Should centers of each cluster be plotted?
ConfidentLevel  Numerical value between 0 and 1. If it’s value is 0.95, five percent of the points with higher distances to the center of each cluster will not be used for the convex hulls and center calculations that will be plotted.

Author(s)
Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

Examples

```r
data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd)
x = olbo$ItemsCoords[, 1]
y = olbo$ItemsCoords[, 2]
plot(x,y, cex = 0.8, pch=17, xlim=c(-2,2),ylim=c(-2,2))
GroupsF = as.factor(LevelSatPhd[,4])
PlotClusterVariable(olbo$ItemsCoords, Groups = GroupsF,
    colors = c(1,2,3,4),chulls = TRUE,centers = TRUE,ConfidentLevel=NULL)
```

Description

This function plots an ordinal variable to an existing biplot or to a new one, using the estimated parameters for the variable.

Usage

```r
plotOrdBipCoeffVariable(nameVariable, coeffic, D = 1, planex = 1, planey = 2,
    xi = -3.5, xu = 3.5, yi = -3.5, yu = 3.5, margin = 0,
    numFactors = 2, CexVar = 0.7, ColorVar = "blue",
    PchVar = 0.7, addToPlot = FALSE, showIIC = TRUE,
    iiicxi = -2.5, iiicxu = 2.5)
```

Arguments

- **nameVariable**: Name of the variable the user wants to plot.
- **coeffic**: Vector with the estimated coefficients and the thresholds in this order.
- **D**: Value that define the graded response model. In case of Mirt estimation it should be 1.702. The default value is 1.
- **planex**: Dimension for X axis.
- **planey**: Dimension for Y axis.
xi Minimum value on the x-axis.
xu Maximum value on the x-axis.
yi Minimum value on the y-axis.
yu Maximum value on the y-axis.
margin This value stablishes the space between the plotted items and the border of the window.
numFactors Number of dimensions of the solution
CexVar Size of the category points. It can be an array with the cex information for each variable.
ColorVar Color for the variables. It can be an array with the color information for each variable.
PchVar Symbol for the variables. It could be an array with the pch information for each variable.
addToPlot Boolean parameter to decide if the user wants to add the ordinal variable representation to an existing plot.
showIIC Boolean parameter to decide if the user wants to see the item information curves for each variable. Default value is FALSE.
iicxi Lower limit for the X-axis when plotting item information curves.
iicxu Upper limit for the X-axis when plotting item information curves.

Author(s)
Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintenance: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

Examples

data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd,sFormula=NULL,
numFactors=2,coordinates="EM",penalization=0.2)
nNameVariable="Salary"
ColumnNames = olbo$dataSet$ColumnNames
olb = olbo$estimObject
catOrdBiplotPenal = GetOrdinalBiplotObjectPenal(ColumnNames,olb)
coeffic = catOrdBiplotPenal$matBiplot[,1]$coef
plotOrdBipCoeffVariable(nameVariable,coeffic)
plotOrdinalVariable

Plot an ordinal variable to an existing biplot.

Description

This function plots an ordinal variable to an existing biplot or to a new one, using only the variable and the coordinates of the items.

Usage

plotOrdinalVariable(ordinalfVar, nameVariable, estimRows, planex = 1, planey = 2, xi = -3.5, xu = 3.5, yi = -3.5, yu = 3.5, margin = 0, CexVar = 0.7, ColorVar = "blue", PchVar = 0.7, addToPlot = FALSE, showIIC = TRUE, iicxi = -2.5, iicxu = 2.5, tol = 1e-04, maxiter = 100, penalization = 0.1)

Arguments

- ordinalfVar: The ordinal variable. It must be an ordered factor.
- nameVariable: Name of the variable that the user wants to represent.
- estimRows: Matrix with the estimated coordinates for the items in the reduced dimension.
- planex: Dimension for X axis.
- planey: Dimension for Y axis.
- xi: Minimum value on the x-axis.
- xu: Maximum value on the x-axis.
- yi: Minimum value on the y-axis.
- yu: Maximum value on the y-axis.
- margin: This value establishes the space between the plotted items and the border of the window.
- CexVar: Size of the category points. It can be an array with the cex information for each variable.
- ColorVar: Color for the variables. It can be an array with the color information for each variable.
- PchVar: Symbol for the variables. It could be an array with the pch information for each variable.
- addToPlot: Boolean parameter to decide if the user wants to add the ordinal variable representation to an existing plot.
- showIIC: Boolean parameter to decide if the user wants to see the item information curves for each variable. Default value is FALSE.
- iicxi: Lower limit for the X-axis when plotting item information curves.
- iicxu: Upper limit for the X-axis when plotting item information curves.
- tol: Tolerance for the iterations.
- maxiter: Maximum number of iterations.
- penalization: Penalization used to avoid singularities.
Author(s)
Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

Examples

data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd,sFormula=NULL,
   numFactors=2,coordinates="EM")
ordinalfVar = factor(LevelSatPhd[,1],ordered=TRUE)
levels(ordinalfVar) = c("VS","SS","SD","VD")
estimRows = olbo$ItemsCoords
nameVariable = "Salary"
plotOrdinalVariable(ordinalfVar,nameVariable,estimRows,planex = 1,
   planey = 2,xi=-1.5,xu=1.5,yi=-1.5,yu=1.5,
   margin=0.2,CexVar=0.7,showIIC = TRUE)

pordlogist
Ordinal logistic regression with ridge penalization

Description
This function does a logistic regression between a dependent ordinal variable y and some independent variables x, and solves the separation problem using ridge penalization.

Usage
pordlogist(y, x, penalization = 0.1, tol = 1e-04, maxiter = 200, show = FALSE)

Arguments
y  Dependent variable.
x  A matrix with the independent variables.
penalization  Penalization used to avoid singularities.
tol  Tolerance for the iterations.
maxiter  Maximum number of iterations.
show  Should the iteration history be printed?

Details
The problem of the existence of the estimators in logistic regression can be seen in Albert (1984); a solution for the binary case, based on the Firth’s method, Firth (1993) is proposed by Heinze(2002). All the procedures were initially developed to remove the bias but work well to avoid the problem of separation. Here we have chosen a simpler solution based on ridge estimators for logistic regression Cessie(1992).
Rather than maximizing $L_j(G|b_j, B_j)$ we maximize

$$L_j(G|b_j, B_j) - \lambda (\|b_j\| + \|B_j\|)$$

Changing the values of $\lambda$ we obtain slightly different solutions not affected by the separation problem.

**Value**

An object of class "pordlogist". This has components:

- **nobs**: Number of observations
- **J**: Maximum value of the dependent variable
- **nvar**: Number of independent variables
- **fitted.values**: Matrix with the fitted probabilities
- **pred**: Predicted values for each item
- **Covariances**: Covariances matrix
- **clasif**: Matrix of classification of the items
- **PercentClasif**: Percent of good classifications
- **coefficients**: Estimated coefficients for the ordinal logistic regression
- **thresholds**: Thresholds of the estimated model
- **logLik**: Logarithm of the likelihood
- **penalization**: Penalization used to avoid singularities
- **Deviance**: Deviance of the model
- **DevianceNull**: Deviance of the null model
- **Dif**: Difference between the two deviances values calculated
- **df**: Degrees of freedom
- **pval**: p-value of the contrast
- **CoxSnell**: Cox-Snell pseudo R squared
- **Nagelkerke**: Nagelkerke pseudo R squared
- **MacFaden**: Nagelkerke pseudo R squared
- **iter**: Number of iterations made

**Author(s)**

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon

Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>
References


See Also

`OrdinalLogBiplotEM, CheckDataSet`

Examples

```r
data(LevelSatPhd)
dataSet = CheckDataSet(LevelSatPhd)
datanom = dataSet$datanom
olb = OrdinalLogBiplotEM(datanom, dim = 2, nnodos = 10,
                         tol = 0.001, maxiter = 100, penalization = 0.2)
model = pordlogist(datanom[, 1], olb$RowCoordinates, tol = 0.001,
                    maxiter = 100, penalization = 0.2)
model
```

**summary.ordinal.logistic.biplot**

Summary Method Function for Objects of Class ‘ordinal.logistic.biplot’

Description

This function shows a summary of the principal results for the estimation for individuals and variables, like some Pseudo R-squared indices, the percent of correct classifications for each regression, the logLikelihood and "Estimate coefficients", "Std. Error", "z value" or "Pr(>|z|)" values.

Usage

```r
## S3 method for class 'ordinal.logistic.biplot'
summary(object, data = FALSE, itemCoords = FALSE,...)
```
Arguments

object This parameter keeps the ordinal logistic biplot object
data Boolean parameter to show the number of observations. Default value is FALSE.
itemCoords Boolean parameter to show the coordinates of the individuals. Default value is FALSE.
... Additional parameters to summary.

Details

This function is a method for the generic function summary() for class "ordinal.logistic.biplot". It can be invoked by calling summary(x) for an object x of the appropriate class.

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

See Also

OrdinalLogisticBiplot

Examples

data(LevelSatPhd)
olbo = OrdinalLogisticBiplot(LevelSatPhd,sFormula=NULL,numFactors=2,
coordinates="EM",penalization=0.2,show=FALSE)
summary(olbo)

summary.pordlogist Summary Method Function for Objects of Class 'pordlogist'

Description

This function shows a summary of the principal results for the estimation for individuals and variables, like number of observations, the number of iterations, the covariances matrix, some Pseudo R-squared indices with the correct classification percentage of each regression and the logLikelihood with "Estimate coefficients", "Std. Error", "z value" or "Pr(|z|)" values.

Usage

## S3 method for class 'pordlogist'
summary(object,...)

Arguments

object This parameter keeps the ordinal logistic biplot object for a variable.
... Additional parameters to summary.
Details

This function is a method for the generic function summary() for class "pordlogist". It can be invoked by calling summary(x) for an object x of the appropriate class.

Author(s)

Julio Cesar Hernandez Sanchez, Jose Luis Vicente-Villardon
Maintainer: Julio Cesar Hernandez Sanchez <juliocesar_avila@usal.es>

See Also

pordlogist, CheckDataSet, OrdinalLogBiplotEM

Examples

data(LevelSatPhd)
dataSet = CheckDataSet(LevelSatPhd)
datanom = dataSet$datanom
olb = OrdinalLogBiplotEM(datanom, dim = 2, nnodos = 10, tol = 0.001, maxiter = 100, penalization = 0.2)
model = pordlogist(datanom[, 1], olb$RowCoordinates, tol = 0.001, maxiter = 100, penalization = 0.2)
summary(model)
Index

*Topic **BiplotObject**
  GetOrdinalBiplotObjectPenal, 7

*Topic **Density**
  BiplotDensity, 3

*Topic **EM**
  OrdinalLogBiplotEM, 10

*Topic **MirtOrdinalObject**
  GetOrdinalBiplotObjectMIRT, 6

*Topic **algorithm**
  OrdinalLogBiplotEM, 10

*Topic **biplot**
  OrdinalLogisticBiplot, 11

*Topic **check**
  CheckDataSet, 4

*Topic **cluster**
  PlotClusterVariable, 16

*Topic **datasets**
  LevelSatPhd, 9

*Topic **data**
  CheckDataSet, 4

*Topic **logistic**
  OrdinalLogisticBiplot, 11

*Topic **mirt**
  EstimationRowsMIRT, 5

*Topic **models**
  OrdinalLogisticBiplot, 11

*Topic **package**
  OrdinalLogisticBiplot-package, 2

*Topic **plot**
  plot.ordinal.logistic.biplot, 14
  plotOrdBipCoeffVariable, 17
  plotOrdinalVariable, 19

*Topic **summary**
  summary.ordinal.logistic.biplot, 22
  summary.pordlogist, 23

BiplotDensity, 3

CheckDataSet, 4, 22, 24

EstimationRowsMIRT, 5, 7

GetOrdinalBiplotObjectMIRT, 6
GetOrdinalBiplotObjectPenal, 7, 7

LevelSatPhd, 9

OrdinalLogBiplotEM, 3, 9, 10, 14, 22, 24
OrdinalLogisticBiplot, 3, 7, 9, 11, 16, 23
OrdinalLogisticBiplot-package, 2

plot.ordinal.logistic.biplot, 14
PlotClusterVariable, 16
plotOrdBipCoeffVariable, 17
plotOrdinalVariable, 19
pordlogist, 11, 20, 24

summary.ordinal.logistic.biplot, 22
summary.pordlogist, 23