Package ‘DFA.CANCOR’

November 16, 2024

Type Package

Title Linear Discriminant Function and Canonical Correlation Analysis

Version 0.3.6

Date 2024-11-16

Author Brian P. O'Connor [aut, cre]

Maintainer Brian P. O'Connor <brian.oconnor@ubc.ca>

Description Produces SPSS- and SAS-like output for linear discriminant function analysis and canonical correlation analysis. The methods are described in Manly & Alberto (2017, ISBN:9781498728966), Rencher (2002, ISBN:0-471-41889-7), and Tabachnik & Fidell (2019, ISBN:9780134790541).

Imports graphics, stats, utils, grDevices, BayesFactor, MASS, mvoutlier, MVN

LazyLoad yes

LazyData yes

NeedsCompilation no

Repository CRAN

Date/Publication 2024-11-16 21:50:02 UTC

Contents

`DFA.CANCOR-package` *DFA.CANCOR*

Description

Provides SPSS- and SAS-like output for linear discriminant function analysis (via the DFA function) and for canonical correlation analysis (via the CANCOR function), and for providing effect sizes and significance tests for pairwise group comparisons (via the GROUP.DIFFS function). There are also functions for assessing the assumptions of normality, linearity, and homogeneity of variances and covariances.

`CANCOR` *Canonical correlation analysis*

Description

Produces SPSS- and SAS-like output for canonical correlation analysis. Portions of the code were adapted from James Steiger (www.statpower.net).

Usage

```
CANCOR(data, set1, set2, plot, plotCV, plotcoefs, verbose)
```

Arguments

`data`	A dataframe where the rows are cases & the columns are the variables.
`set1`	The names of the continuous variables for the first set, e.g., set1 = c('varA', 'varB', 'varC').
`set2`	The names of the continuous variables for the second set, e.g., set2 = c('varD', 'varE', 'varF').
`plot`	Should a plot of the coefficients be produced? The options are: TRUE (default) or FALSE.
`plotCV`	The canonical variate number for the plot, e.g., plotCV = 1.
`plotcoefs`	The coefficient for the plots. The options are 'structure' (default) or 'standardized'.
`verbose`	Should detailed results be displayed in the console? The options are: TRUE (default) or FALSE.

Value

If verbose = TRUE, the displayed output includes Pearson correlations, multivariate significance tests, canonical function correlations and bivariate significance tests, raw canonical coefficients, structure coefficients, standardized coefficients, and a bar plot of the structure or standardized coefficients.

The returned output is a list with elements

`cancorrels`	canonical correlations and their significance tests
`mv_Wilks`	The Wilks' lambda multivariate test
`mv_Pillai`	The Pillai-Bartlett multivariate test
`mv_Hotelling`	The Lawley-Hotelling multivariate test
`mv_Roy`	Roy's greatest characteristic root multivariate test
`mv_BartlettV`	Bartlett's V multivariate significance test
`mv_Rao`	Rao's' multivariate significance test
`CoefRawSet1`	raw canonical coefficients for Set 1
`CoefRawSet2`	raw canonical coefficients for Set 2
`CoefStruct11`	structure coefficients for Set 1 variables with the Set 1 variates
`CoefStruct21`	structure coefficients for Set 2 variables with the Set 1 variates
`CoefStruct12`	structure coefficients for Set 1 variables with the Set 2 variates
`CoefStruct22`	structure coefficients for Set 2 variables with the Set 2 variates
`CoefStandSet1`	standardized coefficients for Set 1 variables
`CoefStandSet2`	standardized coefficients for Set 2 variables
`CorrelSet1`	Pearson correlations for Set 1
`CorrelSet2`	Pearson correlations for Set 2
`CorrelSet1n2`	Pearson correlations between Set 1 & Set 2
`set1_scores`	Canonical variate scores for Set 1
`set2_scores`	Canonical variate scores for Set 2

Author(s)

Brian P. O'Connor

References

Manly, B. F. J., & Alberto, J. A. (2017). *Multivariate statistical methods: A primer (4th Edition).* Chapman & Hall/CRC, Boca Raton, FL.

Rencher, A. C. (2002). *Methods of Multivariate Analysis* (2nd ed.). New York, NY: John Wiley & Sons.

Sherry, A., & Henson, R. K. (2005). Conducting and interpreting canonical correlation analysis in personality research: A user-friendly primer. *Journal of Personality Assessment, 84,* 37-48.

Steiger, J. (2019). *Canonical correlation analysis.*
www.statpower.net/Content/312/Lecture%20Slides/CanonicalCorrelation.pdf

Tabachnik, B. G., & Fidell, L. S. (2019). *Using multivariate statistics (7th ed.).* New York, NY: Pearson.

Examples

```
# data that simulate those from De Leo & Wulfert (2013)
CANCOR(data = data_CANCOR$DeLeo_2013,
       set1 = c('Tobacco_Use','Alcohol_Use','Illicit_Drug_Use','Gambling_Behavior',
                'Unprotected_Sex','CIAS_Total'),
       set2 = c('Impulsivity','Social_Interaction_Anxiety','Depression',
                'Social_Support','Intolerance_of_Deviance','Family_Morals',
                'Family_Conflict','Grade_Point_Average'),
       plot = TRUE, plotCV = 1, plotcoefs='structure',
       verbose = TRUE)



# data from Ho (2014, Chapter 17)
CANCOR(data = data_CANCOR$Ho_2014,
       set1 = c("willing_use","likely_use","intend_use","certain_use"),
       set2 = c("perceived_risk","perceived_severity","self_efficacy",
                "response_efficacy","maladaptive_coping","fear"),
       plot = 'yes', plotCV = 1)


# data from Rencher (2002, pp. 366, 369, 372)
CANCOR(data = data_CANCOR$Rencher_2002,
       set1 = c("y1","y2","y3"),
       set2 = c("x1","x2","x3","x1x2","x1x3","x2x3","x1sq","x2sq","x3sq"),
       plot = 'yes', plotCV = 1)


# data from Tabachnik & Fidell (2019, p. 451, 460)    small dataset
CANCOR(data = data_CANCOR$TabFid_2019_small,
       set1 = c('TS','TC'),
       set2 = c('BS','BC'),
       plot = TRUE, plotCV = 1, plotcoefs='structure',
       verbose = TRUE)


# data from Tabachnik & Fidell (2019, p. 463)     complete dataset
CANCOR(data = data_CANCOR$TabFid_2019_complete,
       set1 = c("esteem","control","attmar","attrole"),
       set2 = c("timedrs","attdrug","phyheal","menheal","druguse"),
       plot = TRUE, plotCV = 1, plotcoefs='structure',
       verbose = TRUE)
```

```
# UCLA dataset   https://stats.oarc.ucla.edu/r/dae/canonical-correlation-analysis/
CANCOR(data = data_CANCOR$UCLA,
       set1 = c("Locus_Control","Self_Concept","Motivation"),
       set2 = c("Read","Write","Math","Science","Sex"),
       plot = TRUE, plotCV = 1, plotcoefs='standardized',
       verbose = TRUE)
```

---

data_CANCOR *data_CANCOR*

Description

A list with example data that were used in various presentations of canonical correlation analysis

Usage

```
data(data_CANCOR)
```

Details

A list with the example data that were used in the following presentations of canonical correlation analysis: De Leo and Wulfert (2013), Ho (2014), Rencher (2002), Tabachnick and Fidell (2019), and by the UCLA statistics tutorial at https://stats.oarc.ucla.edu/r/dae/canonical-correlation-analysis/.

References

De Leo, J. A., & Wulfert, E. (2013). Problematic internet use and other risky behaviors in college students: An application of problem-behavior theory. *Psychology of Addictive Behaviors, 27(1),* 133-141.

Ho, R. (2014). *Handbook of univariate and multivariate data analysis with IBM SPSS.* Boca Raton, FL: CRC Press.

Rencher, A. (2002). *Methods of multivariate analysis* (2nd ed.). New York, NY: John Wiley & Sons.

Tabachnick, B. G., & Fidell, L. S. (2019). Chapter 16: Multiway frequency analysis. *Using multivariate statistics.* New York, NY: Pearson.

Examples

```
names(data_CANCOR)

head(data_CANCOR$DeLeo_2013)

head(data_CANCOR$Ho_2014)
```

```
head(data_CANCOR$Rencher_2002)

head(data_CANCOR$TabFid_2019_small)

head(data_CANCOR$TabFid_2019_complete)
```

`data_DFA` *data_DFA*

Description

A list with example data that were used in various presentations of discrimination function analysis

Usage

```
data(data_DFA)
```

Details

A list with the example data that were used in the following presentations of discrimination function analysis: Field (2012), Green and Salkind (2008), Ho (2014), Huberty and Olejnik (2006), Noursis (2012), Rencher (2002), Sherry (2006), and Tabachnick and Fidell (2019).

References

Field, A., Miles, J., & Field, Z. (2012). Chapter 18 Categorical data. *Discovering statistics using R.* Los Angeles, CA: Sage.

Green, S. B., & Salkind, N. J. (2008). Lesson 35: Discriminant analysis (pp. 300-311). In, *Using SPSS for Windows and Macintosh: Analyzing and understanding data.* New York, NY: Pearson.

Ho, R. (2014). *Handbook of univariate and multivariate data analysis with IBM SPSS.* Boca Raton, FL: CRC Press.

Huberty, C. J., & Olejnik, S. (2019). *Applied MANOVA and discriminant analysis* (2nd. ed.). New York, NY: John Wiley & Sons.

Noursis, M. J. (2012). *IBM SPSS Statistics 19 advanced statistical procedures companion.* Upper Saddle River, NJ: Prentice Hall.

Rencher, A. (2002). *Methods of multivariate analysis* (2nd ed.). New York, NY: John Wiley & Sons.

Sherry, A. (2006). Discriminant analysis in counseling research. *Counseling Psychologist, 34,* 661-683.

Tabachnick, B. G., & Fidell, L. S. (2019). Chapter 16: Multiway frequency analysis. *Using multivariate statistics.* New York, NY: Pearson.

Examples

```
names(data_DFA)

head(data_DFA$Field_2012)

head(data_DFA$Green_2008)

head(data_DFA$Ho_2014)

head(data_DFA$Huberty_2019_p45)

head(data_DFA$Huberty_2019_p285)

head(data_DFA$Norusis_2012)

head(data_DFA$Rencher_2002_football)

head(data_DFA$Rencher_2002_root)

head(data_DFA$Sherry_2006)

head(data_DFA$TabFid_2019_complete)

head(data_DFA$TabFid_2019_small)
```

`DFA` *Discriminant function analysis*

Description

Produces SPSS- and SAS-like output for linear discriminant function analysis.

Usage

```
DFA(data, groups, variables, plot, predictive, priorprob, covmat_type, CV, verbose)
```

Arguments

`data`	A dataframe where the rows are cases & the columns are the variables.
`groups`	The name of the groups variable in the dataframe, e.g., groups = 'Group'.
`variables`	The names of the continuous variables in the dataframe that will be used in the DFA, e.g., variables = c('varA', 'varB', 'varC').

`plot` Should a plot of the mean standardized discriminant function scores for the groups be produced? The options are: TRUE (default) or FALSE.

`predictive` Should a predictive DFA be conducted? The options are: TRUE (default) or FALSE.

`priorprob` If predictive = TRUE, how should the prior probabilities of the group sizes be computed? The options are:
'EQUAL' for equal group sizes; or
'SIZES' (default) for the group sizes to be based on the sizes of the groups in the dataframe.

`covmat_type` The kind of covariance to be used for a predictive DFA. The options are:
'within' (for the pooled within-groups covariance matrix, which is the default) or
'separate' (for separate-groups covariance matrices).

`CV` If predictive = TRUE, should cross-validation (leave-one-out cross-validation) analyses also be conducted? The options are: TRUE (default) or FALSE.

`verbose` Should detailed results be displayed in console? The options are: TRUE (default) or FALSE.

Details

The predictive DFA option using separate-groups covariance matrices (which is often called 'quadratic DFA') is conducted following the procedures described by Rencher (2002). The covariance matrices in this case are based on the scores on the continuous variables. In contrast, the 'separate-groups' option in SPSS involves use of the group scores on the discriminant functions (not the original continuous variables), which can produce different classifications.

When data has many cases (e.g., > 1000), the leave-one-out cross-validation analyses can be time-consuming to run. Set CV = FALSE to bypass the predictive DFA cross-validation analyses.

See the documentation below for the GROUP.DIFFS function for information on the interpretation of the Bayes factors and effect sizes that are produced for the group comparisons.

Value

If verbose = TRUE, the displayed output includes descriptive statistics for the groups, tests of univariate and multivariate normality, the results of tests of the homogeneity of the group variance-covariance matrices, eigenvalues & canonical correlations, Wilks' lambda & peel-down statistics, raw and standardized discriminant function coefficients, structure coefficients, functions at group centroids, one-way ANOVA tests of group differences in scores on each discriminant function, one-way ANOVA tests of group differences in scores on each original DV, significance tests for group differences on the original DVs according to Bird et al. (2014), a plot of the group means on the standardized discriminant functions, and extensive output from predictive discriminant function analyses (if requested).

The returned output is a list with elements

`evals` eigenvalues and canonical correlations

`mv_Wilks` The Wilks' lambda multivariate test

`mv_Pillai` The Pillai-Bartlett multivariate test

`mv_Hotelling`	The Lawley-Hotelling multivariate test
`mv_Roy`	Roy's greatest characteristic root multivariate test
`coefs_raw`	canonical discriminant function coefficients
`coefs_structure`	structure coefficients
`coefs_standardized`	standardized coefficients
`coefs_standardizedSPSS`	standardized coefficients from SPSS
`centroids`	unstandardized canonical discriminant functions evaluated at the group means
`centroidSDs`	group standard deviations on the unstandardized functions
`centroidsZ`	standardized canonical discriminant functions evaluated at the group means
`centroidSDsZ`	group standard deviations on the standardized functions
`dfa_scores`	scores on the discriminant functions
`anovaDFoutput`	One-way ANOVAs using the scores on a discriminant function as the DV
`anovaDVoutput`	One-way ANOVAs on the original DVs
`MFWER1.sigtest`	Significance tests when controlling the MFWER by (only) carrying out multiple t tests
`MFWER2.sigtest`	Significance tests for the two-stage approach to controling the MFWER
`classes_PRED`	The predicted group classifications
`classes_CV`	The classifications from leave-one-out cross-validations, if requested
`posteriors`	The posterior probabilities for the predicted group classifications
`grp_post_stats`	Group mean posterior classification probabilities
`classes_CV`	Classifications from leave-one-out cross-validations
`freqs_ORIG_PRED`	Cross-tabulation of the original and predicted group memberships
`chi_square_ORIG_PRED`	Chi-square test of independence
`PressQ_ORIG_PRED`	Press's Q significance test of classifiation accuracy for original vs. predicted group memberships
`kappas_ORIG_PRED`	Agreement (kappas) between the predicted and original group memberships
`PropOrigCorrect`	Proportion of original grouped cases correctly classified
`freqs_ORIG_CV`	Cross-Tabulation of the cross-validated and predicted group memberships
`chi_square_ORIG_CV`	Chi-square test of indepedence
`PressQ_ORIG_CV`	Press's Q significance test of classifiation accuracy for cross-validated vs. predicted group memberships
`kappas_ORIG_CV`	Agreement (kappas) between the cross-validated and original group memberships
`PropCrossValCorrect`	Proportion of cross-validated grouped cases correctly classified

Author(s)

Brian P. O'Connor

References

Bird, K. D., & Hadzi-Pavlovic, D. (2013). Controlling the maximum familywise Type I error rate in analyses of multivariate experiments. *Psychological Methods, 19(2),* p. 265-280.

Manly, B. F. J., & Alberto, J. A. (2017). *Multivariate statistical methods: A primer (4th Edition).* Chapman & Hall/CRC, Boca Raton, FL.

Rencher, A. C. (2002). *Methods of Multivariate Analysis* (2nd ed.). New York, NY: John Wiley & Sons.

Sherry, A. (2006). Discriminant analysis in counseling research. *Counseling Psychologist, 34,* 661-683.

Tabachnik, B. G., & Fidell, L. S. (2019). *Using multivariate statistics (7th ed.).* New York, NY: Pearson.

Examples

```
# data from Field et al. (2012, Chapter 16 MANOVA)
DFA_Field=DFA(data = data_DFA$Field_2012,
    groups = 'Group',
    variables = c('Actions','Thoughts'),
    predictive = TRUE,
    priorprob = 'EQUAL',
    covmat_type='within', # altho better to use 'separate' for these data
    verbose = TRUE)



# plots of posterior probabilities by group
# hoping to see correct separations between cases from different groups

# first, display the posterior probabilities
print(cbind(round(DFA_Field$posteriors[1:3],3), DFA_Field$posteriors[4]))

# group NT vs CBT
plot(DFA_Field$posteriors$posterior_NT, DFA_Field$posteriors$posterior_CBT,
     pch = 16, col = c('red', 'blue', 'green')[DFA_Field$posteriors$Group],
     xlim=c(0,1), ylim=c(0,1),
     main = 'DFA Posterior Probabilities by Original Group Memberships',
     xlab='Posterior Probability of Being in Group NT',
     ylab='Posterior Probability of Being in Group CBT' )
legend(x=.8, y=.99, c('CBT','BT','NT'), cex=1.2, col=c('red', 'blue', 'green'), pch=16, bty='n')

# group NT vs BT
plot(DFA_Field$posteriors$posterior_NT, DFA_Field$posteriors$posterior_BT,
     pch = 16, col = c('red', 'blue', 'green')[DFA_Field$posteriors$Group],
```

```
     xlim=c(0,1), ylim=c(0,1),
     main = 'DFA Posterior Probabilities by Group Membership',
     xlab='Posterior Probability of Being in Group NT',
     ylab='Posterior Probability of Being in Group BT' )
legend(x=.8, y=.99, c('CBT','BT','NT'), cex=1.2,col=c('red', 'blue', 'green'), pch=16, bty='n')

# group CBT vs BT
plot(DFA_Field$posteriors$posterior_CBT, DFA_Field$posteriors$posterior_BT,
     pch = 16, col = c('red', 'blue', 'green')[DFA_Field$posteriors$Group],
     xlim=c(0,1), ylim=c(0,1),
     main = 'DFA Posterior Probabilities by Group Membership',
     xlab='Posterior Probability of Being in Group CBT',
     ylab='Posterior Probability of Being in Group BT' )
legend(x=.8, y=.99, c('CBT','BT','NT'), cex=1.2, col=c('red', 'blue', 'green'), pch=16, bty='n')


# data from Green & Salkind (2008, Lesson 35)
DFA(data = data_DFA$Green_2008,
    groups = 'job_cat',
    variables = c('friendly','gpa','job_hist','job_test'),
    plot=TRUE,
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='within',
    CV=TRUE,
    verbose=TRUE)


# data from Ho (2014, Chapter 15)
# with group_1 as numeric
DFA(data = data_DFA$Ho_2014,
    groups = 'group_1_num',
    variables = c("fast_ris", "disresp", "sen_seek", "danger"),
    plot=TRUE,
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='within',
    CV=TRUE,
    verbose=TRUE)


# data from Ho (2014, Chapter 15)
# with group_1 as a factor
DFA(data = data_DFA$Ho_2014,
    groups = 'group_1_fac',
    variables = c("fast_ris", "disresp", "sen_seek", "danger"),
    plot=TRUE,
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='within',
    CV=TRUE,
    verbose=TRUE)
```

```
# data from Huberty (2006, p 45)
DFA_Huberty=DFA(data = data_DFA$Huberty_2019_p45,
    groups = 'treatmnt_S',
    variables = c('Y1','Y2'),
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='separate', # altho better to used 'separate' for these data
    verbose = TRUE)


# data from Huberty (2006, p 285)
DFA_Huberty=DFA(data = data_DFA$Huberty_2019_p285,
    groups = 'Grade',
    variables = c('counsum','gainsum','learnsum','qelib','qefac','qestacq',
                  'qeamt','qewrite','qesci'),
    predictive = TRUE,
    priorprob = 'EQUAL',
    covmat_type='within',
    verbose = TRUE)


# data from Norusis (2012, Chaper 15)
DFA_Norusis=DFA(data = data_DFA$Norusis_2012,
    groups = 'internet',
    variables = c('age','gender','income','kids','suburban','work','yearsed'),
    predictive = TRUE,
    priorprob = 'EQUAL',
    covmat_type='within',
    verbose = TRUE)


# data from Rencher (2002, p 170)  - rootstock
DFA(data = data_DFA$Rencher_2002_root,
    groups = 'rootstock',
    variables = c('girth4','ext4','girth15','weight15'),
    predictive = TRUE,
    priorprob = 'SIZES',
covmat_type='within',
verbose = TRUE)


# data from Rencher (2002, p 280)  - football
DFA(data = data_DFA$Rencher_2002_football,
    groups = 'grp',
    variables = c('WDIM','CIRCUM','FBEYE','EYEHD','EARHD','JAW'),
    predictive = TRUE,
    priorprob = 'SIZES',
covmat_type='separate',
verbose = TRUE)


# Sherry (2006)   -  with Group as numeric
```

```
DFA_Sherry <- DFA(data = data_DFA$Sherry_2006,
                  groups = 'Group_num',
                  variables = c('Neuroticism','Extroversion','Openness',
                                'Agreeableness','Conscientiousness'),
                  predictive = TRUE,
                  priorprob = 'SIZES',
                  covmat_type='separate',
                  verbose = TRUE)


# Sherry (2006)   -  with Group as a factor
DFA_Sherry <- DFA(data = data_DFA$Sherry_2006,
                  groups = 'Group_fac',
                  variables = c('Neuroticism','Extroversion','Openness',
                                'Agreeableness','Conscientiousness'),
                  predictive = TRUE,
                  priorprob = 'SIZES',
                  covmat_type='separate',
                  verbose = TRUE)

# plots of posterior probabilities by group
# hoping to see correct separations between cases from different groups

# first, display the posterior probabilities
print(cbind(round(DFA_Sherry$posteriors[1:3],3), DFA_Sherry$posteriors[4]))

# group 1 vs 2
plot(DFA_Sherry$posteriors$posterior_1, DFA_Sherry$posteriors$posterior_2,
     pch = 16, cex = 1, col = c('red', 'blue', 'green')[DFA_Sherry$posteriors$Group],
     xlim=c(0,1), ylim=c(0,1),
     main = 'DFA Posterior Probabilities by Original Group Memberships',
     xlab='Posterior Probability of Being in Group 1',
     ylab='Posterior Probability of Being in Group 2' )
legend(x=.8, y=.99, c('1','2','3'), cex=1.2, col=c('red', 'blue', 'green'), pch=16, bty='n')

# group 1 vs 3
plot(DFA_Sherry$posteriors$posterior_1, DFA_Sherry$posteriors$posterior_3,
     pch = 16, col = c('red', 'blue', 'green')[DFA_Sherry$posteriors$Group],
     xlim=c(0,1), ylim=c(0,1),
     main = 'DFA Posterior Probabilities by Group Membership',
     xlab='Posterior Probability of Being in Group 1',
     ylab='Posterior Probability of Being in Group 3' )
legend(x=.8, y=.99, c('1','2','3'), cex=1.2,col=c('red', 'blue', 'green'), pch=16, bty='n')

# group 2 vs 3
plot(DFA_Sherry$posteriors$posterior_2, DFA_Sherry$posteriors$posterior_3,
     pch = 16, col = c('red', 'blue', 'green')[DFA_Sherry$posteriors$Group],
     xlim=c(0,1), ylim=c(0,1),
     main = 'DFA Posterior Probabilities by Group Membership',
     xlab='Posterior Probability of Being in Group 2',
     ylab='Posterior Probability of Being in Group 3' )
legend(x=.8, y=.99, c('1','2','3'), cex=1.2, col=c('red', 'blue', 'green'), pch=16, bty='n')
```

```
# Tabachnik & Fiddel (2019, p 307, 311)   - small - with group as numeric
DFA(data = data_DFA$TabFid_2019_small,
    groups = 'group_num',
    variables = c('perf','info','verbexp','age'),
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='within',
    verbose = TRUE)


# Tabachnik & Fiddel (2019, p 307, 311)   - small - with group as a factor
DFA(data = data_DFA$TabFid_2019_small,
    groups = 'group_fac',
    variables = c('perf','info','verbexp','age'),
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='within',
    verbose = TRUE)


# Tabachnik & Fiddel (2019, p 324)   - complete  - with WORKSTAT as numeric
DFA(data = data_DFA$TabFid_2019_complete,
    groups = 'WORKSTAT_num',
    variables = c('CONTROL','ATTMAR','ATTROLE','ATTHOUSE'),
    plot=TRUE,
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='within',
    CV=TRUE,
    verbose=TRUE)

# Tabachnik & Fiddel (2019, p 324)   - complete  -  with WORKSTAT as a factor
DFA(data = data_DFA$TabFid_2019_complete,
    groups = 'WORKSTAT_fac',
    variables = c('CONTROL','ATTMAR','ATTROLE','ATTHOUSE'),
    plot=TRUE,
    predictive = TRUE,
    priorprob = 'SIZES',
    covmat_type='within',
    CV=TRUE,
    verbose=TRUE)
```

`GROUP.DIFFS` *Group Mean Differences on a Continuous Outcome Variable*

Description

Produces a variety of statistics for all possible pairwise independent groups comparisons of means on a continuous outcome variable.

Usage

```
GROUP.DIFFS(data, GROUPS=NULL, DV=NULL, var.equal=FALSE, p.adjust.method="holm",
            Ncomps=NULL, verbose=TRUE)
```

Arguments

`data` A dataframe where the rows are cases & the columns are the variables. If GROUPS and DV are not specified, then the GROUPS variable should be in the first column and the DV should be in the second column of data.

`GROUPS` The name of the groups variable in the dataframe, e.g., groups = 'Group'.

`DV` The name of the dependent (outcome) variable in the dataframe, e.g., DV = 'esteem'.

`var.equal` (from stats::t.test) A logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.

`p.adjust.method` The method to be used to adjust the p values for the number of comparisons. The options are "holm" (the default), "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none".

`Ncomps` The number of pairwise comparisons for the adjusted p values. If unspecified, it will be the number of all possible comparisons (i.e., the family-wise number of number of comparisons). Ncomps could alternatively be set to, e.g., the experiment-wise number of number of comparisons.

`verbose` Should detailed results be displayed in console?
The options are: TRUE (default) or FALSE.

Details

The function conducts all possible pairwise comparisons of the levels of the GROUPS variable on the continuous outcome variable. It supplements independent groups t-test results with effect size statistics and with the Bayes factor for each pairwise comparison.

The d values are the Cohen d effect sizes, i.e., the mean difference expressed in standard deviation units.

The g values are the Hedges g value corrections to the Cohen d effect sizes.

The r values are the effect sizes for the group mean difference expressed in the metric of Pearson's r.

The BESD values are the binomial effect size values for the group mean differences. The BESD casts the effect size in terms of the success rate for the implementation of a hypothetical procedure (e.g., the percentage of cases that were cured, or who died.) For example, an r = .32 is equivalent to increasing the success rate from 34% to 66% (or, possibly, reducing an illness or death rate from 66% to 34%).

The Bayes factor values are obtained from the ttest.tstat function in the BayesFactor package.

For example, a Bayes_Factor_alt_vs_null = 3 indicates that the data are 3 times *more* likely under the alternative hypothesis than under the null hypothesis. A Bayes_Factor_alt_vs_null = .2 indicates that the data are five times *less* likely under the alternative hypothesis than under the null hypothesis (1 / .2).

Conversely, a Bayes_Factor_null_vs_alt = 3 indicates that the data are 3 times *more* likely under the null hypothesis than under the alternative hypothesis. A Bayes_Factor_null_vs_alt = .2 indicates that the data are five times *less* likely under the null hypothesis than under the alternative hypothesis (1 / .2).

Value

If verbose = TRUE, the displayed output includes the means, standard deviations, and Ns for the groups, the t-test results for each pairwise comparison, the mean difference and its 95% confidence interval, four indices of effect size for each pairwise comparison (r, d, g, and BESD), and the Bayes factor. The returned output is a matrix with these values.

Author(s)

Brian P. O'Connor

References

Funder, D. C., & Ozer, D. J. (2019). Evaluating effect size in psychological research: Sense and nonsense. *Advances in Methods and Practices in Psychological Science, 2(2),* 156168.

Jarosz, A. F., & Wiley, J. (2014). What are the odds? A practical guide to computing and reporting Bayes factors. *Journal of Problem Solving, 7,* 29.

Randolph, J. & Edmondson, R.S. (2005). Using the binomial effect size display (BESD) to present the magnitude of effect sizes to the evaluation audience. *Practical Assessment Research & Evaluation, 10,* 14.

Rosenthal, R., Rosnow, R.L., & Rubin, D.R. (2000). *Contrasts and effect sizes in behavioral research: A correlational approach.* Cambridge UK: Cambridge University Press.

Rosenthal, R., & Rubin, D. B. (1982). A simple general purpose display of magnitude and experimental effect. *Journal of Educational Psychology, 74,* 166-169.

Rouder, J. N., Haaf, J. M., & Vandekerckhove, J. (2018). Bayesian inference for psychology, part IV: parameter estimation and Bayes factors. *Psychonomic Bulletin & Review, 25(1),* 102113.

Examples

```
GROUP.DIFFS(data_DFA$Field_2012, var.equal=FALSE, p.adjust.method="fdr")


GROUP.DIFFS(data = data_DFA$Sherry_2006, var.equal=FALSE, p.adjust.method="bonferroni")
```

GROUP.PROFILES *Group Profile Plots*

Description

Produces profile plots of group means for one or more continuous outcome variables.

Usage

```
GROUP.PROFILES(data, groups, variables,
        plot_type ='bar', bar_type = 'all',
        rescale='standardize',
        CI_level= 95, ylim=NULL,
        verbose=TRUE)
```

Arguments

`data`	A dataframe where the rows are cases and the columns are the variables.
`groups`	The name of the groups variable in data, e.g., groups = 'Group'.
`variables`	The name of the dependent (outcome) variable(s) in data, e.g., variables = c('esteem','anxiety').
`plot_type`	The options are 'bar' for bar plot, or 'profile' for a lines profile plot.
`bar_type`	When plot_type = 'bar', the options for bar_type are 'all', for placing the bar plots for all variables in one plot, or 'separate', for placing the bar plots for the variables in separate plots.
`rescale`	(optional) Should the variables be rescaled into a common metric? The options are 'no' (the default), 'standardize', or 'data', in which case rescaling will be done using the data variables (see the Details below).
`CI_level`	(optional) The confidence interval for the input, if provided (in whole numbers). The default is 95.
`ylim`	(optional) Limits for the y-axis, e.g., ylim = c(0, 5). Not implemented when multiple bar plots are requested.
`verbose`	(optional) Should detailed results be displayed in console? The options are: TRUE (default) or FALSE.

Details

The continuous 'variables' can be rescaled into the same metric, to facilitate interpretation when the means for multiple variables are placed on one plot. The variables can be standardized, or they can be rescaled using the minimum and maximum values in the data variables as the new range for the rescaled variables.

When plot_type = 'bar' and bar_type = 'separate', a maximum of four plots will be produced, for the first four 'variables'.

Value

If verbose = TRUE, the displayed output includes the means, standard deviations, Ns, and confidence intervals for the groups on the variables.

Author(s)

Brian P. O'Connor

Examples

```
GROUP.PROFILES(data = data_DFA$Ho_2014,
               groups = 'group_1_fac',
               variables = c("fast_ris", "disresp", "sen_seek", "danger"),
               rescale= 'data',
               plot_type ='bar',
               bar_type = 'separate')

#first run DFA
DFA_output <- DFA(data = data_DFA$Field_2012,
                  groups = 'Group',
                  variables = c('Actions','Thoughts'),
                  predictive = TRUE,
                  priorprob = 'EQUAL',
                  covmat_type='separate',
                  verbose = TRUE)

# then produce a profile plot of the group centroids on the discriminant functions
GROUP.PROFILES(data = DFA_output$dfa_scores,
               groups = 'group',
               variables = c('Function.1','Function.2'),
               rescale= 'no',
               plot_type ='profile',
               bar_type = 'separate')
```

HOMOGENEITY *Homogeneity of variances and covariances*

Description

Produces tests of the homogeneity of variances and covariances.

Usage

```
HOMOGENEITY(data, groups, variables, verbose)
```

Arguments

`data`	A dataframe where the rows are cases & the columns are the variables.
`groups`	(optional) The name of the groups variable in the dataframe (if there is one) e.g., groups = 'Group'.
`variables`	(optional) The names of the continuous variables in the dataframe for the analyses, e.g., variables = c('varA', 'varB', 'varC').
`verbose`	Should detailed results be displayed in the console? The options are: TRUE (default) or FALSE.

Value

If "variables" is specified, the analyses will be run on the "variables" in "data". If verbose = TRUE, the displayed output includes descriptive statistics and tests of univariate and multivariate homogeneity.

Bartlett's test compares the variances of k samples. The data must be normally distributed.

The non-parametric Fligner-Killeen test also compares the variances of k samples and it is robust when there are departures from normality.

Box's M test is a multivariate statistical test of the equality of multiple variance-covariance matrices. The test is prone to errors when the sample sizes are small or when the data do not meet model assumptions, especially the assumption of multivariate normality. For large samples, Box's M test may be too strict, indicating heterogeneity when the covariance matrices are not very different.

The returned output is a list with elements

`covmatrix`	The variance-covariance matrix for each group
`Bartlett`	Bartlett test of homogeneity of variances (parametric)
`Figner_Killeen`	Figner-Killeen test of homogeneity of variances (non parametric)
`PooledWithinCovarSPSS`	the pooled within groups covariance matrix from SPSS
`PooledWithinCorrelSPSS`	the pooled within groups correlation matrix from SPSS
`sscpWithin`	the within sums of squares and cross-products matrix
`sscpBetween`	the between sums of squares and cross-products matrix
`BoxLogdets`	the log determinants for Box's test
`BoxMtest`	Box's' test of the equality of covariance matrices

Author(s)

Brian P. O'Connor

References

Box, G. E. P. (1949). A general distribution theory for a class of likelihood criteria. *Biometrika, 36 (3-4),* 317-346.

Bartlett, M. S. (1937). Properties of sufficiency and statistical tests. *Proceedings of the Royal Society of London Series A 160,* 268-282.

Conover, W. J., Johnson, M. E., & Johnson, M. M. (1981). A comparative study of tests for homogeneity of variances, with applications to the outer continental shelf bidding data. *Technometrics, 23,* 351-361.

Warner, R. M. (2013). *Applied statistics: From bivariate through multivariate techniques.* Thousand Oaks, CA: SAGE.

Examples

```
# data from Field et al. (2012)
HOMOGENEITY(data = data_DFA$Field_2012,
            groups = 'Group', variables = c('Actions','Thoughts'))


# data from Sherry (2006)
HOMOGENEITY(data = data_DFA$Sherry_2006,
            groups = 'Group',
            variables = c('Neuroticism','Extroversion','Openness',
                          'Agreeableness','Conscientiousness'))
```

LINEARITY *Linearity*

Description

Provides tests of the possible linear and quadratic associations between two continuous variables.

Usage

```
LINEARITY(data, variables, groups, idvs, dv, verbose)
```

Arguments

`data`	A dataframe where the rows are cases & the columns are the variables.
`variables`	(optional) The names of the continuous variables in the dataframe for the analyses, e.g., variables = c('varA', 'varB', 'varC').
`groups`	(optional) The name of the groups variable in the dataframe (if there is one), e.g., groups = 'Group'.
`idvs`	(optional) The names of the predictor variables, e.g., variables = c('varA', 'varB', 'varC').
`dv`	(optional) The name of the dependent variable, if output for just one dependent variable is desired.
`verbose`	(optional) Should detailed results be displayed in the console? The options are: TRUE (default) or FALSE.

Value

If "variables" is specified, the analyses will be run on the "variables" in "data". If "groups" is specified, the analyses will be run for every value of "groups". If verbose = TRUE, the linear and quadratic regression coefficients and their statistical tests are displayed.

The returned output is a list with the regression coefficients and their statistical tests.

Author(s)

Brian P. O'Connor

References

Tabachnik, B. G., & Fidell, L. S. (2019). *Using multivariate statistics (7th ed.).* New York, NY: Pearson.

Examples

```
# data from Sherry (2006), using all variables
LINEARITY(data=data_DFA$Sherry_2006, groups='Group',
          variables=c('Neuroticism','Extroversion','Openness',
                      'Agreeableness','Conscientiousness') )


# data from Sherry (2006), specifying independent variables and a dependent variable
LINEARITY(data=data_DFA$Sherry_2006, groups='Group',
          idvs=c('Neuroticism','Extroversion','Openness','Agreeableness'),
          dv=c('Conscientiousness'),
          verbose=TRUE )

# data that simulate those from De Leo & Wulfert (2013)
LINEARITY(data=data_CANCOR$DeLeo_2013,
          variables=c('Tobacco_Use','Alcohol_Use','Illicit_Drug_Use',
                      'Gambling_Behavior', 'Unprotected_Sex','CIAS_Total',
                      'Impulsivity','Social_Interaction_Anxiety','Depression',
                      'Social_Support','Intolerance_of_Deviance','Family_Morals',
                      'Family_Conflict','Grade_Point_Average'),
          verbose=TRUE )
```

NORMALITY *Univariate and multivariate normality*

Description

Produces tests of univariate and multivariate normality using the MVN package.

Usage

```
NORMALITY(data, groups, variables, verbose)
```

Arguments

`data`	A dataframe or numeric matrix where the rows are cases & the columns are the variables.
`groups`	(optional) The name of the groups variable in the dataframe, e.g., groups = 'Group'.
`variables`	(optional) The names of the continuous variables in the dataframe for the analyses, e.g., variables = c('varA', 'varB', 'varC').
`verbose`	Should detailed results be displayed in the console? The options are: TRUE (default) or FALSE.

Details

If "groups" is not specified, the analyses will be run on all of the variables in "data". If "variables" is specified, the analyses will be run on the "variables" in "data". If "groups" is specified, the analyses will be run for every value of "groups". If verbose = TRUE, the displayed output includes descriptive statistics and tests of univariate and multivariate normality.

Value

The returned output is a list with the following elements:

`descriptives`	descriptive statistics, including skewness and kurtosis
`univariate_tests`	the univariate normality tests
`multivariate_tests`	the multivariate normality tests

Author(s)

Brian P. O'Connor

References

Doornik, J. A. & Hansen, H. (2008). An Omnibus test for univariate and multivariate normality. *Oxford Bulletin of Economics and Statistics 70,* 927-939.

Henze, N., & Wagner, T. (1997), A new approach to the BHEP tests for multivariate normality. *Journal of Multivariate Analysis, 62,* 1-23.

Johnson, R. A., & Wichern, D. W. (2007). *Applied Multivariate Statistical Analysis (3rd. ed.).* New Jersey, NJ: Prentice Hall.

Korkmaz, S., Goksuluk, D., Zararsiz, G. (2014). MVN: An R package for assessing multivariate normality. *The R Journal, 6(2),* 151-162.

Mardia, K. V. (1970), Measures of multivariate skewnees and kurtosis with applications. *Biometrika, 57(3),* 519-530.

Mardia, K. V. (1974), Applications of some measures of multivariate skewness and kurtosis for testing normality and robustness studies. *Sankhy A, 36,* 115-128.

Royston, J. P. (1992). Approximating the Shapiro-Wilk W-Test for non-normality. *Statistics and Computing, 2,* 117-119.

Shapiro, S., & Wilk, M. (1965). An analysis of variance test for normality. *Biometrika, 52,* 591611.

Szekely,G. J., & Rizzo, M. L. (2017). The energy of data. *Annual Review of Statistics and Its Application 4,* 447-79.

Tabachnik, B. G., & Fidell, L. S. (2019). *Using multivariate statistics (7th ed.).* New York, NY: Pearson.

Examples

```
# data that simulate those from De Leo & Wulfert (2013)
NORMALITY(data = na.omit(data_CANCOR$DeLeo_2013[c(
          'Unprotected_Sex','Tobacco_Use','Alcohol_Use','Illicit_Drug_Use',
          'Gambling_Behavior','CIAS_Total','Impulsivity','Social_Interaction_Anxiety',
          'Depression','Social_Support','Intolerance_of_Deviance','Family_Morals',
          'Family_Conflict','Grade_Point_Average')]))


# data from Field et al. (2012)
NORMALITY(data = data_DFA$Field_2012,
          groups = 'Group',
          variables = c('Actions','Thoughts'))

# data from Tabachnik & Fidell (2013, p. 589)
NORMALITY(data = na.omit(data_CANCOR$TabFid_2019_small[c('TS','TC','BS','BC')]))

# UCLA dataset
UCLA_CCA_data <- read.csv("https://stats.idre.ucla.edu/stat/data/mmreg.csv")
colnames(UCLA_CCA_data) <- c("LocusControl", "SelfConcept", "Motivation",
                             "read", "write", "math", "science", "female")
summary(UCLA_CCA_data)
NORMALITY(data = na.omit(UCLA_CCA_data[c("LocusControl","SelfConcept","Motivation",
                                         "read","write","math","science")]))
```

`OUTLIERS` *OUTLIERS*

Description

Provides tests and qqplots for multivariate outliers.

Usage

```
OUTLIERS(data, variables, ID=NULL, iterate=TRUE,
            alpha_univ=.05, plot_univariates=TRUE,
           MCD=TRUE, MCD.quantile = .75, alpha=0.025, cutoff_type = 'adjusted',
            qqplot=TRUE, plot_iters=NULL,
            verbose=TRUE)
```

Arguments

`data` A dataframe where the rows are cases & the columns are the variables.

`variables` The names of the continuous variables in the dataframe for the analyses, e.g., variables = c('varA', 'varB', 'varC').

`ID` (optional) The names of the case identification variable in data, if there is one. If ID is not specified, then the sequence of row numbers will be used as the case IDs.

`iterate` (optional) Should multiple iterations be conducted when searching for outliers? The options are: TRUE (default) or FALSE.

`alpha_univ` (optional) The p (alpha) level for univariate outliers. The default = .05.

`plot_univariates` (optional) Should univariate plots be provided?
The options are: TRUE (default) or FALSE.

`MCD` (optional) Should the Minimum Covariance Determinant method be used to compute the means and covariances?
The options are: TRUE (default) or FALSE.

`MCD.quantile` (optional) The MCD quantile, which is the the minimum number of the data points regarded as good points (MASS package). The default = .75, as recommended by Leys et al. (2018).

`alpha` (optional) alpha

`cutoff_type` (optional) The kind of cutoff to be computed. The options are adjusted' (the default) or 'quan'.

`qqplot` (optional) Should qqplots be provided?
The options are: TRUE (default) or FALSE.

`plot_iters` (optional) A vector with the iterations for the qqplot. For example, "plot_iters = c(1,2,6,7)" will produce a qqplot for each of iterations 1, 2, 6, and 7 on the output figure. The default is "plot_iters = c(1,2,3,4)".

`verbose` (optional) Should detailed results be displayed in console? TRUE (default) or FALSE

Details

This function provides both statistical and graphical methods of identifying multivariate outliers. Both methods are based on Mahalanobis distances.

A Mahalanobis distance is an estimate of how far each case is from the center of the joint distribution of the variables in multivariate space. Cases that are distant from the swarm of most other cases may be multivariate outliers.

Squared Mahalanobis distances have an approximate chi-squared distribution (when there is multivariate normality). Statistically, a multivariate outlier is said to exist when the squared Mahalanobis distance for a case is greater than a specified cut-off value that is derived from the chi-square distribution.

The computations for Mahalanobis distances are based on estimates of the means and covariances for the dataset. However, the means and covariances that are based on all of the data are affected by the existence of multivariate outliers. This renders the simple, whole-sample estimates of Mahalanobis distances, and thus the identification of outliers, problematic.

Better estimates of the means and covariances are obtained using the Minimum Covariance Determinant (MCD) method, which identifies the most central subset of the data. Mahalanobis distances are considered more "robust" when they are computed using the MCD means and covariances. The default for the **MCD argument** for this function is set to TRUE for this reason. Setting it to FALSE will result in the procedure using the whole-sample based means and covariances, which is not recommended.

Once obtained, Mahalanobis distances (robust or not) are assessed for statistical significance by comparing them with a specified quantile from the chi-squared distribution. There are two options for determining the specified quantile cutoff value. The simple, traditional approach is to use the alpha quantile of the chi-squared distribution with the degrees of freedom equal to the number of variables. In the present function, the default alpha threshold is 0.025.

A modern, alternative method of determining cutoff values is to use the adaptive reweighted estimator procedure (Filzmoser, Garrett, & Reimann, 2005), which derives a cutoff value that is appropriate for each specific dataset and sample size. These threshold values are called "adjusted quantiles".

The **cutoff_type argument** for this function can be set to "adjusted" for an adjusted quantile, or to "quan" for the traditional alpha quantile.

A "qqplot" of the squared Mahalanobis distances can be used to graphically assess multivariate normality and the existence of outliers. In this case, the (sorted) squared Mahalanobis distances are plotted against the corresponding quantiles of the chi-square distribution. When the the squared Mahalanobis distances fit the hypothesized distribution, the points in the Q-Q plot will fall on a straight, y = x line (chi-squared values are squared z scores). Deviations from the straight line suggest violations of multivariate normality and the possible existence of multivariate outliers.

The search for multivariate outliers can be conducted more than once for a given dataset. If outliers are identified on the first step (iteration), they can be removed from the dataset and another search for outliers can be conducted on the remaining data. It is not uncommon for multiple iterations to be required before no further outliers are found. Bigger outliers can mask smaller but still possibly important outliers. It is probably best to run the analyses for multiple iterations. In the present function, multiple iterations are conducted when the **iterate argument** is set to TRUE.

The present function provides up to four possible qqplots in the one-page output figure for a data analysis. By default, these plots will be for the first four interations that produced outliers. Use the **plot_iters argument** to produce plots from alternative iterations. For example, "plot_iters = c(1,2,6,7)" will place the qqplots from iterations 1, 2, 6, and 7 on the output figure.

Value

The returned output is a list with the outliers.

Author(s)

Brian P. O'Connor

References

Filzmoser, P., Garrett, R. G., & Reimann, C. (2005). Multivariate outlier detection in exploration geochemistry. *Computers & Geosciences, 31,* 579-587.

Leys, C., Klein, O., Dominicy, Y., & Ley, C. (2018). Detecting multivariate outliers: Use a robust variant of the Mahalanobis distance. *Journal of Experimental Social Psychology, 74,* 150-156.

Rodrigues, I. M., & Boente, G. (2011). Multivariate outliers. *International Encyclopedia of Statistical Science* (pp. 910-912). Berlin:Springer-Verlag.

Rousseeuw, P. J., & Leroy, A. M. (1987). *Robust Regression and Outlier Detection*. New York, NY: John Wiley & Sons.

Examples

```
OUTLIERS(data = iris, variables = c('Sepal.Length','Sepal.Width','Petal.Length'),
         ID=NULL, iterate=TRUE,
         alpha_univ=.05, plot_univariates=TRUE,
         MCD=TRUE, MCD.quantile = .75, alpha=0.025, cutoff_type = 'adjusted',
         qqplot=TRUE, plot_iters=c(1,2,5,6),
         verbose=TRUE)
```

PLOT_LINEARITY *Plot for linearity*

Description

Plots the linear, quadratic, and loess regression lines for the association between two continuous variables.

Usage

```
PLOT_LINEARITY(data, idv, dv, groups=NULL, groupNAME=NULL, legposition=NULL,
                     leginset=NULL, verbose=TRUE)
```

Arguments

`data`	A dataframe where the rows are cases & the columns are the variables.
`idv`	The name of the predictor variable.
`dv`	The name of the dependent variable.
`groups`	(optional) The name of the groups variable in the dataframe, e.g., groups = 'Group'.
`groupNAME`	(optional) The value (level, name, or number) from the groups variable that identifies the subset group whose data will be used for the analyses, e.g., groupNAME = 1.
`legposition`	(optional) The position of the legend, as specified by one of the following possible keywords: "bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright", "right" or "center".
`leginset`	(optional) The inset distance(s) of the legend from the margins as a fraction of the plot region when legend is placed by keyword.
`verbose`	Should detailed results be displayed in the console? The options are: TRUE (default) or FALSE.

Value

If verbose = TRUE, the linear and quadratic regression coefficients and their statistical tests are displayed.

The returned output is a list with the regression coefficients and the plot data.

Author(s)

Brian P. O'Connor

References

Tabachnik, B. G., & Fidell, L. S. (2019). *Using multivariate statistics (7th ed.).* New York, NY: Pearson.

Examples

```
# data that simulate those from De Leo & Wulfert (2013)
PLOT_LINEARITY(data=data_CANCOR$DeLeo_2013, groups=NULL,
               idv='Family_Conflict', dv='Grade_Point_Average', verbose=TRUE)


# data from Sherry (2006), ignoring the groups
PLOT_LINEARITY(data=data_DFA$Sherry_2006, groups=NULL, groupNAME=NULL,
               idv='Neuroticism', dv='Conscientiousness', verbose=TRUE)

# data from Sherry (2006), group 2 only
PLOT_LINEARITY(data=data_DFA$Sherry_2006, groups ='Group', groupNAME=2,
               idv='Neuroticism', dv='Conscientiousness', verbose=TRUE)
```

Index