Package ‘multid’

May 17, 2023

**Title**  Multivariate Difference Between Two Groups

**Version**  0.8.0

**Description**  Estimation of multivariate differences between two groups (e.g., multivariate sex differences) with regularized regression methods and predictive approach. See Lönnqvist & Ilmari-nen (2021) <doi:10.1007/s11109-021-09681-2> and Ilmari-nen et al. (2022) <doi:10.1177/08902070221088155>

Includes tools that help in understanding difference score reliability, predictions of difference score variables, conditional intra-class correlations, and heterogeneity of variance estimates. Package development was supported by the Academy of Finland research grant 338891.

**License**  GPL-3

**Encoding**  UTF-8

**BugReports**  https://github.com/vjilmari/multid/issues

**RoxygenNote**  7.2.3

**Imports**  dplyr (>= 1.0.7), glmnet (>= 4.1.2), stats (>= 4.0.2), pROC (>= 1.18.0), lavaan (>= 0.6.9), emmeans (>= 1.6.3), lme4 (>= 1.1.27.1), quantreg (>= 5.88)

**Suggests**  knitr (>= 1.39), rmarkdown (>= 2.14), overlapping (>= 1.7), rio (>= 0.5.29)

**VignetteBuilder**  knitr

**NeedsCompilation**  no

**Author**  Ville-Juhani Ilmarinen [aut, cre]

(https://orcid.org/0000-0001-9493-379X)

**Maintainer**  Ville-Juhani Ilmarinen <vj.ilmarinen@gmail.com>

**Repository**  CRAN

**Date/Publication**  2023-05-17 21:40:06 UTC

**R topics documented:**

- colwise_pool .................................................. 2
- cvv .......................................................... 3
- cvv_manual .................................................... 4
Column-wise pooling of standard deviations

**Description**

Column-wise pooling of standard deviations

**Usage**

```r
colwise_pool(data, n1, n2, m1, m2, sd1, sd2)
```

**Arguments**

- `data`  
  Data frame of `d_pooled_sd` output for multiple samples.
- `n1`  
  Sample sizes of group 1.
- `n2`  
  Sample sizes of group 2.
- `m1`  
  Means of group 1.
- `m2`  
  Means of group 2.
- `sd1`  
  Standard deviations of group 1.
- `sd2`  
  Standard deviations of group 2.

**Value**

pooled SDs for groups and across groups
Description

Calculates three different indices for variation between two or more variance estimates. VR = Variance ratio between the largest and the smallest variance. CVV = Coefficient of variance variation (Box, 1954). SVH = Standardized variance heterogeneity (Ruscio & Roche, 2012).

Usage

cvv(data)

Arguments

data Data frame of two or more columns or list of two or more variables.

Value

A vector including VR, CVV, and SVH.

References


Examples

d <- list(  
  X1 = rnorm(10, sd = 10),  
  X2 = rnorm(100, sd = 7.34),  
  X3 = rnorm(1000, sd = 6.02),  
  X4 = rnorm(100, sd = 5.17),  
  X5 = rnorm(10, sd = 4.56)  
)
  cvv(d)
Description

Calculates three different indices for variation between two or more variance estimates. VR = Variance ratio between the largest and the smallest variance. CVV = Coefficient of variance variation (Box, 1954). SVH = Standardized variance heterogeneity (Ruscio & Roche, 2012).

Usage

cvv_manual(sample_sizes, variances)

Arguments

- sample_sizes: Numeric vector of length > 1. Sample sizes used for each variance estimate.

Value

A vector including VR, CVV, and SVH.

References


Examples

cvv_manual(sample_sizes=c(10,100,1000,75,3),
          variances=c(1.5,2,2.5,3,3.5))
**ddsc_sem**

**Usage**

```r
ddsc_sem(
  data,
  x,
  y1,
  y2,
  center_yvars = FALSE,
  covariates = NULL,
  estimator = "ML",
  level = 0.95,
  sampling.weights = NULL,
  q_sesoi = 0,
  min_cross_over_point_location = 0
)
```

**Arguments**

- **data**: A data frame.
- **x**: Character string. Variable name of independent variable.
- **y1**: Character string. Variable name of first component score of difference score.
- **y2**: Character string. Variable name of second component score of difference score.
- **center_yvars**: Logical. Should y1 and y2 be centered around their grand mean? (Default FALSE)
- **covariates**: Character string or vector. Variable names of covariates (Default NULL).
- **estimator**: Character string. Estimator used in SEM (Default "ML").
- **level**: Numeric. The confidence level required for the result output (Default .95)
- **sampling.weights**: Character string. Name of sampling weights variable.
- **q_sesoi**: Numeric. The smallest effect size of interest for Cohen’s q estimates (Default 0; See Lakens et al. 2018).
- **min_cross_over_point_location**: Numeric. Z-score for the minimal slope cross-over point of interest (Default 0).

**Value**

- **descriptives**: Means, standard deviations, and intercorrelations.
- **parameter_estimates**: Parameter estimates from the structural equation model.
- **variance_test**: Variances and covariances of component scores.
- **data**: Data frame with original and scaled variables used in SEM.
- **results**: Summary of key results.
diff_two_dep_cors

References


Examples

```r
## Not run:
set.seed(342356)
D <- data.frame(
  y1 = rnorm(50),
  y2 = rnorm(50),
  x = rnorm(50)
)
ddsc_sem(
  data = D,
  y1 = "y1", y2 = "y2",
  x = "x",
  q_sesoi = 0.20,
  min_cross_over_point_location = 1
)$results

## End(Not run)
```

diff_two_dep_cors  

**Difference between two dependent Pearson’s correlations (with common index)**

Description

Calculates Cohen’s q effect size statistic for difference between two correlations, r_yx1 and r_yx2. Tests if Cohen’s q is different from zero while accounting for dependency between the two correlations.

Usage

```r
diff_two_dep_cors(data, y, x1, x2, level = 0.95, missing = "default")
```

Arguments

- `data`  
  Data frame.

- `y`  
  Character. Variable name of the common index variable.

- `x1`  
  Character. Variable name.

- `x2`  
  Character. Variable name.

- `level`  
  Numeric. The confidence level required for the result output (Default .95)

- `missing`  
  Character. Treatment of missing values (e.g., "ML", default = listwise deletion)
d_pooled_sd

Value

Parameter estimates from the fitted structural path model.

Examples

```r
set.seed(3864)
d<-data.frame(y=rnorm(100),x=rnorm(100))
d$x1<-d$x+rnorm(100)
d$x2<-d$x+rnorm(100)
diff_two_dep_cors(data=d,y="y",x1="x1",x2="x2")
```

---

**d_pooled_sd**  
*Standardized mean difference with pooled standard deviation*

Description

Standardized mean difference with pooled standard deviation

Usage

```r
d_pooled_sd(
  data,  
  var,  
  group.var,  
  group.values,  
  rename.output = TRUE,  
  infer = FALSE
)
```

Arguments

- **data**: A data frame.
- **var**: A continuous variable for which difference is estimated.
- **group.var**: The name of the group variable.
- **group.values**: Vector of length 2, group values (e.g. c("male", "female) or c(0,1)).
- **rename.output**: Logical. Should the output values be renamed according to the group.values? Default TRUE.
- **infer**: Logical. Statistical inference with Welch test? (default FALSE)

Value

Descriptive statistics and mean differences
Examples

d_pooled_sd(iris[iris$Species == "setosa" | iris$Species == "versicolor", ],
  var = "Petal.Length", group.var = "Species",
  group.values = c("setosa", "versicolor"), infer = TRUE
)

D_regularized  

Multivariate group difference estimation with regularized binomial regression

Description

Multivariate group difference estimation with regularized binomial regression

Usage

D_regularized(
  data,
  mv.vars,
  group.var,
  group.values,
  alpha = 0.5,
  nfolds = 10,
  s = "lambda.min",
  type.measure = "deviance",
  rename.output = TRUE,
  out = FALSE,
  size = NULL,
  fold = FALSE,
  fold.var = NULL,
  pcc = FALSE,
  auc = FALSE,
  pred.prob = FALSE,
  prob.cutoffs = seq(0, 1, 0.2),
  append.data = FALSE
)

Arguments

data  A data frame or list containing two data frames (regularization and estimation data, in that order).

mv.vars  Character vector. Variable names in the multivariate variable set.

group.var  The name of the group variable.

group.values  Vector of length 2, group values (e.g. c("male", "female") or c(0,1)).

alpha  Alpha-value for penalizing function ranging from 0 to 1: 0 = ridge regression, 1 = lasso, 0.5 = elastic net (default).
**D_regularized**

- `n_folds` (number of folds) is used for obtaining lambda (range from 3 to n-1, default 10).
- `s` (lambda value) is used for predicted values. Either "lambda.min" (default) or "lambda.1se".
- `type.measure` (measure used during cross-validation). Default "deviance".
- `rename.output` (logical). Should the output values be renamed according to the group.values? Default TRUE.
- `out` (logical). Should results and predictions be calculated on out-of-bag data set? Default FALSE.
- `size` (integer). Number of cases in regularization data per each group. Default 1/4 of cases.
- `fold` (logical). Is regularization applied across sample folds with separate predictions for each fold? Default FALSE.
- `fold.var` (character string). Name of the fold variable. Default NULL.
- `pcc` (logical). Include probabilities of correct classification? Default FALSE.
- `auc` (logical). Include area under the receiver operating characteristics? Default FALSE.
- `pred.prob` (logical). Include table of predicted probabilities? Default FALSE.
- `prob.cutoffs` (vector). Cutoffs for table of predicted probabilities. Default seq(0,1,0.20).
- `append.data` (logical). If TRUE, the data is appended to the predicted variables.

**Value**

- `D` Multivariate descriptive statistics and differences.
- `pred.dat` A data.frame with predicted values.
- `cv.mod` Regularized regression model from cv.glmnet.
- `P.table` Table of predicted probabilities by cutoffs.

**References**


**See Also**

- `cv.glmnet`
Examples

```
D_regularized(
    data = iris[iris$Species == "setosa" | iris$Species == "versicolor", ],
    group.var = "Species", group.values = c("setosa", "versicolor")
)$D

# out-of-bag predictions
D_regularized(
    data = iris[iris$Species == "setosa" | iris$Species == "versicolor", ],
    group.var = "Species", group.values = c("setosa", "versicolor"),
    out = TRUE, size = 15, pcc = TRUE, auc = TRUE
)$D

# separate sample folds
# generate data for 10 groups
set.seed(34246)
N = 100
n2 <- 10
d <-
data.frame(
    sex = sample(c("male", "female"), N * n2, replace = TRUE),
    fold = sample(x = LETTERS[1:n2], size = N * n2, replace = TRUE),
    x1 = rnorm(N * n2),
    x2 = rnorm(N * n2),
    x3 = rnorm(N * n2)
)

# Fit and predict with same data
D_regularized(
    data = d,
    mv.vars = c("x1", "x2", "x3"),
    group.var = "sex",
    group.values = c("female", "male"),
    fold.var = "fold",
    fold = TRUE,
    rename.output = TRUE
)$D

# Out-of-bag data for each fold
D_regularized(
    data = d,
    mv.vars = c("x1", "x2", "x3"),
    group.var = "sex",
    group.values = c("female", "male"),
    fold.var = "fold",
    size = 17,
    out = TRUE,
    fold = TRUE,
    rename.output = TRUE
)$D
```
D_regularized_fold

Use manually defined data folds for regularization and obtain estimates for each separately.

Description

Use manually defined data folds for regularization and obtain estimates for each separately.

Usage

D_regularized_fold(
  data,
  mv.vars,
  group.var,
  group.values,
  alpha = 0.5,
  s = "lambda.min",
  type.measure = "deviance",
  rename.output = TRUE,
  fold.var,
  append.data = FALSE
)

Arguments

data A data frame.

mv.vars Character vector. Variable names in the multivariate variable set.

group.var The name of the group variable.

group.values Vector of length 2, group values (e.g. c("male", "female") or c(0,1)).

alpha Alpha-value for penalizing function ranging from 0 to 1: 0 = ridge regression, 1 = lasso, 0.5 = elastic net (default).

s Which lambda value is used for predicted values? Either "lambda.min" (default) or "lambda.1se".

type.measure Which measure is used during cross-validation. Default "deviance".

rename.output Logical. Should the output values be renamed according to the group.values? Default TRUE.

fold.var Character string. Name of the fold variable.

append.data Logical. If TRUE, the original data is appended to the predicted variables.

Value

D Multivariate descriptive statistics and differences.

pred.dat A data.frame with predicted values.

cv.mod Regularized regression model from cv.glmnet.
D_regularized_fold_out

Use separate data partitions for regularization and estimation across defined data folds.

Description

Use separate data partitions for regularization and estimation across defined data folds.

Usage

D_regularized_fold_out(
  data,
  mv.vars,
  group.var,
  group.values,
  alpha = 0.5,
  s = "lambda.min",
  type.measure = "deviance",
  rename.output = TRUE,
  size = NULL,
  fold.var,
  pcc = FALSE,
D_regularized_fold_out

    auc = FALSE,
pred.prob = FALSE,
prob.cutoffs = seq(from = 0, to = 1, by = 0.2),
append.data = FALSE
)

Arguments

data A data frame.

mv.vars Character vector. Variable names in the multivariate variable set.
group.var The name of the group variable.
group.values Vector of length 2, group values (e.g. c("male", "female) or c(0,1)).
alpha Alpha-value for penalizing function ranging from 0 to 1: 0 = ridge regression, 1 = lasso, 0.5 = elastic net (default).
s Which lambda value is used for predicted values? Either "lambda.min" (default) or "lambda.1se".
type.measure Which measure is used during cross-validation. Default "deviance".
rename.output Logical. Should the output values be renamed according to the group.values? Default TRUE.
size Integer. Size of regularization data per each group. Default 1/4 of cases.
fold.var Name of the fold variable.
pcc Logical. Include probabilities of correct classification? Default FALSE.
auc Logical. Include area under the receiver operating characteristics? Default FALSE.
pred.prob Logical. Include table of predicted probabilities? Default FALSE.
prob.cutoffs Vector. Cutoffs for table of predicted probabilities. Default seq(0,1,0.20).
append.data Logical. If TRUE, the testing data split is appended to the predicted variables.

Value

D Multivariate descriptive statistics and differences.
pred.dat A data.frame with predicted values.
cv.mod Regularized regression model from cv.glmnet.
P.table Table of predicted probabilities by cutoffs.

Examples

set.seed(34246)
n1 <- 100
n2 <- 10
d <-
data.frame(
    sex = sample(c("male", "female"), n1 * n2, replace = TRUE),
    fold = sample(x = LETTERS[1:n2], size = n1 * n2, replace = TRUE),
    x1 = rnorm(n1 * n2),
)
D_regularized_out

x2 = rnorm(n1 * n2),
x3 = rnorm(n1 * n2)
)
D_regularized_out(
data = d,
mv.vars = c("x1", "x2", "x3"),
group.var = "sex",
group.values = c("female", "male"),
fold.var = "fold",
size = 17,
pcc = TRUE
)$$D

---

**D_regularized_out**

*Use separate data partition for regularization and estimation.*

### Description

Use separate data partition for regularization and estimation.

### Usage

```r
d_regularized_out(
data, 
mv.vars, 
group.var, 
group.values, 
alpha = 0.5, 
nfolds = 10, 
s = "lambda.min", 
type.measure = "deviance", 
rename.output = TRUE, 
size = NULL, 
pcc = FALSE, 
auc = FALSE, 
pred.prob = FALSE, 
prob.cutoffs = seq(from = 0, to = 1, by = 0.2), 
append.data = FALSE
)
```

### Arguments

- **data** A data frame or list containing two data frames (regularization and estimation data, in that order).
- **mv.vars** Character vector. Variable names in the multivariate variable set.
- **group.var** The name of the group variable.
- **group.values** Vector of length 2, group values (e.g. c("male", "female) or c(0,1)).
alpha  Alpha-value for penalizing function ranging from 0 to 1: 0 = ridge regression, 1 = lasso, 0.5 = elastic net (default).
nfolds  Number of folds used for obtaining lambda (range from 3 to n-1, default 10).
s  Which lambda value is used for predicted values? Either "lambda.min" (default) or "lambda.1se".
type.measure  Which measure is used during cross-validation. Default "deviance".
rename.output  Logical. Should the output values be renamed according to the group.values? Default TRUE.
size  Integer. Size of regularization data per each group. Default 1/4 of cases.
pcc  Logical. Include probabilities of correct classification? Default FALSE.
auc  Logical. Include area under the receiver operating characteristics? Default FALSE.
pred.prob  Logical. Include table of predicted probabilities? Default FALSE.
prob.cutoffs  Vector. Cutoffs for table of predicted probabilities. Default seq(0,1,0.20).
append.data  Logical. If TRUE, the testing data split is appended to the predicted variables.

Value

D  Multivariate descriptive statistics and differences.
pred.dat  A data.frame with predicted values.cv.mod  Regularized regression model from cv.glmnet.P.table  Table of predicted probabilities by cutoffs.

Examples

D_regularized_out(
  data = iris[iris$Species == "setosa" |
    iris$Species == "versicolor", ],
  mv.vars = c(
    "Petal.Length", "Petal.Width"
  ),
  group.var = "Species",
  group.values = c("setosa", "versicolor"),
  size = 40,
  pcc = TRUE
)$D
D_regularized_vanilla

Use same data partition for regularization and estimation.

Description

Use same data partition for regularization and estimation.

Usage

D_regularized_vanilla(
  data,
  mv.vars,
  group.var,
  group.values,
  alpha = 0.5,
  nfolds = 10,
  s = "lambda.min",
  type.measure = "deviance",
  rename.output = TRUE,
  append.data = FALSE
)

Arguments

data A data frame.

mv.vars Character vector. Variable names in the multivariate variable set.

group.var The name of the group variable.

group.values Vector of length 2, group values (e.g. c("male", "female) or c(0,1)).

alpha Alpha-value for penalizing function ranging from 0 to 1: 0 = ridge regression, 1 = lasso, 0.5 = elastic net (default).

nfolds Number of folds used for obtaining lambda (range from 3 to n-1, default 10).

s Which lambda value is used for predicted values? Either "lambda.min" (default) or "lambda.1se".

type.measure Which measure is used during cross-validation. Default "deviance".

rename.output Logical. Should the output values be renamed according to the group.values? Default TRUE.

append.data Logical. If TRUE, the original data is appended to the predicted variables.

Value

D Multivariate descriptive statistics and differences.

pred.dat A data.frame with predicted values.

cv.mod Regularized regression model from cv.glmnet.
ml_dadas

See Also
cv.glmnet

Examples

D_regularized(
  data = iris[iris$Species == "setosa" | iris$Species == "versicolor", ],
  group.var = "Species", group.values = c("setosa", "versicolor")
)$D

ml_dadas  Predicting algebraic difference scores in multilevel model

Description

Decomposes difference score predictions to predictions of difference score components by probing
simple effects at the levels of the binary moderator.

Usage

ml_dadas(
  model,
  predictor,
  diff_var,
  diff_var_values,
  scaled_estimates = FALSE,
  re_cov_test = FALSE,
  var_boot_test = FALSE,
  nsim = NULL,
  level = 0.95,
  seed = NULL,
  abs_diff_test = 0
)

Arguments

model  Multilevel model fitted with lmerTest.
predictor  Character string. Variable name of independent variable predicting difference score.
diff_var  Character string. A variable indicative of difference score components (two groups).
diff_var_values  Vector. Values of the component score groups in diff_var.
scaled_estimates  Logical. Are scaled estimates obtained? Does fit a reduced model for correct standard deviations. (Default FALSE)
re_cov_test Logical. Significance test for random effect covariation? Does fit a reduced model without the correlation. (Default FALSE)

var_boot_test Logical. Compare variance by lower-level groups at the upper-level in a reduced model with bootstrap? (Default FALSE)

nsim Numeric. Number of bootstrap simulations.

level Numeric. The confidence level required for the var_boot_test output (Default .95)

seed Numeric. Seed number for bootstrap simulations.

abs_diff_test Numeric. A value against which absolute difference between component score predictions is tested (Default 0).

Value

dadas A data frame including main effect, interaction, regression coefficients for component scores, dadas, and comparison between interaction and main effect.

scaled_estimates Scaled regression coefficients for difference score components and difference score.

vpc_at_reduced Variance partition coefficients in the model without the predictor and interactions.

re_cov_test Likelihood ratio significance test for random effect covariation.

boot_var_diffs List of different variance bootstrap tests.

Examples

```r
## Not run:
set.seed(95332)
n1 <- 10 # groups
n2 <- 10 # observations per group
dat <- data.frame(
  group = rep(c(LETTERS[1:n1]), each = n2),
  w = sample(c(-0.5, 0.5), n1 * n2, replace = TRUE),
  x = rep(sample(1:5, n1, replace = TRUE), each = n2),
  y = sample(1:5, n1 * n2, replace = TRUE)
)
library(lmerTest)
fit <- lmerTest::lmer(y ~ x * w + (w | group),
  data = dat)
round(ml_dadas(fit,
  predictor = "x",
  diff_var = "w",
  diff_var_values = c(0.5, -0.5)
)$dadas, 3)
## End(Not run)
```
pcc

Returns probabilities of correct classification for both groups in independent data partition.

Description

Returns probabilities of correct classification for both groups in independent data partition.

Usage

\[
\text{pcc}(\text{data, pred.var, group.var, group.values})
\]

Arguments

\begin{itemize}
\item \textbf{data} Data frame including predicted values (e.g., \text{pred.dat} from \text{D_regularized_out}).
\item \textbf{pred.var} Character string. Variable name for predicted values.
\item \textbf{group.var} The name of the group variable.
\item \textbf{group.values} Vector of length 2, group values (e.g. c("male", "female") or c(0,1)).
\end{itemize}

Value

Vector of length 2. Probabilities of correct classification.

Examples

\[
\text{D_out} \leftarrow \text{D_regularized_out}(
\begin{array}{l}
\text{data = iris[iris$Species == "versicolor" \mid iris$Species == "virginica", ],}
\text{group.var = "Species", group.values = c("versicolor", "virginica"),}
\text{size = 15}
\end{array}
\)
\]

\[
\text{pcc}(
\begin{array}{l}
\text{data = D_out$pred.dat,}
\text{pred.var = "pred",}
\text{group.var = "group",}
\text{group.values = c("versicolor", "virginica")}
\end{array}
\)
\]
**qcc**

Quantile correlation coefficient

**Description**

For computation of tail dependence as correlations estimated at different variable quantiles (Choi & Shin, 2022; Lee et al., 2022) summarized across two quantile regression models where x and y switch roles as independent/dependent variables.

**Usage**

```r
qcc(
  x,
  y,
  tau = c(0.1, 0.5, 0.9),
  data,
  method = "br",
  boot_n = NULL,
  ci_level = 0.95
)
```

**Arguments**

- `x`: Name of x variable. Character string.
- `y`: Name of y variable. Character string.
- `tau`: The quantile(s) to be estimated. A vector of values between 0 and 1, default `c(.1,.5,.9)`. @seealso `rq`
- `data`: Data frame.
- `method`: The algorithmic method used to compute the fit (default "br"). @seealso `rq`
- `boot_n`: Number of bootstrap redraws (default NULL = no bootstrap inference).
- `ci_level`: Level for percentile bootstrap confidence interval. Numeric values between 0 and 1. Default .95.

**Value**

- `r`: Pearson’s correlation estimate for comparison.
- `rho_tau`: Correlations at different tau values (quantiles).
- `r_boot_est`: Pearson’s correlation bootstrap estimates.
- `rho_tau_boot_est`: Bootstrap estimates for correlations at different tau values (quantiles).
References


Examples

```r
set.seed(2321)
d <- data.frame(x = rnorm(2000))
d$y <- 0.10 * d$x + (0.20) * d$x^2 + 0.40 * d$x^3 + (-0.20) * d$x^4 + rnorm(2000)
qcc_boot <- qcc(x = "x", y = "y", data = d, tau = 1:9 / 10, boot_n = 50)
qcc_boot$rho_tau
```

reliability_dms

Reliability calculation for difference score variable that is a difference between two mean variables calculated over upper-level units (e.g., sex differences across countries)

Description

Calculates reliability of difference score (Johns, 1981) based on two separate ICC2 values (Bliese, 2000), standard deviations of mean values over upper-level units, and correlations between the mean values across upper-level units.

Usage

```r
reliability_dms(
  model = NULL,
  data = NULL,
  diff_var,
  diff_var_values,
  var,
  group_var
)
```

Arguments

- **model**: Multilevel model fitted with lmer (default NULL)
- **data**: Long format data frame (default NULL)
- **diff_var**: Character string. A variable indicative of difference score components (two groups).
- **diff_var_values**: Vector. Values of the component score groups in diff_var.
- **var**: Character string. Name of the dependent variable or variable of which mean values are calculated.
- **group_var**: Character string. Upper-level clustering unit.
Value

A vector including ICC2s (r11 and r22), SDs (sd1, sd2, and sd_d12), means (m1, m2, and m_d12), correlation between means (r12), and reliability of the mean difference variable.

References


Examples

```r
set.seed(4317)
n2 <- 20
n1 <- 200
ri <- rnorm(n2, m = 0.5, sd = 0.2)
rs <- 0.5 * ri + rnorm(n2, m = 0.3, sd = 0.15)
d.list <- list()
for (i in 1:n2) {
x <- rep(c(-0.5, 0.5), each = n1 / 2)
y <- ri[i] + rs[i] * x + rnorm(n1)
d.list[[i]] <- cbind(x, y, i)
}
d <- data.frame(do.call(rbind, d.list))
names(d) <- c("x", "y", "cntry")
reliability_dms(
  data = d, diff_var = "x",
  diff_var_values = c(-0.5, 0.5), var = "y", group_var = "cntry"
)
```

Description

Predicting algebraic difference scores in structural equation model

Usage

```r
sem_dadas(
  data, 
  var1, 
  var2,
)```
sem_dadas

```r
center = FALSE,
scale = FALSE,
predictor,
covariates = NULL,
estimator = "MLR",
level = 0.95,
sampling.weights = NULL,
abs_coef_diff_test = 0
)
```

**Arguments**

- **data**
  - A data frame.

- **var1**
  - Character string. Variable name of first component score of difference score ($Y_1$).

- **var2**
  - Character string. Variable name of second component score of difference score ($Y_2$).

- **center**
  - Logical. Should var1 and var2 be centered around their grand mean? (Default FALSE)

- **scale**
  - Logical. Should var1 and var2 be scaled with their pooled sd? (Default FALSE)

- **predictor**
  - Character string. Variable name of independent variable predicting difference score.

- **covariates**
  - Character string or vector. Variable names of covariates (Default NULL).

- **estimator**
  - Character string. Estimator used in SEM (Default "MLR").

- **level**
  - Numeric. The confidence level required for the result output (Default .95)

- **sampling.weights**
  - Character string. Name of sampling weights variable.

- **abs_coef_diff_test**
  - Numeric. A value against which absolute difference between component score predictions is tested (Default 0).

**Value**

- **descriptives**
  - Means, standard deviations, and intercorrelations.

- **parameter_estimates**
  - Parameter estimates from the structural equation model.

- **variance_test**
  - Variances and covariances of component scores.

- **transformed_data**
  - Data frame with variables used in SEM.

- **dadas**
  - One sided dadas-test for positivity of abs($b_{11}$-$b_{21}$)-abs($b_{11}$+$b_{21}$).

- **results**
  - Summary of key results.

**References**

Examples

```r
## Not run:
set.seed(342356)
d <- data.frame(
  var1 = rnorm(50),
  var2 = rnorm(50),
  x = rnorm(50)
)
sem_dadas(
  data = d, var1 = "var1", var2 = "var2",
  predictor = "x", center = TRUE, scale = TRUE,
  abs_coef_diff_test = 0.20
)
$results

## End(Not run)
```

---

**value_correlation**

*Testing and quantifying how much ipsatization (profile centering) influence associations between value and a correlate*

**Description**

Testing and quantifying how much ipsatization (profile centering) influence associations between value and a correlate

**Usage**

```r
value_correlation(
  data, rv, cf, correlate,
  scale_by_rv = FALSE, standardize_correlate = FALSE,
  estimator = "ML", level = 0.95,
  sampling.weights = NULL, sesoi = 0
)
```

**Arguments**

- **data**
  - A data frame.
- **rv**
  - Character string or vector. Variable name(s) of the non-ipsatized value variable(s) (raw value score).
- **cf**
  - Character string. Variable name of the common factor that is used for ipsatizing raw value scores.
value_correlation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>correlate</td>
<td>Character string. Name of the variable to which associations with values are examined.</td>
</tr>
<tr>
<td>scale_by_rv</td>
<td>Logical. Is standard deviation of the raw non-ipsatized value score used for scaling the common factor as well? (Default FALSE)</td>
</tr>
<tr>
<td>standardize_correlate</td>
<td>Logical. Should the correlate be standardized? (Default FALSE)</td>
</tr>
<tr>
<td>estimator</td>
<td>Character string. Estimator used in SEM (Default &quot;ML&quot;).</td>
</tr>
<tr>
<td>level</td>
<td>Numeric. The confidence level required for the result output (Default .95)</td>
</tr>
<tr>
<td>sampling.weights</td>
<td>Character string. Name of sampling weights variable.</td>
</tr>
<tr>
<td>sesoi</td>
<td>Numeric. Smallest effect size of interest. Used for equivalence testing differences in ipsatized and non-ipsatized value associations (Default 0).</td>
</tr>
</tbody>
</table>

Value

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>parameter_estimates</td>
<td>Parameter estimates from the structural equation model.</td>
</tr>
<tr>
<td>transformed_data</td>
<td>Data frame with variables used in SEM (after scaling is applied).</td>
</tr>
<tr>
<td>results</td>
<td>Summary of key results.</td>
</tr>
</tbody>
</table>

Examples

```r
## Not run:
set.seed(342356)
d <- data.frame(
  rv1 = rnorm(50),
  rv2 = rnorm(50),
  rv3 = rnorm(50),
  rv4 = rnorm(50),
  x = rnorm(50)
)
d$cf<-rowMeans(d[,c("rv1","rv2","rv3","rv4")])
fit<-value_correlation(
  data = d, rv = c("rv1","rv2","rv3","rv4"), cf = "cf",
  correlate = "x",scale_by_rv = TRUE,
  standardize_correlate = TRUE,
  sesoi = 0.10
)
round(fit$variability_summary,3)
round(fit$association_summary,3)
## End(Not run)
```
Description

Calculates variance estimates (level-2 Intercept variance) and variance partition coefficients (i.e., intra-class correlation) at selected values of predictor values in two-level linear models with random effects (intercept, slope, and their covariation).

Usage

vpc_at(model, lvl1.var, lvl1.values)

Arguments

model Two-level model fitted with lme4. Must include random intercept, slope, and their covariation.

lvl1.var Character string. Level 1 variable name to which random slope is also estimated.

lvl1.values Level 1 variable values.

Value

Data frame of level 2 variance and std.dev. estimates at level 1 variable values, respective VPCs (ICC1s) and group-mean reliabilities (ICC2s) (Bliese, 2000).

References


Examples

```r
fit <- lme4::lmer(Sepal.Length ~ Petal.Length +
                 (Petal.Length | Species),
                 data = iris)

lvl1.values <-
c(
  mean(iris$Petal.Length) - stats::sd(iris$Petal.Length),
  mean(iris$Petal.Length),
  mean(iris$Petal.Length) + stats::sd(iris$Petal.Length)
)
```
vpc_at

vpc_at(
    model = fit,
    lvl1.var = "Petal.Length",
    lvl1.values = lvl1.values
)
Index

colwise_pool, 2
cv.glmnet, 9, 12, 17
cvv, 3
cvv_manual, 4
d_pooled_sd, 7
D_regularized, 8
D_regularized_fold, 11
D_regularized_fold_out, 12
D_regularized_out, 14
D_regularized_vanilla, 16
ddsc_sem, 4
diff_two_depCors, 6
ml_dadas, 17
pcc, 19
qcc, 20
reliability_dms, 21
rq, 20
sem_dadas, 22
value_correlation, 24
vpc_at, 26