# Package ‘multigroup’

February 23, 2020

**Type** Package

**Title** Multigroup Data Analysis

**Version** 0.4.5

**Date** 2020-02-10

**Author** Aida Eslami, El Mostafa Qannari, Stephanie Bougeard, Gaston Sanchez
Questions and comments go to Aida Eslami <aida.eslami@yahoo.fr> and
Stephanie Bougeard <stephanie.bougeard@anses.fr>

**Maintainer** Aida Eslami <aida.eslami@yahoo.fr>

**Depends** R (>= 2.15.0)

**Imports** MASS

**Description** Multivariate analysis methods including principal component analysis,
partial least square regression, and multiblock analysis to describe,
summarize, and visualize data with a group structure.



**Suggests** testthat

**RoxygenNote** 7.0.2

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2020-02-23 17:50:05 UTC

## R topics documented:

---

`BGC` *Between Group Comparison*

---

## Description

Between Group Comparison (BGC)

## Usage

```
BGC(Data, Group, numc = NULL, ncomp = NULL, Scale = FALSE, graph = FALSE)
```

## Arguments

| | |
|---|---|
| `Data` | a numeric matrix or data frame |
| `Group` | a vector of factors associated with group structure |
| `numc` | number of components assocaited with PCA on each group |
| `ncomp` | number of components, if NULL number of components is equal to 2 |
| `Scale` | scaling variables, by defalt is FALSE. By default data are centered within groups |
| `graph` | should loading and component be plotted |

## Value

list with the following results:

| | |
|---|---|
| `Data` | Original data |
| `Con.Data` | Concatenated centered data |
| `split.Data` | Group centered data |
| `Group` | Group as a factor vector |
| `loadings.common` | Matrix of common loadings |
| `lambda` | The specific variances of groups |
| `exp.var` | Percentages of total variance recovered associated with each dimension |

### References

W. J. Krzanowski (1979). Between-groups comparison of principal components, *Journal of the American Statistical Association*, 74, 703-707.

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). General overview of methods of analysis of multi-group datasets, *Revue des Nouvelles Technologies de l'Information*, 25, 108-123.

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). Analyses factorielles de donnees structurees en groupes d'individus, *Journal de la Societe Francaise de Statistique*, 154(3), 44-57.

### See Also

mgPCA, FCPCA, DCCSWA, DSTATIS, DGPA, summarize, TBWvariance, loadingsplot, scoreplot, iris

### Examples

```
Data = iris[,-5]
Group = iris[,5]
res.BGC = BGC(Data, Group, graph=TRUE)
loadingsplot(res.BGC, axes=c(1,2))
scoreplot(res.BGC, axes=c(1,2))
```

---

## DCCSWA — *Dual Common Component and Specific Weights Analysis*

---

### Description

Dual Common Component and Specific Weights Analysis: to find common structure among variables of different groups

### Usage

```
DCCSWA(Data, Group, ncomp = NULL, Scale = FALSE, graph = FALSE)
```

### Arguments

| | |
|---|---|
| Data | a numeric matrix or data frame |
| Group | a vector of factors associated with group structure |
| ncomp | number of components, if NULL number of components is equal to 2 |
| Scale | scaling variables, by defalt is FALSE. By default data are centered within groups |
| graph | should loading and component be plotted |

## Value

list with the following results:

| | |
|---|---|
| Data | Original data |
| Con.Data | Concatenated centered data |
| split.Data | Group centered data |
| Group | Group as a factor vector |
| loadings.common | Matrix of common loadings |
| saliences | Each group having a specific contribution to the determination of this common space, namely the salience, for each dimension under study |
| lambda | The specific variances of groups |
| exp.var | Percentages of total variance recovered associated with each dimension |

## References

E. M. Qannari, P. Courcoux, and E. Vigneau (2001). Common components and specific weights analysis performed on preference data. *Food Quality and Preference*, 12(5-7), 365-368.

A. Eslami (2013). Multivariate data analysis of multi-group datasets: application to biology. University of Rennes I.

## See Also

mgPCA, FCPCA, BGC, DSTATIS, DGPA, summarize, TBWvariance, loadingsplot, scoreplot, iris

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res.DCCSWA = DCCSWA(Data, Group, graph=TRUE)
loadingsplot(res.DCCSWA, axes=c(1,2))
scoreplot(res.DCCSWA, axes=c(1,2))
```

---

# DGPA *Dual Generalized Procrustes Analysis*

---

## Description

Dual Generalized Procrustes Analysis to study multigroup data

## Usage

```
DGPA(Data, Group, ncomp = NULL, Scale = FALSE, graph = FALSE)
```

## Arguments

| | |
|---|---|
| `Data` | a numeric matrix or data frame |
| `Group` | a vector of factors associated with group structure |
| `ncomp` | number of components, if NULL number of components is equal to 2 |
| `Scale` | scaling variables, by defalt is FALSE. By default data are centered within groups |
| `graph` | should loading and component be plotted |

## Value

list with the following results:

| | |
|---|---|
| `Data` | Original data |
| `Con.Data` | Concatenated centered data |
| `split.Data` | Group centered data |
| `Group` | Group as a factor vector |
| `loadings.common` | Matrix of common loadings |
| `lambda` | The specific variances of groups |
| `exp.var` | Percentages of total variance recovered associated with each dimension |

## References


J. Gower (1975). Generalized procrustes analysis. *Psychometrika*, 40(1), 3-51.

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). General overview of methods of analysis of multi-group datasets, *Revue des Nouvelles Technologies de l'Information*, 25, 108-123.

@references A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). Analyses factorielles de donnees structurees en groupes d'individus, *Journal de la Societe Francaise de Statistique*, 154(3), 44-57.


## See Also

`mgPCA`, `FCPCA`, `DCCSWA`, `DSTATIS`, `BGC`, `summarize`, `TBWvariance`, `loadingsplot`, `scoreplot`, `iris`

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res.DGPA = DGPA(Data, Group, graph=TRUE)
loadingsplot(res.DGPA, axes=c(1,2))
scoreplot(res.DGPA, axes=c(1,2))
```

DSTATIS *Dual STATIS*

## Description

Dual STATIS

## Usage

```
DSTATIS(Data, Group, ncomp = NULL, Scale = FALSE, graph = FALSE)
```

## Arguments

| | |
|---|---|
| `Data` | a numeric matrix or data frame |
| `Group` | a vector of factors associated with group structure |
| `ncomp` | number of components, if NULL number of components is equal to 2 |
| `Scale` | scaling variables, by defalt is False. By default data are centered within groups. |
| `graph` | should loading and component be plotted |

## Value

list with the following results:

| | |
|---|---|
| `Data` | original data |
| `Con.Data` | Concatenated centered data |
| `split.Data` | Group centered data |
| `Group` | Group as a factor vector |
| `RV` | The RV coefficient matrix |
| `weights` | Vector of weights |
| `compromise.matrix` | Compromise variance-covariance matrix |
| `loadings.common` | Matrix of common loadings |
| `lambda` | The specific variances of group |

## References


C. Lavit (1988). *Analyse conjointe de tableaux quantitatifs*. Masson.

C. Lavit, Y. Escoufier, R. Sabatier and P. Traissac (1994). The ACT (STATIS method). *Computational Statistics & Data Analysis*, 18, 97-117.

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). General overview of methods of analysis of multi-group datasets, *Revue des Nouvelles Technologies de l'Information*, 25, 108-123.

## See Also

mgPCA, FCPCA, DCCSWA, BGC, DGPA, summarize, TBWvariance, loadingsplot, scoreplot, iris

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res.DSTATIS = DSTATIS(Data, Group, graph=TRUE)
loadingsplot(res.DSTATIS, axes=c(1,2))
scoreplot(res.DSTATIS, axes=c(1,2))
```

---

# FCPCA *Flury's Common Principal Component Analysis*

---

## Description

Common principal component Analysis

## Usage

```
FCPCA(Data, Group, Scale = FALSE, graph = FALSE)
```

## Arguments

| | |
|---|---|
| Data | a numeric matrix or data frame |
| Group | a vector of factors associated with group structure |
| Scale | scaling variables, by default is False. By default data are centered within groups. |
| graph | should loading and component be plotted |

## Value

list with the following results:

| | |
|---|---|
| Data | Original data |
| Con.Data | Concatenated centered data |
| split.Data | Group centered data |
| Group | Group as a factor vector |
| loadings.common | Matrix of common loadings |
| lambda | The specific variances of group |
| exp.var | Percentages of total variance recovered associated with each dimension |

## References

B. N. Flury (1984). Common principal components in k groups. *Journal of the American Statistical Association*, 79, 892-898.

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). General overview of methods of analysis of multi-group datasets, *Revue des Nouvelles Technologies de l'Information*, 25, 108-123.

## See Also

mgPCA, DGPA, DCCSWA, DSTATIS, BGC, summarize, TBWvariance, loadingsplot, scoreplot, iris

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res.FCPCA = FCPCA(Data, Group, graph=TRUE)
loadingsplot(res.FCPCA, axes=c(1,2))
scoreplot(res.FCPCA, axes=c(1,2))
```

---

`loadingsplot` *loadings plot*

---

## Description

plots of variables (loadings)

## Usage

```
loadingsplot(x, axes = c(1, 2), INERTIE = NULL, cex = NULL, font.lab = NULL)
```

## Arguments

| | |
|---|---|
| `x` | results of the proposed multigroup methods in the package |
| `axes` | a vector of two selected components |
| `INERTIE` | if there is information about inertia |
| `cex` | character expansion for text by default .85 |
| `font.lab` | type of font by default 3 |

## Value

loadings plot

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res.mgPCA = mgPCA(Data, Group, graph=TRUE)
loadingsplot(res.mgPCA, axes=c(1,2))
```

loadingsplotXY *loadings plot of X and Y*

## Description

plots of variables (loadings)

## Usage

```
loadingsplotXY(
  X,
  Y,
  axes = c(1, 2),
  INERTIE = NULL,
  cex = NULL,
  font.lab = NULL
)
```

## Arguments

| | |
|---|---|
| `X` | common loadingsassociated with X |
| `Y` | common loadingsassociated with Y |
| `axes` | a vector of two selected components |
| `INERTIE` | if there is information about inertia |
| `cex` | character expansion for text by default .85 |
| `font.lab` | type of font by default 3 |

## Value

loadings plot

## Examples

```
data(oliveoil)
DataX = oliveoil[,2:6]
DataY = oliveoil[,7:12]
Group = as.factor(oliveoil[,1])
res.mgPLS = mgPLS (DataX, DataY, Group)
X=res.mgPLS$loadings.commo$X; Y=res.mgPLS$loadings.commo$Y
loadingsplotXY(X, Y, axes=c(1,2), INERTIE=res.mgPLS$noncumper.inertiglobal)
```

mbmgPCA *multiblock and multigroup Principal Component Analysis*

## Description

multiblock and multigroup PCA (mbmgPCA)

## Usage

```
mbmgPCA(
  Data,
  Group,
  nBlock,
  Block.name = NULL,
  ncomp = NULL,
  niter = NULL,
  ScaleGroup = FALSE,
  ScaleDataA = FALSE,
  ScaleDataB = FALSE,
  norm = FALSE
)
```

## Arguments

| | |
|---|---|
| `Data` | a numeric (quantitative) matrix or data frame |
| `Group` | a vector of factors associated with group structure |
| `nBlock` | a vector of number of variables in each block |
| `Block.name` | vector of name of blocks |
| `ncomp` | number of components, if NULL number of components is equal to min(rank(Data), M-1) |
| `niter` | number of iteration, if NULL number of iteration is equal to 10 |
| `ScaleGroup` | scaling variables in each group and block, by defalt is FALSE |
| `ScaleDataA` | scaling variables in each block after group preprocessing, by defalt is FALSE |
| `ScaleDataB` | scaling variables in each block befor group preprocessing, by defalt is FALSE |
| `norm` | normalize each block, by defalt is FALSE |

## Value

list with the following results:

| | |
|---|---|
| `K.Data` | Block data |
| `concat.Data` | Concatenated data |
| `concat.block.Data` | Block concatenated data |

| | |
|---|---|
| res.iter | Result of iteration |
| CRIT.h | Maximization criterion for each diemnsion |
| CRIT | Maximization criterion |
| crit.group | Maximization criterion associated with each group |
| crit.block | Maximization criterion associated with each block |
| omega | Weight of each block in construction of common scores |
| block.common.loading | Common loadings for each block |
| block.group.loadings | Partial loadings for each block and group |
| similarity | Similarity among common and partial loadings for each block |
| global.scores | Global scores among blocks |
| block.scores | Scores for each block |
| block.group.scores | Scores for each block and group |
| block.scores | Scores for each block |
| global.expvar | Global explained variance |
| cum.exp.var.block.group | Cumulative explained variance for each block and group |

## References

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard, Under Review. Multivariate data analysis of multi-groups datasets. Application to sensory analysis, *Chemolab*, 25, 108-123.

## See Also

mgPCA

## Examples

```
data(wine)
Select=c(which(wine[,2]=="Env1"),which(wine[,2]=="Env2"),which(wine[,2]=="Reference"))
WineData = wine[Select,-c(1,2)]
Group <- as.factor(c(rep("Env1",7), rep("Env2",5), rep("Reference",7)))
nBlock <- c(5, 3, 10, 9)
BlockNames    <- c("Olfaction at rest", "Vision", "Olfaction  after shaking", "Taste")
res = mbmgPCA(Data = WineData, Group, nBlock , Block.name=BlockNames, ncomp=5)
```

mgPCA *Multigroup Principal Component Analysis*

## Description

Multigroup PCA algorithm (NIPALS for Multigroup PCA)

## Usage

```
mgPCA(Data, Group, ncomp = NULL, Scale = FALSE, graph = FALSE)
```

## Arguments

| | |
|---|---|
| `Data` | a numeric matrix or data frame |
| `Group` | a vector of factors associated with group structure |
| `ncomp` | number of components, if NULL number of components is equal to 2 |
| `Scale` | scaling variables, by defalt is FALSE. By default data are centered within groups |
| `graph` | should loading and component be plotted |

## Value

list with the following results:

| | |
|---|---|
| `Data` | Original data |
| `Con.Data` | Concatenated centered data |
| `split.Data` | Group centered data |
| `Group` | Group as a factor vector |
| `loadings.group` | Loadings associated with each group |
| `score.group` | Scores associated with each group |
| `loadings.common` | Matrix of common loadings |
| `score.Global` | Global scores |
| `cumper.inertigroup` | Cumulative percentage of group components inertia |
| `cumper.inertiglobal` | Cumulative percentage of global component inertia |
| `noncumper.inertiglobal` | Percentage of global component inertia |
| `lambda` | The specific variances of groups |
| `exp.var` | Percentages of total variance recovered associated with each dimension |
| `Similarity.Common.Group.load` | Cumulative similarity between group and common loadings |
| `Similarity.noncum.Common.Group.load` | NonCumulative similarity between group and common loadings |

## References

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). General overview of methods of analysis of multi-group datasets, *Revue des Nouvelles Technologies de l'Information*, 25, 108-123.

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). Analyses factorielles de donnces structurces en groupes d'individus, *Journal de la Societe Francaise de Statistique*, 154(3), 44-57.

## See Also

BGC, FCPCA, DCCSWA, DSTATIS, DGPA, summarize, TBWvariance, loadingsplot, scoreplot, iris

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res.mgPCA = mgPCA (Data, Group)
barplot(res.mgPCA$noncumper.inertiglobal)
#----------------
#Similarity index: group loadings are compared to the common structure (first dimension)
Xzero = rep(0, 3)
MIN = min(res.mgPCA$Similarity.noncum.Common.Group.load[[1]][-1, 1])-0.0005
XLAB = paste("Dim1, %",res.mgPCA$noncumper.inertiglobal[1])
plot(Xzero, res.mgPCA$Similarity.noncum.Common.Group.load[[1]][-1, 1], pch=15, ylim=c(MIN, 1),
main="Similarity between groups and common structure", xlab=XLAB, ylab="", xaxt="n")
abline(v=0)
abline(h=seq(MIN, 1, by=0.05), col="black", lty=3)
XX=res.mgPCA$Similarity.noncum.Common.Group.load[[1]][-1, 1, drop=FALSE]
text(Xzero, XX, labels=rownames(XX), pos=4)
#----------------
# Similarity index: group loadings are compared to the common structure (dimensions 1 and 2)
XX1=res.mgPCA$Similarity.noncum.Common.Group.load[[1]][-1, 1]
XX2=res.mgPCA$Similarity.noncum.Common.Group.load[[2]][-1, 1]
simil <- cbind(XX1, XX2)
YLAB = paste("Dim1, %",res.mgPCA$noncumper.inertiglobal[2])
plot(simil, xlab=XLAB, ylab=YLAB, main="Similarity between groups and common structure", pch=20)
text(simil, labels=rownames(simil), cex=1, font.lab=1, pos=3)
#------------------
loadingsplot(res.mgPCA, axes=c(1,2), INERTIE=res.mgPCA$noncumper.inertiglobal)
scoreplot(res.mgPCA, axes=c(1,2))
```

---

mgPLS *Multigroup Partial Least Squares Regression*

---

## Description

Multigroup PLS regression

### Usage

```
mgPLS(
  DataX,
  DataY,
  Group,
  ncomp = NULL,
  Scale = FALSE,
  Gcenter = FALSE,
  Gscale = FALSE
)
```

### Arguments

| | |
|---|---|
| `DataX` | a numeric matrix or data frame associated with independent dataset |
| `DataY` | a numeric matrix or data frame associated with dependent dataset |
| `Group` | a vector of factors associated with group structure |
| `ncomp` | number of components, if NULL number of components is equal to 2 |
| `Scale` | scaling variables, by defalt is FALSE. By default data are centered within groups |
| `Gcenter` | global variables centering, by defalt is FALSE. |
| `Gscale` | global variables scaling, by defalt is FALSE. |

### Value

list with the following results:

| | |
|---|---|
| `DataXm` | Group X data |
| `DataYm` | Group Y data |
| `Concat.X` | Concatenated X data |
| `Concat.Y` | Concatenated Y data |
| `coefficients` | Coefficients associated with X data |
| `coefficients.Y` | Coefficients associated with regressing Y on Global components X |
| `Components.Global` | Conctenated Components for X and Y |
| `Components.Group` | Components associated with groups in X and Y |
| `loadings.common` | Common vector of loadings for X and Y |
| `loadings.Group` | Group vector of loadings for X and Y |
| `expvar` | Explained variance associated with global components X |
| `cum.expvar.Group` | Cumulative explained varaince in groups of X and Y |
| `Similarity.Common.Group.load` | Cumulative similarity between group and common loadings |
| `Similarity.noncum.Common.Group.load` | NonCumulative similarity between group and common loadings |

### References

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). Multi-group PLS regressMathematics and Statistics, Springer Proceedings (ed), *New Perspectives in Partial Least Squares and Related Methods*, 56, 243-255.

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2014). Algorithms for multi-group PLS. *Journal of Chemometrics*, 28(3), 192-201.

### See Also

mgPCA, mbmgPCA

### Examples

```
data(oliveoil)
DataX = oliveoil[,2:6]
DataY = oliveoil[,7:12]
Group = as.factor(oliveoil[,1])
res.mgPLS = mgPLS (DataX, DataY, Group)
barplot(res.mgPLS$noncumper.inertiglobal)
#----- Regression coefficients
#res.mgPLS$coefficients[[2]]
#----- Similarity index: group loadings are compared to the common structure (in  X and Y spaces)
XX1= res.mgPLS$Similarity.noncum.Common.Group.load$X[[1]][-1, 1, drop=FALSE]
XX2=res.mgPLS$Similarity.noncum.Common.Group.load$X[[2]][-1, 1, drop=FALSE]
simX <- cbind(XX1, XX2)
YY1=res.mgPLS$Similarity.noncum.Common.Group.load$Y[[1]][-1, 1, drop=FALSE]
YY2=res.mgPLS$Similarity.noncum.Common.Group.load$Y[[2]][-1, 1, drop=FALSE]
simY <- cbind(YY1,YY2)
XLAB = paste("Dim1, %",res.mgPLS$noncumper.inertiglobal[1])
YLAB = paste("Dim1, %",res.mgPLS$noncumper.inertiglobal[2])
plot(simX[, 1], simX[, 2], pch=15, xlim=c(0, 1), ylim=c(0, 1),
                            main="Similarity indices in X space",
                            xlab=XLAB, ylab=YLAB)
abline(h=seq(0, 1, by=0.2), col="black", lty=3)
text(simX[, 1], simX[, 2], labels=rownames(simX), pos=2)
plot(simY[, 1], simY[, 2], pch=15, xlim=c(0, 1), ylim=c(0, 1),
                             main="Similarity indices in Y space",
                                       xlab=XLAB, ylab=YLAB)
abline(h=seq(0, 1, by=0.2), col="black", lty=3)
text(simY[, 1], simY[, 2], labels=rownames(simY), pos=2)
```

---

## multigroup  *multigroup: methods for multigroup data analysis*

---

### Description

This package includes several methods to study multigroup data, where the same set of variables are measured on different groups of individuals.

### Some Functions

multigroup provides a set of functions for multigroup analysis:

- `BGC`: Between Group Comparison
- `DCCSWA`: Dual Common Component and Specific Weights Analysis
- `DGPA`: Dual Generalized Procrustes Analysis
- `DSTATIS`: Dual STATIS
- `FCPCA`: Flury's Common Principal Component Analysis
- `mgPCA`:Multigroup Principal Component Analysis
- `mgPLS`: Multigroup Partial Least Squares Regression
- `mbmgPCA`: Multiblock and multigroup PCA

---

## `oliveoil` *Sensory and physico-chemical data of olive oils*

---

### Description

A data set with scores on 6 attributes from a sensory panel and measurements of 5 physico-chemical quality parameters on 16 olive oil samples. The first five oils are Greek, the next five are Italian and the last six are Spanish (Package pls).

### Usage

```
data(oliveoil)
```

### Format

A data frame with 16 observations on the following 2 variables. sensory a matrix with 6 columns. Scores for attributes yellow, green, brown, glossy, transp, and syrup. chemical a matrix with 5 columns. Measurements of acidity, peroxide, K232, K270, and DK (Package pls).

### Source

Package pls

---

`plot.mg` *Plots for multigroup objects*

---

## Description

plots of variables (loadings) and individuals (scores) if TRUE

## Usage

```
## S3 method for class 'mg'
plot(x, axes = c(1, 2), cex = NULL, font.lab = NULL, ...)
```

## Arguments

| | |
|---|---|
| `x` | results of multigroup method in the package |
| `axes` | by default the first two components |
| `cex` | character expansion for text by default .85 |
| `font.lab` | type of font by default 3 |
| `...` | Further arguments are ignored |

## Value

loadings and scores plots

---

`scoreplot` *Score plot for multigroup data*

---

## Description

plots of individuals

## Usage

```
scoreplot(x, axes = c(1, 2), cex = NULL, font.lab = NULL)
```

## Arguments

| | |
|---|---|
| `x` | results of the proposed multigroup methods in the package |
| `axes` | a vector of two selected components |
| `cex` | character expansion for text by default .85 |
| `font.lab` | type of font by default 3 |

## Value

score plot

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res.mgPCA = mgPCA (Data, Group, graph=TRUE)
scoreplot(res.mgPCA, axes=c(1,2))
```

---

summarize *Summary*

---

## Description

Summary of multigroup data in global and group parts

## Usage

```
summarize(Data, Group)
```

## Arguments

| | |
|---|---|
| Data | a numeric matrix or data frame |
| Group | a vector of factors associated with group structure |

## Value

list with the following results:

| | |
|---|---|
| Global.summary | summary of globala data |
| Group.summary | summary of group datasets |
| mean.between.data | matrix of Group mean |
| mean.within.data | matrix of group centered data |

## See Also

mgPCA, DGPA, DCCSWA, DSTATIS, BGC, TBWvariance, iris

## Examples

```
Data = iris[,-5]
Group = iris[,5]
res = summarize(Data, Group)
```

`TBWvariance` *Total, within- and between-group variances*

### Description

Calculation of total, within- and between-group variance-covariance matrices

### Usage

```
TBWvariance(Data, Group)
```

### Arguments

| | |
|---|---|
| `Data` | a numeric matrix or data frame |
| `Group` | a vector of factors associated with group structure |

### Value

list with the following results:

| | |
|---|---|
| `Within.Var` | within-group variance-covariance matrix |
| `Between.Var` | between-group variance-covariance matrix |
| `Total.Var` | total variance-covariance matrix |
| `Btween.per` | Within-group variance percentage |
| `Btween.per` | Between-group variance percentage |

### References

A. Eslami, E. M. Qannari, A. Kohler and S. Bougeard (2013). General overview of methods of analysis of multi-group datasets, *Revue des Nouvelles Technologies de l'Information*, 25, 108-123.

### See Also

`mgPCA`, `DGPA`, `DCCSWA`, `DSTATIS`, `BGC`, `summarize`, `iris`

### Examples

```
Data = iris[,-5]
Group = iris[,5]
res = TBWvariance(Data, Group)
```

`wine` *Wine data*

## Description

The data used here refer to 21 wines of Val de Loire.

## Usage

```
data(wine)
```

## Format

A data frame with 21 rows (the number of wines) and 31 columns: the first column corresponds to the label of origin, the second column corresponds to the soil, and the others correspond to sensory descriptors.

## Source

Centre de recherche INRA d'Angers, Package FactoMineR

# Index