Package 'holland'

July 22, 2025

Type Package

Title Statistics for Holland's Theory of Vocational Choice

Version 0.1.2-3

License GPL-3

LazyData true

Encoding UTF-8

Depends R (>= 3.5.0), stats, utils, MplusAutomation, mvtnorm

Date 2025-4-18

Maintainer Joerg-Henrik Heine <jhheine@googlemail.com>

Description Offers a convenient way to compute parameters
in the framework of the theory of vocational choice introduced by
J.L. Holland, (1997). A comprehensive summary to this theory
of vocational choice is given in Holland, J.L. (1997). Making vocational
choices. A theory of vocational personalities and work environments.
Lutz, FL: Psychological Assessment.

RoxygenNote 7.3.2

NeedsCompilation no

Author Joerg-Henrik Heine [aut, cre, com],
Florian Hartmann [aut, com],
Bernhard Ertl [rev, com],
Christian Tarnai [com, ths]

Repository CRAN

Date/Publication 2025-04-17 23:40:02 UTC

Contents

`AIST_2005_F_1270` *RIASEC correlations - female AIST norm sample*

Description

a matrix of correlation coefficients based on $n = 1270$ observations, obtaind from the german female norm sample of the AIST published in Bergman & Eder (2005).

Usage

```
data(AIST_2005_F_1270)
```

Format

A 6 x 6 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Bergmann, C.,& Eder, F. (2005). *AIST-R Allgemeiner Interessen-Struktur-Test - Revision*. Goettingen: Hogrefe.

Examples

```
data(AIST_2005_F_1270)
dim(AIST_2005_F_1270)
AIST_2005_F_1270
```

AIST_2005_M_1226 *RIASEC correlations - male AIST norm sample*

Description

a matrix of correlation coefficients based on n = 1226 observations, obtaind from the german male norm sample of the AIST published in Bergman & Eder (2005).

Usage

```
data(AIST_2005_M_1226)
```

Format

A 6 x 6 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Bergmann, C., & Eder, F. (2005). *AIST-R Allgemeiner Interessen-Struktur-Test - Revision*. Goettingen: Hogrefe.

Examples

```
data(AIST_2005_M_1226)
dim(AIST_2005_M_1226)
AIST_2005_M_1226
```

AIST_2019_F_2015 *RIASEC correlations - female AIST sample*

Description

a matrix of correlation coefficients based on $n = 2015$ observations, obtaind from the german female sample of the AIST published in Bergman & Eder (2019).

Usage

```
data(AIST_2019_F_2015)
```

Format

A 6 x 6 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Bergmann, C. & Eder, F. (2019). *AIST 3. Allgemeiner Interessen-Struktur-Test mit Umwelt-Struktur-Test (UST-3) – Version 3*. Goettingen: Hogrefe.

Examples

```
data(AIST_2019_F_2015)
dim(AIST_2019_F_2015)
AIST_2019_F_2015
```

AIST_2019_M_1661 *RIASEC correlations - male AIST sample*

Description

a matrix of correlation coefficients based on n = 1661 observations, obtaind from the german male sample of the AIST published in Bergman & Eder (2019).

Usage

```
data(AIST_2019_M_1661)
```

Format

A 6 x 6 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Bergmann, C. & Eder, F. (2019). *AIST 3. Allgemeiner Interessen-Struktur-Test mit Umwelt-Struktur-Test (UST-3) – Version 3*. Goettingen: Hogrefe.

Examples

```
data(AIST_2019_M_1661)
dim(AIST_2019_M_1661)
AIST_2019_M_1661
```

Circ_emp *Running Mplus for empirical RIASEC angular locations*

Description

This function generates and runs Mplus syntax which is discussed in Nagy et. al. (2009) and extracts the estimated model parameters - the angular locations of the six RIASEC dimensions, based on their correlations, in the asumed circumplex array within the theory of vocational choice by John Holland (1997).

Usage

```
Circ_emp(
  N,
  Cor,
  konstrukt = c("R", "I", "A", "S", "E", "C"),
  showOutput = TRUE,
  Mplus_command = "Mplus",
  replaceOutfile = "always",
  name = NULL,
  ...
)
```

Arguments

N	number of observations for correlations as numeric
Cor	either an R matrix object with RIASEC correlations or the name of a correlation matrix, stored as a .dat text file, as character. If parameter Cor is a matrix then correlation will be saved as .dat file in current workspace directory.
konstrukt	optionaly a character vector with length = 6 containing labels for construct dimensions - default is konstrukt = c("R","I","A","S","E","C").
showOutput	default set to TRUE - see description of the function runModels() in package MplusAutomation.
Mplus_command	default set to "Mplus" as in the default in MplusAutomation - see examples and see description of the function runModels() in package MplusAutomation.
replaceOutfile	default set to "always" - see description of the function runModels() in package MplusAutomation.
name	optional a name (as character) for the Mplus syntax to be saved - default is the object name or the name given in paramter Cor as character (with ending changed to '.inp').
...	additional parameters passed through

Details

more to come ...

Value

returns a list object containing the empirical RIASEC angular locations extracted from the Mplus result file after running the Mplus syntax.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
# generating runing and extracting parameters from Mplus files
# refering to a correlation data stored in object ‘example2'
data(example2)# loading fictional example correlation matrix
# not Run until Mplus is installed on your Sytem #####
## Not run: test <- Circ_emp(300,example2)
test
### ploting the result as a circumplex
plot(test)
# for black and white printing
plot(test,ltype=c(1,2),lcolor=c("grey","grey","black","black"))
## End(Not run)
```

`Circ_pro` *Running Mplus for empirical RIASEC and additional construct angular locations*

Description

This function generates and runs Mplus syntax which is discussed in Nagy et. al. (2009) and extracts the estimated model parameters - the angular locations of the six RIASEC dimensions and angular locations for the additional (projected) construct, based on their correlations, in the asumed circumplex array within the theory of vocational choice by John Holland (1997).

Usage

```
Circ_pro(
  N,
  Cor,
  M = dim(Cor)[1] - 6,
  showOutput = TRUE,
  Mplus_command = "Mplus",
  replaceOutfile = "always",
  ...
)
```

Arguments

`N`	number of observations for correlations as numeric
`Cor`	must be an R matrix object with RIASEC and additional construct correlations. The order of the entrys of the correlation matrix must start with the additional construct dimensions - thus the last six collums (rows) in the correlation matrix are the six RIASEC dimensions.
`M`	number of additional construct dimensions to project into the RIASEC circumplex - e.g. `M=5` for big-five personality dimensions - at default `M` is calculated from dimensions of `Cor`.

showOutput	default set to TRUE - see description of the function runModels() in package MplusAutomation.
Mplus_command	default set to "Mplus" as in the default in MplusAutomation - see examples and see description of the function runModels() in package MplusAutomation.
replaceOutfile	default set to "always" - see description of the function runModels() in package MplusAutomation.
...	additional parameters passed through

Details

This function uses the function extractModelParameters() in package MplusAutomation.

By deault the labels of the dimensions are taken from the column / row names of the matrix objekt given in Cor. If there are no named columns / rows, dimension lables are created.

more to come ...

Value

returns a list object containing the empirical RIASEC and additional construct angular locations extracted from the Mplus result file after running the Mplus syntax.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
# generating runing and extracting parameters from Mplus files
# refering to a R object (example4) with correlation data
data(example4) # loading fictional example correlation matrix
# not Run until Mplus is installed on your Sytem #####
## Not run: test <- Circ_pro(300,example4)
test
### ploting the result as a circumplex
plot(test)
# for black and white printing
plot(test,ltype=c(1,2),lcolor=c("grey","grey","black","black"))
## End(Not run)
```

Circ_test *Running Mplus for testing empirical RIASEC angular locations*

Description

This function generates and runs Mplus syntax which is discussed in Nagy et. al. (2009) and extracts the estimated model fit - tested against given (perfect - as default) angular locations of the six RIASEC dimensions, based on their correlations, in the asumed circumplex array within the theory of vocational choice by John Holland (1997).

Usage

```
Circ_test(
  N,
  Cor,
  test = "perfect",
  w = FALSE,
  showOutput = TRUE,
  Mplus_command = "Mplus",
  replaceOutfile = "always",
  name = NULL,
  ...
)
```

Arguments

N	number of observations for correlations as numeric
Cor	either an R matrix object with RIASEC correlations or the name of a correlation matrix, stored as a .dat text file, as character (see details).
test	either character (default test="perfect"), whicht tests against a perfect circumplex array, or a numeric vector with length = 6 giving the six angular locations (in radians) to test against.
w	logical - write fit coefficiants as csv table? - default: w = FALSE.
showOutput	default set to TRUE - see description of the function runModels() in package MplusAutomation.
Mplus_command	default set to "Mplus" as in the default in MplusAutomation - see examples and see description of the function runModels() in package MplusAutomation.
replaceOutfile	default set to "always" - see description of the function runModels() in package MplusAutomation.
name	optional a name (as character) for the Mplus syntax to be saved - default is the object name or the name given in paramter Cor as character (with ending changed to '.inp').
...	additional parameters passed through

Details

more to come ...

Value

a list with coefficients for model fit extracted from the Mplus result file after running the Mplus syntax.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
# generating running and extraction of Mplus files
# refering to a correlation data stored object example2
data(example2) # loading fictional example correlation matrix
# not Run until Mplus is installed on your System #####
## Not run: test <- Circ_test(N = 300, Cor = example2, test="perfect")
test
## End(Not run)
```

---

con_brown_c_holland *Congruence Index c according to Brown & Gore (1994)*

Description

This function computes an index od congruence according to Brown & Gore (1994).

Usage

```
con_brown_c_holland(a, b)
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.

Details

The function finds the congruence according to Brown & Gore (1994) between the three-letter Holland-codes given in argument a, which is the person code, and argument b, which is the environment code. The Index is (currently) only defined for three letters from the Holland code. The degree of congruence is output, according to its definition by Brown & Gore (1994), as a reciprocal value of a distance. This means, for example, that a value of '18' is the result for a perfect fit !

Value

a numeric with value for congruence.

References

Brown & Gore (1994). An Evaluation of interest congruence indices: Distribution Characteristics and Measurement Properties. *Journal of Vocational Behaviour, 45*, 310-327.

Examples

```
con_brown_c_holland(a="RIA",b="SEC") # max. difference
con_brown_c_holland(a="RIA",b="RIA") # max. similarity
```

---

con_compindex_holland *Compatibility Index according to Wiggins & Moody (1981)*

Description

The function computes the three-letter *Compatibility* index according to Wiggins & Moody (1981).

Usage

```
con_compindex_holland(a, b)
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.

Details

The function finds the congruence according to Wiggins & Moody (1981) between the three-letter Holland codes given in argument a, which is the person code, and argument b, which is the environment code. The Index as defined by Wiggins & Moody (1981) targets (only) three letters from the Holland code. The degree of congruence is output, according to its definition by Wiggins & Moody (1981), as a reciprocal value of a distance. This means, for example, that a value of '8' is the result for a perfect fit of two three-letter codes !

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Wiggins, J.D.; Moody, A. (1981). Compatibility index description; Training and Associates: Dover, DE.

Examples

```
con_compindex_holland(a="RIA",b="SEC") # max. difference
con_compindex_holland(a="RIA",b="RIA") # max. similarity
```

---

con_hamming_holland *Congruence Index based on the Hamming distance*

Description

The function computes the location-weighted, cost-sensitive Hammig distance (Hamming, 1950).

Usage

```
con_hamming_holland(a, b, costs = "hexa", weights = c(1.5, 1.25, 1))
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.
costs	character with default set to `costs = "hexa"` to use a matrix with replacement costs based on the RIASEC structure for the calculation of the distance.
weights	a numeric vector with length equal to a and b assigning a weight to the places (1:6) of the letter-codes defined in a and b.

Details

The function finds the distance according to Hamming (1950) between two sequences (see Abbott, 1995), which are the Holland codes given in argument a, which is the person code, and argument b, which is the environment code.

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Hamming, R. (1950). Error detecting and error correcting codes. *Bell System Technical Journal, The, 29*(2), 147–160.

Abbott, A. (1995, August). Sequence Analysis: New Methods for Old Ideas. *Annual Review of Sociology*, 21, 93–113.

Examples

```
con_hamming_holland(a="RIA",b="SEC") # max. difference
con_hamming_holland(a="RIA",b="RIA") # max. similarity
con_hamming_holland(a="RIASEC",b="SECRIA", weights=c(1.5,1.25,1,0.75,0.5,0.25)) # max. difference
```

---

con_iachan_holland *Congruence Index according to Iachan (1984)*

Description

The function computes the congruence index according to Iachan (1984).

Usage

```
con_iachan_holland(a, b)
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.

Details

The function finds the congruence according to Iachan (1984) between the three-letter Holland-codes given in argument a, which is the person code, and argument b, which is the environment code. The Index is defined for three letters from the Holland code. The degree of congruence is output, according to its definition by Iachan (1984), as a reciprocal value of a distance. This means, for example, that a value of '28' is the result for a perfect fit !

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Iachan, R. (1984). A measure of agreement for use with the Holland classification system. *Journal of Vocational Behavior, 24* (2), 133–141.

Examples

```
con_iachan_holland(a="RIA",b="SEC") # max. difference
con_iachan_holland(a="RIA",b="RIA") # max. similarity
```

`con_levenshtein_holland` *Congruence Index based on the Levenshtein distance*

Description

The function computes the location-weighted, cost-sensitive (referring to the hexagon relationships) Levenshtein distance (Levenshtein, 1966) see also Needleman & Wunsch (1970).

Usage

```
con_levenshtein_holland(a, b, costs = "hexa", weights = c(1.5, 1.25, 1))
```

Arguments

`a`	a character vector with person Holland codes.
`b`	a character vector with environment Holland codes.
`costs`	character with default set to `costs = "hexa"` to use a matrix with replacement costs based on the RIASEC structure for the calculation of the distance.
`weights`	a numeric vector with length equal to `a` and `b` assigning a weight to the places (1:6) of the letter-codes defined in `a` and `b`.

Details

The function finds the distance according to Levenshtein (1966) between two sequences (see Abbott, 1995), which are the Holland codes given in argument a, which is the person code, and argument b, which is the environment code. Computational details can be found in Needleman & Wunsch, (1970).

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Levenshtein, V. I. (1966). Binary Codes Capable of Correcting Deletions, *Insertions and Reversals. Soviet Physics Doklady*, 10, 707.

Abbott, A. (1995, August). Sequence Analysis: New Methods for Old Ideas. *Annual Review of Sociology*, 21, 93–113.

Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology, 48*(3), 443–453. http://doi.org/10.1016/0022-2836(70)90057-4

Examples

```
con_levenshtein_holland(a="RIA",b="SEC") # max. difference
con_levenshtein_holland(a="RIA",b="RIA") # max. similarity
# with 6 characters in Holland-code
w <- c(1.5,1.25,1,0.75,0.5,0.25)
con_levenshtein_holland(a="RIASEC",b="SECRIA", weights=w) # max. difference
```

---

con_n3_holland *Congruence Index according to Joerin Fux (2005)*

Description

The function computes the congruence index according to Joerin Fux (2005).

Usage

```
con_n3_holland(a, b)
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.

Details

The function finds the congruence according to Joerin Fux (2005) between the up to six-letter Holland-codes given in argument a, which is the person code, and argument b, which is the environment code. The Index as defined by Jörin (2005) targets (only) three letters from the Holland code. The degree of congruence is output, according to its definition by Joerin Fux (2005), as a reciprocal value of a distance. This means, for example, that a value of '3' is the result for a perfect fit of two three-letter codes !

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Joerin Fux, S. (2005). *Persönlichkeit und Berufstätigkeit: Theorie und Instrumente von John Holland im deutschsprachigen Raum, unter Adaptation und Weiterentwicklung von Self-directed Search (SDS) und Position Classification Inventory (PCI)*. 1. Aufl. Göttingen: Cuvillier.

Examples

```
con_n3_holland(a="RIA",b="SEC") # max. difference
con_n3_holland(a="RIA",b="RIA") # max. similarity
```

`con_oneletter_holland` *Congruence Index according to Holland (1963)*

Description

The function computes the one letter congruence index according to Holland (1963).

Usage

```
con_oneletter_holland(a, b, hexadist = FALSE, letter = 1)
```

Arguments

`a`	a character vector with person Holland codes.
`b`	a character vector with environment Holland codes.
`hexadist`	logical with default set to `hexadist = FALSE`. If set to `hexadist = TRUE` the spacial distances in the hexagon are considered for the calculation of the first letter congruence.
`letter`	a integer indicating the position of the letter to be used.

Details

The function finds the congruence according to Holland (1963) between the Holland codes given in argument a, which is the person code, and argument b, which is the environment code. The Index is based on one letter from the Holland code, which is, according to Holland (1963) typically the first letter. But this can be changed to using any of the six letters, see argument `letter`. The degree of congruence is output as distance. This means, for example, that a value of '0' is the result for a perfect fit ! The function offers via the argument `hexadist` the extension to consider the spatial distance of the six dimensions in the hexagon for the calculation of the congruence (cf. Bowles, S. M., & Tunick, R. H. 2008).

Value

a numeric with value for congruence.

References

Holland, J.L. (1963). Explorations of a theory of vocational choice and achievement: II. A four-year prediction study. *Psychological Reports, 12*, 547–594

Bowles, S. M. & Tunick, R. H. (2008). *Is Congruence Dead? An Examination of the Correlation Between Holland's Congruence and Job Satisfaction Using Improved Methodology.* Morgantown, West Virginia: West Virginia University.

Examples

```
con_oneletter_holland(a="RIASEC",b="AIRCES")
con_oneletter_holland(a="RIASEC",b="AIRCES",hexadist=TRUE)
con_oneletter_holland(a="RIASEC",b="AIRCES",letter=2)
con_oneletter_holland(a="RIASEC",b="AIRCES",letter=6)
```

---

con_threeletter_holland

Congruence Index according to Wolfe & Betz (1981)

Description

The function computes the three-letter congruence index according to Wolfe & Betz (1981).

Usage

```
con_threeletter_holland(a, b)
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.

Details

The function finds the congruence according to Wolfe & Betz (1981) between the three-letter Holland codes given in argument a, which is the person code, and argument b, which is the environment code. The Index as defined by Wolfe & Betz (1981) targets (only) three letters from the Holland code. The degree of congruence is output, according to its definition by Wolfe & Betz (1981), as a reciprocal value of a distance. This means, for example, that a value of '2' is the result for a perfect fit of two three-letter codes !

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Wolfe, L. K. & Betz, N. E. (1981). Traditionality of choice and sex-role identification as moderators of the congruence of occupational choice in college women. *Journal of Vocational Behavior, 18*(1), 43–55. https://doi.org/10.1016/0001-8791(81)90028-2

Examples

```
con_threeletter_holland(a="RIA",b="SEC") # max. difference
con_threeletter_holland(a="RIA",b="RIA") # max. similarity
```

con_twoletter_holland *Congruence Index according to Healy & Mourton (1983)*

Description

The function computes the two-letter congruence index according to Healy & Mourton (1983).

Usage

```
con_twoletter_holland(a, b)
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.

Details

The function finds the congruence according to Healy & Mourton (1983) between the two-letter Holland codes given in argument a, which is the person code, and argument b, which is the environment code. The Index as defined by Healy & Mourton (1983) targets (only) two letters from the Holland code. The degree of congruence is output, according to its definition by Healy & Mourton (1983), as a reciprocal value of a distance. This means, for example, that a value of '3' is the result for a perfect fit of two two-letter codes !

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Healy, C. C. & Mourton, D. L. (1983). Derivatives of the Self-Directed Search: Potential clinical and evaluative uses. *Journal of Vocational Behavior, 23*(3), 318–328. https://doi.org/10.1016/0001-8791(83)90045-3

Examples

```
con_twoletter_holland(a="RI",b="SE") # max. difference
con_twoletter_holland(a="RI",b="RI") # max. similarity
```

con_zs_holland *Congruence Index according to Zener & Schnuelle (1976)*

Description

The function computes the congruence index according to Zener & Schnuelle (1976).

Usage

```
con_zs_holland(a, b)
```

Arguments

a	a character vector with person Holland codes.
b	a character vector with environment Holland codes.

Details

The function finds the congruence according to Zener & Schnuelle (1976) between the three-letter Holland-codes given in argument a, which is the person code, and argument b, which is the environment code. The Index as defined by Zener & Schnuelle (1976) targets (only) three letters from the Holland code. The degree of congruence is output, according to its definition by Zener & Schnuelle (1976), as a reciprocal value of a distance. This means, for example, that a value of '6' is the result for a perfect fit of two three-letter codes !

Value

a numeric with value for congruence.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Zener, T. B. & Schnuelle, L. (1976). Effects of the self-directed search on high school students. *Journal of Counseling Psychology, 23*(4), 353–359.

Examples

```
con_zs_holland(a="RIA",b="SEC") # max. difference
con_zs_holland(a="RIA",b="RIA") # max. similarity
```

deg *radians to degrees*

Description

This function converts angular locations from radians to degrees

Usage

```
deg(x, m = FALSE, rev = FALSE)
```

Arguments

x	numeric values in radians
m	logical should values > (2 * 3.14159265358979323846264338327 9) be divided modulo
rev	logical if TRUE list objekt is returnd with number of revolutions

Details

no details

Value

values in degrees; optionaly number of revolutions

Examples

```
## RIASEC angular locations in radians to degrees
deg(c(0, 1.047198, 2.094395, 3.141593, 4.188790, 5.235988))
deg(6.283185)
deg(6.283185,TRUE)
deg(12.56637)
deg(12.56637,TRUE)
# 14.137167 radians is two full revolutions and a quater
# which is 90 degrees or 1.570796 radians- check it!
deg(14.137167,TRUE,TRUE)
```

`dif_7_holland` *Seven differentiation indices for Holland profiles*

Description

The function computes seven differentiation indices for Holland profiles as cited in Bergman (1993) and Eder (1998).

Usage

```
dif_7_holland(A, ind = c("DI1", "DI2", "DI3", "DI4", "DI5", "DI6", "DI7"))
```

Arguments

`A`	a numeric vector with Holland score values for the interest profile of length = 6.
`ind`	a character indicating which index (see table 1) to return.

Details

The function finds seven different (see argument `ind`) differentiation indices as cited Bergman (1993) and Eder (1998) for the Holland-interest profile given in argument A, which is the person interest profile consisting of six values (either raw scores or norms) for each of the six dimensions of vocational interests.

specific information on the indices of differentiation:

Table 1: Differentiation indices for Holland profiles.
Source: Bergmann, (1993, p. 267).

Index	Brief description	Author / Source
DI1	Difference between highest and second highest interest score	(Frantz & Walsh, 1972)
DI2	Difference between highest and third highest interest score	(Spokane & Walsh, I978)
DI3	Difference between highest score and the average of the second and fourth highest score	(Iachan, 1984)
DI4	Difference between highest score and the average of the third and fifth highest score	(Iachan, 1984)
DI5	Difference between highest and lowest score	(Holland, 1973)
DI6	Difference between highest and lowest score, standardized by the overall level of interest	(Peiser & Meir,1978)
DI7	Dispersion of interest scores	(Healy & Mourton, 1983)

Value

a numeric with value for differentiation.

References

Holland, J.L. 1963. A theory of vocational choice. I. Vocational images and choice. *Vocational Guidance Quarterly, 11*(4), 232–239.

Bergmann, C. (1993). Differenziertheit der Interessen und berufliche Entwicklung. *Zeitschrift für Differentielle und Diagnostische Psychologie, 14*(4), 265–279.

Frantz, T. T. & Walsh, E. P. (1972). Exploration of Holland's theory of vocational choice in graduate school environments. *Journal of Vocational Behaviour*, 2, 223-232.

Spokane, A. R. & Walsh, W. B. (1978). Occupational level and Holland's theory for employed men and women. *Journal of Vocational Behaviour*, 12, 145-154.

Iachan, R. (1984). A family of differentiation indices. *Psychometrika*, 49, 217-222.

Holland, J. L. (1973). *Making vocational choices*. Englewood Cliffs, New Jersey: Prentice Hall Inc.

Peiser, C. & Meir, E. I. (1978). Congruency, consistency, and differentiation of vocational interests as predictors of vocational satisfaction and preference stability. *Journal of Vocational Behaviour*, 12, 270-278.

Healy, C. C. & Mourton, D. L. (1983). Derivatives of the Self-Directed Search: Potential clinical and evaluative uses. *Journal of Vocational Behavior, 23*(3), 318–328. https://doi.org/10.1016/0001-8791(83)90045-3

Eder, F. (1998). Differenziertheit der Interessen als Prädiktor der Interessenentwicklung. In J. Abel & C. Tarnai (Hrsg.), *Pädagogisch-psychologische Interessenforschung in Studium und Beruf* (S. 63–77). Münster: Waxmann.

Examples

```
# fictional interest profile:
A <- c(70, 90, 120, 75, 100, 130)
names(A) <- c("R","I","A","S","E","C")

# differentiation according to Frantz & Walsh (1972)
dif_7_holland(A, ind = "DI1")

# all of the differentiation indices
ind <- c("DI1","DI2","DI3","DI4","DI5","DI6","DI7")
sapply(ind, function(x)dif_7_holland(A,x),USE.NAMES = FALSE)
```

---

example1 *RIASEC correlations - perfect*

Description

fictional correlation matrix representing a perfect RIASEC circumplex

Usage

```
data(example1)
```

Format

A 6 x 6 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Examples

```
data(example1)
dim(example1)
example1
```

---

example2 *RIASEC correlations - not so perfect*

Description

another fictional correlation matrix representing a *not so perfect* RIASEC circumplex

Usage

```
data(example2)
```

Format

A 6 x 6 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Examples

```
data(example2)
dim(example2)
example2
```

example3 *RIASEC and Big-Five correlations - perfect*

Description

another fictional correlation matrix representing RIASEC (circumplex) correlations and correlations (with) five dimensions of personality (big-five).

Usage

```
data(example3)
```

Format

A 11 x 11 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Examples

```
data(example3)
dim(example3)
example3
```

example4 *RIASEC and Big-Five correlations - not so perfect*

Description

another fictional correlation matrix representing *not so perfect* RIASEC (circumplex) correlations and correlations (with) five dimensions of personality (big-five).

Usage

```
data(example3)
```

Format

A 11 x 11 matrix object with named columns and rows.

Details

can be written as a .dat text file using the function `write_dat()`.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Examples

```
data(example4)
dim(example4)
example4
```

`holland_PACKAGE` *Statistics for the Framework of Holland's Theory of Vocational Choice*

Description

Offers a convenient way to compute parameters in the framework of vocational choice by J.L. Holland, (1997).

Details

The core of J.L. Holland's model of vocational interest orientations consists in the assumption of a vocational personality, which can be described with six basic dimensions. Based on this basic assumption, different theorems and (derived) constructs are part of the theory of vocational interest orientations (see Holland, 1997).

In its current version, the package 'holland' provides three main functional areas that allow for some statistical analysis from the theory of vocational interest orientation of J. L. Holland (see Holland, 1997).

One functional area is related to the concept of *congruence* between a person's interest orientation and a particular vocational environment. For this, the package (currently) offers ten R-functions with which different indices for the congruence can be calculated (see all functions starting with 'con_', e.g. `con_oneletter_holland`).

The second function area is related to the concept of *differentiation*, which is currently only covered with the function `dif_7_holland` to compute seven different indices of differentiation.

The last function area addresses the so-called *calculus* hypothesis, according to which the six interest orientations are arranged in the form of a hexagonal structure. The package 'holland' offers, among other functions, three (wrapper) functions, which are directly addressed to the user. Within the calculus hypothesis the arrangement of empirical data can be determined (cf. function `Circ_emp`) and their fit to the hexagonal structure can be determined (cf. function `Circ_test`). Furthermore, other construct domains (e.g., big-five personality) with their dimensions can be projected into the hexagonal structure (cf. Function `Circ_pro`). These three functions are based on the method of structural equation modeling proposed by Nagy et al. (2009), which was implemented as Mplus syntax. The application of the three functions therefore requires an installation of the commercial software Mplus (cf. also `MplusAutomation`).

Author(s)

- Joerg-Henrik Heine <jhheine@googlemail.com>

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources. #' @references Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

See Also

utils, MplusAutomation

Examples

```
#######################################
# see description of functions ...
```

---

kormean *Take the Mean of two Correlation Matrices*

Description

This function takes the mean of two correlation matrices using the Fisher-Z transformation of the coefficients in both matrices.

Usage

```
kormean(x, y, xn = NA, yn = NA)
```

Arguments

x	a correlation matrix
y	a correlation matrix
xn	numeric value (optionally) the number of observations for correlation matrix given in x
yn	numeric value (optionally) the number of observations for correlation matrix given in y

Details

this function uses the numerical values given in parameters xn and yn to compute the weighted mean of the Fisher-Z transformed coefficients in both correlation matrices. If either parameter xn or yn is not assigned a numerical value, the unweighted mean of both matrices is computed.

Value

the mean correlations of both matrices as a matrix object

Examples

```
## Correlation matrix for overall ASIT norm sample
data(AIST_2005_F_1270) # female sub-sample
data(AIST_2005_M_1226) # male sub-sample
kormean(x=AIST_2005_F_1270,y=AIST_2005_M_1226,xn=1270,yn=1226)
```

Mplus_eeal *Extracting empirical RIASEC angular locations from Mplus output*

Description

This function extracts the empirical RIASEC angular locations from an Mplus output file, which is the result of processing the Mplus syntax generated with the function `Mplus_esyn()`.

Usage

```
Mplus_eeal(target, konstrukt = c("R", "I", "A", "S", "E", "C"), ...)
```

Arguments

`target`	name of the Mplus output file name within the working directory as character. May also be a full path or relative path. Example: target = "MyMplus.out"
`konstrukt`	optionally a character vector with length = 6 containing labels for construct dimensions - default is konstrukt = c("R","I","A","S","E","C").
`...`	additional parameters passed through

Details

This function uses the function `readModels()` in package `MplusAutomation`. more to come

Value

returns a list object containing the RIASEC angular locations extracted from the Mplus output.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
## generating an example Mplus syntax referring to a R object with correlation data
data(example1) # loading fictional example correlation matrix
Mplus_esyn(N = 300, Cor = example1)
##### not run until Mplus inputfile 'example1.inp' is processed by Mplus ####
## Not run: Mplus_eeal("example1.out")
#################################################################
### clean up work directory
file.remove("example1.inp") # remove generated Mplus syntax from work dir.
file.remove("example1.dat") # remove generated cor. data from work dir.
```

`Mplus_epal` *Extracting RIASEC and additional construct angular locations from Mplus output*

Description

This function extracts the empirical RIASEC angular locations and angular locations for the additional (projected) construct from an Mplus output file, which is the result of processing the Mplus syntax generated with the function `Mplus_psyn()`.

Usage

```
Mplus_epal(target, M, konstrukt = "", ...)
```

Arguments

`target`	name of the Mplus output file name within the working directory as character. May also be a full path or relative path. Example: target = "MyMplus.out"
`M`	number of additional construct dimensions projected into the RIASEC circumplex - e.g. `M=5` for big-five personality dimensions
`konstrukt`	optionally a character vector with length = M containing labels for construct dimensions.
`...`	additional parameters passed through

Details

This function uses the function `readModels()` in package `MplusAutomation`.

more to come

Value

returns a list object containing the RIASEC and additional construct angular locations extracted from the Mplus output.

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
## generating an example Mplus syntax refering to a R object with correlation data
data(example1) # loading fictional example correlation matrix
Mplus_esyn(N = 300, Cor = example1)
##### not run until Mplus inputfile example1.inp is processed by Mplus #####
## Not run: Mplus_epal("example1.out",M=0)
#################################################################
### clean up work directory
file.remove("example1.inp") # remove generated Mplus syntax from work dir.
file.remove("example1.dat") # remove generated cor. data from work dir.
```

`Mplus_esyn` *Generating Mplus syntax for empirical RIASEC angular locations*

Description

This function generates executable Mplus syntax which is discussed in Nagy et. al. (2009). the resulting Mplus syntax estimates the angular locations of the six RIASEC dimensions, based on their correlations, in the assumed circumplex array within the theory of vocational choice by John Holland (1997).

Usage

```
Mplus_esyn(
  N,
  Cor,
  name = NULL,
  dummyvorl = 1,
  verbose = FALSE,
  eol = "\r\n",
  ...
)
```

Arguments

`N`	number of observations for correlations as numeric
`Cor`	an R matrix object with RIASEC correlations.

`name`	optional a name (as character) for the Mplus syntax to be saved - default is the object name or the name given in paramter `Cor` as character (with ending changed to '.inp').
`dummyvorl`	default is dummyvorl = 1, which results in the Mplus syntax given in Nagy et. al. (2009). !! don't change this !! if changed: *"Alternatively, the name of an external dummy template to be used can also be specified here, which is then read in and used. It is strongly recommended to use the internal syntax template, since the use of an external dummy template cannot be documented further here."*.
`verbose`	logical with default set to `verbose=FALSE`; if set to `verbose=TRUE` the resulting Mplus syntax is *additionally* printed on the console.
`eol`	character see `write.table` to control the generation of correct line endings in text files on different operating systems.
`...`	additional parameters passed through

Details

by default the Mplus syntax file is written in the current working directory.

Value

resulting Mplus syntax will be saved in the current working directory

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
# generating an example Mplus syntax refering to a R object
# with correlation data
data(example1) # loading fictional example correlation matrix
Mplus_esyn(N = 300, Cor = example1)
## Mplus syntax is now saved in the current workspace
################################################################
### clean up work directory
file.remove("example1.inp") # remove generated Mplus syntax from work dir.
file.remove("example1.dat") # remove generated cor. data from work dir.
```

`Mplus_fit` *Reading some fit indices from Mplus output*

Description

This function reads the fit-indices "ChiSq","df","p","CFI","RMSEA","SRMR" and returns them as a list and writes them (optionally) as a csv file into the current workspace directory.

Usage

```
Mplus_fit(target, w = FALSE, ...)
```

Arguments

`target`	name of the Mplus output file - e.g. target = "MyMplus.out" for an Mplus output file in the current workspace directory.
`w`	logical - write fit coefficients as csv table? - default: `w = FALSE`.
`...`	additional parameters passed through

Details

no details

Value

a list containing coefficients for model fit and a '.csv' file in the current workspace directory.

Examples

```
## first prepare an Mplus syntax
data(example2) # loading fictional example correlation matrix
Mplus_esyn(N = 300, Cor = example2)
## !!! now first open the data example2.inp with Mplus and click run !!!
##### not run until Mplus is installed on your system #####
## Not run: Mplus_fit("example2.out")
#################################################################
### clean up work directory
file.remove("example2.inp") # remove generated Mplus syntax from work dir.
file.remove("example2.dat") # remove generated cor. data from work dir.
```

`Mplus_psyn` *Generating Mplus syntax for empirical RIASEC angular locations with projected constructs*

Description

This function generates an extended executable Mplus syntax which is discussed in Nagy et. al. (2009). the resulting Mplus syntax estimates the angular locations of the six RIASEC dimensions, based on their correlations, in the asumed circumplex array within the theory of vocational choice by John Holland (1997). in contrast to the function `Mplus_esyn()` and the syntax discussed by Nagy et. al. (2009), additional constructs (correlated to RIASEC dimensions) are 'projeted' into the circular array of the six vocational interest dimensions.

Usage

```
Mplus_psyn(
  N,
  Cor,
  M,
  mpluserg,
  name = NULL,
  dummyvorl = 1,
  verbose = FALSE,
  eol = "\r\n",
  ...
)
```

Arguments

`N`	number of observations for correlations as numeric
`Cor`	either an R matrix object with RIASEC and additional construct correlations or the name of a correlation matrix, stored as a .dat text file, as character (see details).
`M`	number of additional construct dimensions to project into the RIASEC circumplex - e.g. `M=5` for big-five personality dimensions
`mpluserg`	name of the Mplus output data (as character - e.g. "myoutput.out") from which the fixation RIASEC-parameters should be read - this is usaly the result of Mplus prosessing an input syntax generated with the function `Mplus_esyn()`.
`name`	optional a name (as character) for the Mplus syntax to be saved - default is the object name or the name given in paramter `Cor` as character (with ending changed to '.inp').
`dummyvorl`	default is dummyvorl = 1, which results in the Mplus syntax given in Nagy et. al. (2009). !! don't change this !! if changed: *"Alternatively, the name of an external dummy template to be used can also be specified here, which is then read in and used. It is strongly recommended to use the internal syntax*

	template, since the use of an external dummy template cannot be documented further here.".
verbose	logical with default set to verbose=FALSE; if set to verbose=TRUE the resulting Mplus syntax is *additionally* printed on the console.
eol	character see write.table to control the generation of correct line endings in text files on different operating systems.
...	additional parameters passed through

Details

for execution of this function it is necessary that you have Mplus (version 6.11 or version 7) installed on your system. The function Mplus_psyn will read a Mplus output file, located in the current R workspace directory, which is a result of manualy runing an Mplus input file, generatetd by the function Mplus_esyn(). The name of the Mplus output data must be specified in the argument mpluserg. The structure of the correlation 'data' must follow the structure given in the data example3 in this package - thus the correlation data must start with the additional construct dimensions which should be projected into the RIASEC circumplex.

Value

resulting Mplus syntax wil be saved in the current working directory

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
## first preparing an Mplus output data
data(example1) # loading fictional example correlation matrix for fixing
Mplus_esyn(N = 300, Cor = example1)
## !!! now first open the 'example1.inp' with Mplus and click run !!!
##### not run until Mplus is installed on your system #####
# generating an example Mplus syntax referring to a correlation data
# stored in the already existing file 'example3'
# for projection of 5 personality dimensions into the circumplex
data(example3) # loading fictional example correlation matrix with add. constr.
## Not run:  Mplus_psyn(N = 300, Cor = example3,M = 5, mpluserg = "example1.out")
#################################################################
### clean up work directory
file.remove("example1.inp") # remove generated Mplus syntax from work dir.
file.remove("example1.dat") # remove generated cor. data from work dir.
```

`Mplus_tsyn` *Generating Mplus syntax for testing empirical RIASEC angular locations*

Description

This function generates executable Mplus syntax which is discussed in Nagy et. al. (2009). the resulting Mplus syntax estimates the fit of the empirical angular locations of the six RIASEC dimensions, found based on their correlations, in the assumed circumplex array within the theory of vocational choice by John Holland (1997).

Usage

```
Mplus_tsyn(
  N,
  Cor,
  name = NULL,
  test = "perfect",
  dummyvorl = 1,
  verbose = FALSE,
  eol = "\r\n",
  ...
)
```

Arguments

`N`	number of observations for correlations as numeric
`Cor`	either an R matrix object with RIASEC correlations or the name of a correlation matrix, stored as a .dat text file, as character (see details).
`name`	optional a name (as character) for the Mplus syntax to be saved - default is the object name or the name given in parameter `Cor` as character (with ending changed to '.inp').
`test`	either character (default test="perfect"), which tests against a perfect circumplex array, or a numeric vector with length = 6 giving the six angular locations (in radians) to test against.
`dummyvorl`	default is dummyvorl = 1, which results in the Mplus syntax given in Nagy et. al. (2009). !! don't change this !! if changed: *"Alternativ dazu kann hier auch der Name einer zu verwendenden Dummyvorlage angegeben werden die dann eingelesen und verwendet wird. Es empfiehlt sich dringend die interne Syntaxvorlage zu verwenden - es sei den zum weiterentwickeln und Testen der Funktionen"*.
`verbose`	logical with default set to `verbose=FALSE`; if set to `verbose=TRUE` the resulting Mplus syntax is *additionally* printed on the console.
`eol`	character see `write.table` to control the generation of correct line endings in text files on different operating systems.
`...`	additional parameters passed through

Details

more to come ...

Value

resulting Mplus syntax will be saved in the current working directory

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Nagy, G., Marsh, H. W., Luedtke, O., & Trautwein, U. (2009). Representing circles in our minds: Confirmatory factor analysis of circumplex structures and profiles. In T. Teo & M. S. Khine (Hrsg.), *Structural Equation Modeling in Educational Research: Concepts and applications* (S. 287 - 315). Rotterdam Boston Taipei: Sense Publishers.

Examples

```
## generating an example Mplus syntax referring to a R object with correlation data
data(example2) # loading fictional example correlation matrix
# generate and write Mplus input file in your workspace directory
Mplus_tsyn(N = 300, Cor = example2)
## Mplus syntax is now saved in the current workspace
################################################################
### clean up work directory
file.remove("example2.inp") # remove generated Mplus syntax from work dir.
file.remove("example2.dat") # remove generated cor. data from work dir.
```

---

plot.empCirc *S3 plotting method for object of class* "empCirc"

Description

plotting method for object of class 'empCirc' which results from function Circ_emp and contains the empirical RIASEC angular locations extracted from the Mplus result file after running the Mplus syntax.

Usage

```
## S3 method for class 'empCirc'
plot(
  x,
  main = NULL,
  lcolor = c("blue", "blue", "red", "red"),
  ltype = c(1, 2),
  lwd = 3,
 defhexa = list(hexa = TRUE, seg = TRUE, gr = 5, r = 4, nseg = 7, x.cent = 0, y.cent =
```

```
    0),
  ...
)
```

Arguments

x	object of class 'empCirc'.
main	titel for plot.
lcolor	(character) vector for colors; default set to lcolor=c("blue","blue","red","red").
ltype	vector for line types; default set to ltype=c(1,2)
lwd	numeric defining the line width; default set to lwd=3
defhexa	list of definition to draw the hexagon
...	other parameters passed through

Details

no details

Value

a plot visualizing the empirical RIASEC angular locations within the Hexagon.

---

plot.proCirc *S3 plotting method for object of class* "proCirc"

Description

plotting method for object of class 'proCirc'.

Usage

```
## S3 method for class 'proCirc'
plot(
  x,
  main = NULL,
 defhexa = list(r = 4, lwd = 3, col = "grey", lty = 1, cex = 1, col.lab = "grey", gr =
   5, hexa = TRUE, seg = TRUE, nseg = 7, col.seg = "grey", x.cent = 0, y.cent = 0),
 defemp = list(r = 3, lwd = 3, col = "black", lty = 2, cex = 0.8, col.lab = "black"),
  defcon = list(r = 4.5, lwd = 3, col = "darkgray", lty = 3, cex = 1, col.lab =
    "darkgray"),
  ...
)
```

Arguments

x	object of class 'empCirc'.
main	titel for the plot.
defhexa	a list with plotting options for hexagon.
defemp	a list with plotting options for empirical RIASEC angels.
defcon	a list with plotting options for empirical construct angels
...	other parameters passed through

Details

more to come ...

Value

a plot visualizing the empirical RIASEC and (projected) additional construct angular locations within the Hexagon.

---

plot_profile_holland *Plot an interest profile in the Hexagon*

Description

This function plots an interest profile in the Hexagon

Usage

```
plot_profile_holland(
  x,
  gr = 5,
  x.cent = 0,
  y.cent = 0,
  center = FALSE,
  r = NULL,
  s = 0,
  radial = TRUE,
  col.Hr = "grey",
  lwd.Hr = 1,
  lty.Hr = 1,
  circular = TRUE,
  col.Hc = "grey",
  lwd.Hc = 1,
  lty.Hc = 1,
  circle = FALSE,
  col.C = "grey",
  lwd.C = 1,
```

```
lty.C = 1,
measure = TRUE,
ri.M = NULL,
ro.M = NULL,
m = NULL,
pos.M = "c",
wid.M = 10,
col.M = "grey",
lwd.M = 1,
lty.M = 1,
vector = TRUE,
length.V = 0.25,
angle.V = 30,
code.V = 2,
col.V = "black",
lty.V = 1,
lwd.V = 1,
gridl = FALSE,
col.G = "grey",
lwd.G = 1,
lty.G = 1,
scalab = TRUE,
adj.sl = NULL,
pos.sl = NULL,
offset.sl = 0.5,
vfont.sl = NULL,
cex.sl = 1,
col.sl = NULL,
font.sl = NULL,
polyg = TRUE,
density.P = NULL,
angle.P = 45,
border.P = "black",
col.P = NA,
lwd.P = 1,
lty.P = par("lty"),
fillOddEven.P = FALSE,
lab = NULL,
s.la = 7,
r.la = NULL,
adj.la = NULL,
pos.la = NULL,
offset.la = 0.5,
vfont.la = NULL,
cex.la = 1,
col.la = NULL,
font.la = NULL,
...
```

```
)
```

Arguments

x	numeric vector with length of six values; either raw scores or normed values
gr	numeric optional specification of the (quadratic) plotting area.
x.cent	numeric optional specification of the x center of the plot
y.cent	numeric optional specification of the y center of the plot
center	logical whether to re-scale the minimum profile value to be in the center of the Hexagon
r	numeric optional specification of the radius (possibly vector with max length 6 – will be recycled) of the hexagonal plot (must be then in the same measure scale as x).
s	integer optional specification of the angular starting position (in degrees) when drawing the Hexagon (from 1 to 360).
radial	logical whether to plot radial (sector) lines within the Hexagon.
col.Hr	character expression of colors (possibly vector with max length 6 – will be recycled) for the radial Hexagon lines.
lwd.Hr	numeric (possibly vector with max length 6 – will be recycled) with line widths for the radial Hexagon lines.
lty.Hr	numeric or character expression (possibly vector with max length 6 – will be recycled) with line types for the radial Hexagon lines.
circular	logical whether to plot a circular (border) line around the Hexagon.
col.Hc	character expression of colors (possibly vector with max length 6 – will be recycled) for colors for the circular (border) Hexagon line.
lwd.Hc	numeric (possibly vector with max length 6 – will be recycled) with line widths for the circular (border) Hexagon line.
lty.Hc	numeric or character expression (possibly vector with max length 6 – will be recycled) with line types for the circular (border) Hexagon line.
circle	logical whether to plot a circle around the Hexagon.
col.C	character expression for color of circle around Hexagon.
lwd.C	numeric for line width of circle around Hexagon.
lty.C	numeric or character expression for line type of circle around Hexagon.
measure	logical whether to plot a (radial) measure scale (for each Hexagon dimension).
ri.M	numeric (possibly vector with max length 6 – will be recycled) with numeric value(s) for start (minimum) of the measure scale for each Hexagon dimension respectively (must be then in the same measure scale as x).
ro.M	numeric (possibly vector with max length 6 – will be recycled) with numeric value(s) for end (maximum) of the measure scale for each Hexagon dimension respectively (must be then in the same measure scale as x).
m	numeric with single value for the distance of the tic marks of the measure scale.

pos.M	character one of c("c","l","r") for the orientation of the tic marks with reference to the radial hexagon lines – centered, left, right respectively.
wid.M	numeric (possibly vector with max length 6 – will be recycled) defining the widths of the tic marks in degree.
col.M	character expression of colors (possibly vector with max length 6 – will be recycled) for the color of the tic marks.
lwd.M	numeric (possibly vector with max length 6 – will be recycled) for the line widths of the tic marks.
lty.M	numeric or character expression (possibly vector with max length 6 – will be recycled) for the line types of the tic marks.
vector	logical whether to plot the (resulting) total vector (as arrow) for the interest profile according to Eder, (1998).
length.V	length of the edges of the total vector arrow head (see arrows).
angle.V	angle from the shaft of the total vector arrow to the edge of the arrow head (see arrows).
code.V	integer code, determining kind of total vector arrow to be drawn(see arrows).
col.V	color of the total vector arrow (see arrows).
lty.V	line type of the total vector arrow (see arrows).
lwd.V	line wide of the total vector arrow (see arrows).
gridl	logical whether to plot a circular grid lines at the positions of the tic marks for the measure scale around the Hexagon.
col.G	character expression of colors (possibly vector with max length 6 – will be recycled) for the color of the grid lines.
lwd.G	numeric (possibly vector with max length 6 – will be recycled) for the line widths of the grid lines.
lty.G	numeric or character expression (possibly vector with max length 6 – will be recycled) for the line types of the grid lines.
scalab	logical whether to ad scale labeling with units.
adj.sl	parameter for scale labeling control – see text.
pos.sl	parameter for scale labeling control – see text.
offset.sl	parameter for scale labeling control – see text.
vfont.sl	parameter for scale labeling control – see text.
cex.sl	parameter for scale labeling control – see text.
col.sl	parameter for scale labeling control – see text.
font.sl	parameter for scale labeling control – see text.
polyg	whether to plot a polygon for the interest profile (see argument x).
density.P	parameter for polygon drawing control – see polygon.
angle.P	parameter for polygon drawing control – see polygon.
border.P	parameter for polygon drawing control – see polygon.
col.P	parameter for polygon drawing control – see polygon.

lwd.P	parameter for polygon drawing control – see polygon.
lty.P	parameter for polygon drawing control – see polygon.
fillOddEven.P	parameter for polygon drawing control – see polygon.
lab	character vector (with max length 6 – will be recycled) for labeling the Hexagon dimensions, by default names from x a taken ...
s.la	integer to control angular positions of labels (in degrees).
r.la	numeric optional specification of the radial positions (possibly vector with max length 6 – will be recycled) for the labels of the Hexagon dimensions (must be then in the same measure scale as x).
adj.la	parameter for labeling control – see text.
pos.la	parameter for labeling control – see text.
offset.la	parameter for labeling control – see text.
vfont.la	parameter for labeling control – see text.
cex.la	parameter for labeling control – see text.
col.la	parameter for labeling control – see text.
font.la	parameter for labeling control – see text.
...	parameters passed through

Details

no details – but see arguments.

Value

a plot and if argument vector=TRUE (default) a numeric value indicating the orientation of the total vector for the interest profile in degrees.

References

Eder, F. (1998). Differenziertheit der Interessen als Prädiktor der Interessenentwicklung. In J. Abel & C. Tarnai (Hrsg.), *Pädagogisch-psychologische Interessenforschung in Studium und Beruf* (S. 63–77). Münster: Waxmann.

Examples

```
#### a fictional interest profile:
A <- c(95, 125, 122, 105, 100, 90)
names(A) <- c("R","I","A","S","E","C")
#### plot with default settings
plot_profile_holland(x=A)
#### easy way to change global scaling with argument 'gr'
### additional change the color and thickness of the polygon line ...
plot_profile_holland(x=A,gr=3,border.P="darkblue",lwd.P = 2)
#### give the Hexagon a fixed scale range ...
plot_profile_holland(x=A,gr=10,ri.M=70,ro.M=130,lwd.P=2)
#### center the minimum and addition fix the scale range and step width ...
```

```
plot_profile_holland(x=A,gr=10,center=TRUE,ri.M=70,ro.M=130,m=10,lwd.P=2)

############## More examples
## Not run:
#### center the minimum without a fixed scale range but with a fixed step width ...
plot_profile_holland(x=A,gr=10,center=TRUE,m=5,col.P=NA,border.P="darkblue")
#### change position of the scale labels and polygon filling
plot_profile_holland(x=A,gr=3,pos.M="r",col.P="lightblue",density.P=10,
border.P="darkblue")
#### rotated (clockwise) by +90 degrees
plot_profile_holland(x=A,gr=3,pos.M="r",s=90,col.P="lightblue",density.P=10,
border.P="darkblue")
# add grid lines
plot_profile_holland(x=A,gr=3,pos.M="r",gridl=(TRUE),col.P="lightblue",
density.P=10,border.P="darkblue",lwd.P=2)
# plot 'blank' Hexagon without any interests profile ... tic marks suppressed
# but with Hexagon size adapted to the scaling given in 'A' (argument x)
plot_profile_holland(x=A,gr=3,vector=FALSE,gridl=TRUE,col.G="lightblue",
measure=F,polyg=FALSE)
# but with Hexagon size adapted to fixed range step width
plot_profile_holland(x=A,gr=3,vector=FALSE,gridl=TRUE,ri.M=70,ro.M=130,m=10,
measure=F,polyg=FALSE)
# ... centered
plot_profile_holland(x=A,gr=3,center=TRUE,vector=FALSE,gridl=TRUE,ri.M=70,
ro.M=130,m=20,measure=F,polyg=FALSE)
# ... with surrounding circle
plot_profile_holland(x=A,gr=3,circle=TRUE,center=TRUE,vector=FALSE,gridl=TRUE,
ri.M=70,ro.M=130,m=20,measure=F,polyg=FALSE)
# etc. pp. ... try different styles ...

## End(Not run)
```

`rad` *degrees to radians*

Description

This function converts angular locations from degrees to radians

Usage

```
rad(x, m = FALSE, rev = FALSE)
```

Arguments

`x`	numeric values in degrees
`m`	logical should values > 360 degrees be divided modulo
`rev`	logical if TRUE list objekt is returnd with number of revolutions

Details

no details

Value

values in radians; optionaly number of revolutions

Examples

```
## RIASEC angular locations in degrees to radians
rad(c(60,120,180,240,300,360))
rad(720)
rad(720,TRUE)
rad(360)
rad(360,TRUE)
# 810 degrees is two full revolutions and a quater
# which is 1.570796 radians or 90 degrees - check it!
rad(810,TRUE,TRUE)
```

---

rot.deg *rotate degrees by amount in degrees*

Description

This function rotates angular locations in degrees by an ampunt given in degrees

Usage

```
rot.deg(x, amount = 0, rev = FALSE)
```

Arguments

x	numeric values in degrees
amount	amount to rotate
rev	logical if TRUE list objekt is returnd with number of revolutions

Details

postive values vor amount will result in clockwise rotation an negative values will result in counter-clockwise rotation

Value

values in degrees rotatet by amount; optionaly number of revolutions

Examples

```
## rotation by 30 degrees
# inital value smaler than 330 degrees
rot.deg(200,30)
rot.deg(200,30,TRUE)
# inital value smaler than 360 degrees
rot.deg(350,30)
rot.deg(350,30,TRUE)
# inital value biger than 360 degrees
rot.deg(900,30)
rot.deg(900,30,TRUE)
```

`rot.rad` *rotate radians by amount in radians*

Description

This function rotates angular locations in radians by an ampunt given in radians

Usage

```
rot.rad(x, amount = 0, rev = FALSE)
```

Arguments

`x`	numeric values in radians
`amount`	amount to rotate (in radians)
`rev`	logical if TRUE list objekt is returnd with number of revolutions

Details

postive values vor `amount` will result in clockwise rotation an negative values will result in counter-clockwise rotation

Value

values in radians rotatet by amount; optionaly number of revolutions

Examples

```
## rotation by 0.5235988 radians
# inital value smaler than 5.759587 radians
rot.rad(3.490659,0.5235988)
rot.rad(3.490659,0.5235988,TRUE)
# inital value smaler than 6.283185 radians
rot.rad(6.108652,0.5235988)
rot.rad(6.108652,0.5235988,TRUE)
# inital value biger than 6.283185 radians
rot.rad(15.70796,0.5235988)
rot.rad(15.70796,0.5235988,TRUE)
```

sco2let *Convert Holland score profiles to Holland-letter-codes*

Description

The function converts a individual (person or environment) score profile consisting of six numerical score values into a Holland-letter code with length varying from 1 to 6 letters.

Usage

```
sco2let(A, len = 3)
```

Arguments

A	a numeric vector with Holland score values for the interest profile of length = 6.
len	a integer with values of either 1, 2, 3, 4, 5 or 6 indicating how many letters to return; default is set to len = 3 to return a Holland three-letter code.

Details

The numeric vector containing the score profile (see argument A) should have named numerical values (which is recommended for clarity). In this case, the order of the scores (e.g. names(A) <- c("R","I","A","S","E","C") or names(A) <- c("C","E","S","I","A","R") or any other) in the vector assigned to argument A does not matter. However, a vector with unnamed numerical values can also be used, in which case the assumption is made that the order of the Holland scores (numerical values) follows the scheme names(A) <- c("R","I","A","S","E","C"); see examples below.

Value

a character with the Holland-letter code (in upper case letters).

References

Holland, J.L. (1997). *Making vocational choices. A theory of vocational personalities and work environments*. Lutz, FL: Psychological Assessment Resources.

Examples

```
# A fictional interest profile:
A <- c(70, 90, 120, 50, 60, 130)
names(A) <- c("R","I","A","S","E","C")
A
sco2let(A)
# which is the same as ...
A <- c(70, 90, 120, 50, 60, 130); names(A)
A
sco2let(A)
```

```
# But see ...
A <- c(70, 90, 120, 50, 60, 130)
names(A) <-c("c","e","s","i","a","r")
A
sco2let(A)
# other length of letter code ...
sco2let(A, len = 1)
sco2let(A, len = 6)
sapply(1:6, function(x){sco2let(A,x)})
```

`sim_score_data` *Simulation of multivariate score data*

Description

This function will simulate Person (raw)-scores for an arbitrary number of dimensions (latent variables), assessed with any type of questionnaire given the maximum and minimum raw score for each dimension.

Usage

```
sim_score_data(
  n = 1000,
  cormat,
  min.score = 0,
  max.score = 40,
  data.frame = FALSE,
  ...
)
```

Arguments

`n`	integer giving the number of cases (observations) in the data to simulate.
`cormat`	a correlation matrix describing the associations between the dimensions – for Hollnd's theory, typical a 6 x 6 matrix with named columns and rows with `c("R","I","A","S","E","C")`.
`min.score`	numeric (possibly vector with max length == ncol(cormat) – will be recycled) with numeric value(s) defining the minimum raw scores per dimension
`max.score`	numeric (possibly vector with max length == ncol(cormat) – will be recycled) with numeric value(s) defining the maximum raw scores per dimension.
`data.frame`	logical whether to return a `data.frame` or a `matrix`
`...`	additional parameters passed through to `rmvnorm`.

Details

For Hollnd's theory, six dimensions (`c("R","I","A","S","E","C")`) are assumed being assessed with an questionnaire with 10 questions per dimension with each question having five response categories which are scored from '0' to '4' – thus min. raw score is 0 and max. rax score is 40 for each of the six dimension respectively.

Value

a data.frame with simulated raw scores.

Examples

```
# get an RIASEC correlation matrix
data(AIST_2005_F_1270)
# simulate raw scores with minimum = 0 and maximum = 40
a<-sim_score_data(n=1000,cormat=AIST_2005_F_1270)
apply(a, 2, range)
apply(a, 2, mean)
apply(a, 2, sd)
# simulate raw scores with minimum = 10 and maximum = 50
b<-sim_score_data(n=1000,cormat=AIST_2005_F_1270,min.score=10,max.score=50)
apply(b, 2, range)
apply(b, 2, mean)
apply(b, 2, sd)
# simulate norm scores (range between 70 and 130)
c<-sim_score_data(n=1000,cormat=AIST_2005_M_1226,min.score=70,max.score=130)
apply(c, 2, range)
apply(c, 2, mean)
apply(c, 2, sd)
```

---

write_dat *writing R matrix objects as .dat text files*

Description

this function writes R matrix objects as .dat text files to be read by Mplus.

Usage

```
write_dat(ob, file = paste(deparse(substitute(ob)), ".dat", sep = ""), ...)
```

Arguments

ob	the R-object to be written as .dat file.
file	optionally the name of the .dat file as character - default is 'objectname'.dat.
...	additional parameters passed through.

Details

no details.

Value

a .dat text file by default written in the current workspace directory.

Examples

```
## writing R-object example1 as example1.dat
## Not run:
data(example1)
write_dat(example1)

## End(Not run)
```

Index