

Package ‘CoTiMA’

October 12, 2021

Type Package

Title Continuous Time Meta-Analysis ('CoTiMA')

Version 0.4.4

Date 2021-10-12

Description The 'CoTiMA' package performs meta-analyses of correlation matrices of repeatedly measured variables taken from studies that used different time intervals. Different time intervals between measurement occasions impose problems for meta-analyses because the effects (e.g. cross-lagged effects) cannot be simply aggregated, for example, by means of common fixed or random effects analysis. However, continuous time math, which is applied in 'CoTiMA', can be used to extrapolate or intrapolate the results from all studies to any desired time lag. By this, effects obtained in studies that used different time intervals can be meta-analyzed. 'CoTiMA' fits models to empirical data using the structural equation model (SEM) package 'ctsem', the effects specified in a SEM are related to parameters that are not directly included in the model (i.e., continuous time parameters; together, they represent the continuous time structural equation model, CTSEM). Statistical model comparisons and significance tests are then performed on the continuous time parameter estimates. 'CoTiMA' also allows analysis of publication bias (Egger's test, PET-PEESE estimates, zcurve analysis etc.) and analysis of statistical power (post hoc power, required sample sizes). See Dormann, C., Guthier, C., & Cortina, J. M. (2019) <doi:10.1177/1094428119847277>. and Guthier, C., Dormann, C., & Voelkle, M. C. (2020) <doi:10.1037/bul0000304>.

License GPL-3

URL <https://github.com/CoTiMA/CoTiMA>

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0), OpenMx (>= 2.18.1), ctsem (>= 3.3.11), lavaan (>= 0.6), foreach (>= 1.5.1)

Imports MBESS ($\geq 4.6.0$), crayon ($\geq 1.3.4$), psych ($\geq 1.9.12$),
doParallel ($\geq 1.0.15$), rootSolve ($\geq 1.8.2$), abind ($\geq 1.4-5$),
RPushbullet ($\geq 0.3.3$), openxlsx ($\geq 4.2.2$), zcurve ($\geq 1.0.7$),
scholar ($\geq 0.2.0$), stringi ($\geq 1.0.7$), MASS

Suggests R.rsp

VignetteBuilder R.rsp

RoxygenNote 7.1.1

NeedsCompilation no

Author Christian Dormann [aut, cph],
Markus Homberg [aut, com, cre],
Christina Guthier [ctb],
Manuel Voelkle [ctb]

Maintainer Markus Homberg <cotima@uni-mainz.de>

Repository CRAN

Date/Publication 2021-10-12 08:20:08 UTC

R topics documented:

A128	5
A313	5
addedByResearcher2	6
addedByResearcher3	6
addedByResearcher313	7
ageM128	7
ageM18	8
ageM2	8
ageM3	9
ageM313	9
ageM32	10
ageSD128	10
ageSD18	11
ageSD2	11
ageSD3	12
ageSD313	12
ageSD32	13
alphas128	13
alphas313	14
burnout128	14
burnout18	15
burnout2	15
burnout3	16
burnout313	16
burnout32	17
combineVariables128	17
combineVariablesNames128	18

CoTiMABiG_D_BO	18
CoTiMAFullFit_3	19
CoTiMAFullFit_6	19
CoTiMAFullInv23Fit_6	20
CoTiMAFullInvEq23Fit_6	20
CoTiMAInitFit_3	21
CoTiMAInitFit_6	21
CoTiMAInitFit_6_NUTS	22
CoTiMAInitFit_D_BO	22
CoTiMAMod1onFullFit_6	23
CoTiMAMod2on23Fit_6	23
CoTiMAPart134Inv3Fit_6	24
CoTiMAPower_D_BO	24
CoTiMAStanctArgs	25
CoTiMAstudyList_3	25
CoTiMAstudyList_6	26
country128	26
country18	27
country2	27
country3	28
country313	28
country32	29
ctmaAllInvFit	29
ctmaBiG	31
ctmaBiGOMX	32
ctmaCombPRaw	33
ctmaCompFit	33
ctmaCorRel	34
ctmaEmpCov	35
ctmaEqual	36
ctmaFit	38
ctmaFitList	41
ctmaFitToPrep	42
ctmaGetPub	42
ctmaInit	43
ctmaLabels	46
ctmaOptimizeInit	47
ctmaPlot	48
ctmaPower	50
ctmaPRaw	52
ctmaPrep	53
ctmaPub	55
ctmaSaveFile	57
ctmaStanResample	58
ctmaSV	58
delta_t128	59
delta_t18	60
delta_t2	60

delta_t3	61
delta_t313	61
delta_t32	62
demands128	62
demands18	63
demands2	63
demands3	64
demands313	64
demands32	65
dl_link	65
empcov128	66
empcov18	66
empcov2	67
empcov3	67
empcov313	68
empcov32	68
malePercent128	69
malePercent18	69
malePercent2	70
malePercent3	70
malePercent313	71
malePercent32	71
moderator128	72
moderator18	72
moderator2	73
moderator3	73
moderator313	74
moderator32	74
moderatorLabels	75
moderatorValues	75
occupation128	76
occupation18	76
occupation2	77
occupation3	77
occupation313	78
occupation32	78
optimFit313	79
pairwiseN128	79
plot.CoTiMAFit	80
pubList_8	80
pubResults_6	81
rawData128	81
recodeVariables128	82
results128	82
sampleSize128	83
sampleSize18	83
sampleSize2	84
sampleSize3	84

sampleSize313	85
sampleSize32	85
source128	86
source2	86
source3	87
source313	87
summary.CoTiMAFit	88
targetVariables128	88
targetVariables2	89
targetVariables3	89
targetVariables313	90
variableNames128	90

Index **91**

A128	<i>A128 example matrix</i>
------	----------------------------

Description

A128 example matrix

Usage

A128

Format

An object of class `matrix` (inherits from `array`) with 2 rows and 2 columns.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

A313	<i>A313 example matrix</i>
------	----------------------------

Description

A313 example matrix

Usage

A313

Format

An object of class `matrix` (inherits from `array`) with 2 rows and 2 columns.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

addedByResearcher2 *addedByResearcher2 example vector*

Description

addedByResearcher2 example vector

Usage

addedByResearcher2

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

addedByResearcher3 *addedByResearcher3 example vector*

Description

addedByResearcher3 example vector

Usage

addedByResearcher3

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

addedByResearcher313 *addedByResearcher313 example vector*

Description

addedByResearcher313 example vector

Usage

addedByResearcher313

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageM128 *ageM128 example vector*

Description

ageM128 example vector

Usage

ageM128

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageM18

ageM18 example vector

Description

ageM18 example vector

Usage

ageM18

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageM2

ageM2 example vector

Description

ageM2 example vector

Usage

ageM2

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageM3

ageM3 example vector

Description

ageM3 example vector

Usage

ageM3

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageM313

ageM313 example vector

Description

ageM313 example vector

Usage

ageM313

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageM32

ageM32 example vector

Description

ageM32 example vector

Usage

ageM32

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageSD128

ageSD128 example vector

Description

ageSD128 example vector

Usage

ageSD128

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageSD18	<i>ageSD18 example vector</i>
---------	-------------------------------

Description

ageSD18 example vector

Usage

ageSD18

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageSD2	<i>ageSD2 example vector</i>
--------	------------------------------

Description

ageSD2 example vector

Usage

ageSD2

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageSD3

ageSD3 example vector

Description

ageSD3 example vector

Usage

ageSD3

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageSD313

ageSD313 example vector

Description

ageSD313 example vector

Usage

ageSD313

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ageSD32

ageSD32 example vector

Description

ageSD32 example vector

Usage

ageSD32

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

alphas128

alphas128 example vector

Description

alphas128 example vector

Usage

alphas128

Format

An object of class `numeric` of length 9.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

alphas313

alphas313 example vector

Description

alphas313 example vector

Usage

alphas313

Format

An object of class numeric of length 6.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

burnout128

burnout128 example vector

Description

burnout128 example vector

Usage

burnout128

Format

An object of class character of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

burnout18

burnout18 example vector

Description

burnout18 example vector

Usage

burnout18

Format

An object of class character of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

burnout2

burnout2 example vector

Description

burnout2 example vector

Usage

burnout2

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

burnout3

burnout3 example vector

Description

burnout3 example vector

Usage

burnout3

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

burnout313

burnout313 example vector

Description

burnout313 example vector

Usage

burnout313

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

burnout32

burnout32 example vector

Description

burnout32 example vector

Usage

burnout32

Format

An object of class character of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

combineVariables128

combineVariables128 example vector

Description

combineVariables128 example vector

Usage

combineVariables128

Format

An object of class list of length 3.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

combineVariablesNames128

combineVariablesNames128 example vector

Description

combineVariablesNames128 example vector

Usage

combineVariablesNames128

Format

An object of class character of length 3.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMABiG_D_BO

ctmaBiG-object reproducing results of Guthier et al. (2020)

Description

ctmaBiG-object reproducing results of Guthier et al. (2020)

Usage

CoTiMABiG_D_BO

Format

An object of class CoTiMAFit of length 11.

Author(s)

C. Guthier, C. Dormann & J. Cortina <CoTiMA@uni-mainz.org>

CoTiMAFullFit_3 *ctmaFit-object with a 'full' CoTiMA of 3 studies*

Description

ctmaFit-object with a 'full' CoTiMA of 3 studies

Usage

CoTiMAFullFit_3

Format

An object of class CoTiMAFit of length 19.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAFullFit_6 *ctmaFit-object with a 'full' CoTiMA of 6 studies*

Description

ctmaFit-object with a 'full' CoTiMA of 6 studies

Usage

CoTiMAFullFit_6

Format

An object of class CoTiMAFit of length 13.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAFullInv23Fit_6 *1st fitted ctmaFit-object in a series of 2 to test equality of 2 cross effects*

Description

1st fitted ctmaFit-object in a series of 2 to test equality of 2 cross effects

Usage

CoTiMAFullInv23Fit_6

Format

An object of class CoTiMAFit of length 13.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAFullInvEq23Fit_6
2nd fitted ctmaFit-object in a series of 2 to test equality of 2 cross effects

Description

2nd fitted ctmaFit-object in a series of 2 to test equality of 2 cross effects

Usage

CoTiMAFullInvEq23Fit_6

Format

An object of class CoTiMAFit of length 15.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAInitFit_3 *ctmaInit-object with of 3 primary studies*

Description

ctmaInit-object with of 3 primary studies

Usage

CoTiMAInitFit_3

Format

An object of class CoTiMAFit of length 16.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAInitFit_6 *ctmaInit-object with 6 primary studies*

Description

ctmaInit-object with 6 primary studies

Usage

CoTiMAInitFit_6

Format

An object of class CoTiMAFit of length 18.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAInitFit_6_NUTS *ctmaInit-object with a 'full' CoTiMA of 6 studies using NUTS sampler*

Description

ctmaInit-object with a 'full' CoTiMA of 6 studies using NUTS sampler

Usage

CoTiMAInitFit_6_NUTS

Format

An object of class CoTiMAFit of length 18.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAInitFit_D_BO *ctmaInit-object created by Guthier et al. (2020) with 48 primary studies*

Description

ctmaInit-object created by Guthier et al. (2020) with 48 primary studies

Usage

CoTiMAInitFit_D_BO

Format

An object of class CoTiMAFit of length 13.

Author(s)

C. Guthier, C. Dormann & J. Cortina <CoTiMA@uni-mainz.org>

CoTiMAMod1onFullFit_6 *ctmaFit-object with a categorical moderator of the full drift matrix*

Description

ctmaFit-object with a categorical moderator of the full drift matrix

Usage

CoTiMAMod1onFullFit_6

Format

An object of class CoTiMAFit of length 16.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAMod2on23Fit_6 *ctmaFit-object with a continuous moderator of 2 cross effects*

Description

ctmaFit-object with a continuous moderator of 2 cross effects

Usage

CoTiMAMod2on23Fit_6

Format

An object of class CoTiMAFit of length 19.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAPart134Inv3Fit_6

ctmaFit-object with with only one cross effect and this one set equal across primary studies

Description

ctmaFit-object with with only one cross effect and this one set equal across primary studies

Usage

CoTiMAPart134Inv3Fit_6

Format

An object of class CoTiMAFit of length 19.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAPower_D_BO

ctmaPower-object reproducing results of Guthier et al. (2020)

Description

ctmaPower-object reproducing results of Guthier et al. (2020)

Usage

CoTiMAPower_D_BO

Format

An object of class CoTiMAFit of length 11.

Author(s)

C. Guthier, C. Dormann & J. Cortina <CoTiMA@uni-mainz.org>

CoTiMAStanctArgs *This are preset arguments*

Description

This are preset arguments

Usage

CoTiMAStanctArgs

Format

An object of class list of length 33.

CoTiMAstudyList_3 *ctmaPrep-object created with 3 primary studies*

Description

ctmaPrep-object created with 3 primary studies

ctmaPrep-object created with 6 primary studies

Usage

CoTiMAstudyList_3

CoTiMAstudyList_3

Format

An object of class CoTiMAFit of length 27.

An object of class CoTiMAFit of length 27.

Author(s)

C. Guthier, C. Dormann & J. Cortina <CoTiMA@uni-mainz.org>

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAstudyList_6 *ctmaPrep-object created with 6 primary studies*

Description

ctmaPrep-object created with 6 primary studies

Usage

CoTiMAstudyList_6

Format

An object of class CoTiMAFit of length 29.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

country128 *country128 example vector*

Description

country128 example vector

Usage

country128

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

country18

country18 example vector

Description

country18 example vector

Usage

country18

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

country2

country2 example vector

Description

country2 example vector

Usage

country2

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

country3

country3 example vector

Description

country3 example vector

Usage

country3

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

country313

country313 example vector

Description

country313 example vector

Usage

country313

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

country32	<i>country32 example vector</i>
-----------	---------------------------------

Description

country32 example vector

Usage

```
country32
```

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

ctmaAllInvFit	<i>ctmaAllInvFit</i>
---------------	----------------------

Description

#' @description Fit a CoTiMA model with all params (drift, T0var, diffusion) invariant across primary studies

Usage

```
ctmaAllInvFit(
  ctmaInitFit = NULL,
  activeDirectory = NULL,
  activateRPB = FALSE,
  digits = 4,
  drift = drift,
  coresToUse = c(1),
  n.manifest = 0,
  indVarying = FALSE,
  scaleTime = NULL,
  optimize = TRUE,
  nopriors = TRUE,
  finishsamples = NULL,
  iter = NULL,
  chains = NULL,
  verbose = NULL,
  loadAllInvFit = c(),
```

```

saveAllInvFit = c(),
silentOverwrite = FALSE,
customPar = TRUE
)

```

Arguments

ctmaInitFit	ctmaInitFit
activeDirectory	activeDirectory
activateRPB	activateRPB
digits	digits
drift	Labels for drift effects. Have to be either of the type V1toV2 or 0 for effects to be excluded, which is usually not recommended)
coresToUse	coresToUse
n.manifest	Number of manifest variables of the model (if left empty it will assumed to be identical with n.latent).
indVarying	Allows ct intercepts to vary at the individual level (random effects model, accounts for unobserved heterogeneity)
scaleTime	scaleTime
optimize	optimize
nopriors	nopriors
finishesamples	finishesamples
iter	iter
chains	chains
verbose	verbose
loadAllInvFit	loadAllInvFit
saveAllInvFit	saveAllInvFit
silentOverwrite	silentOverwrite
customPar	logical. Leverages the first pass using priors and ensure that the drift diagonal cannott easily go too negative (could help with ctsem > 3.4)

Value

returns a fitted CoTiMA object, in which all drift parameters, Time 0 variances and covariances, and diffusion parameters were set invariant across primary studies

ctmaBiG	<i>ctmaBiG</i>
---------	----------------

Description

Analysis of publication bias and generalizability. The function takes a CoTiMA fit object (created with `ctmaInit`) and estimates fixed and random effects of single drift coefficients, heterogeneity (Q, I square, H square, tau square), PET-PEESE corrections, Egger's tests, and z-curve analysis yielding expected replication and detection rates (ERR, EDR).

Usage

```
ctmaBiG(
  ctmaInitFit = NULL,
  activeDirectory = NULL,
  PETPEESEalpha = 0.1,
  activateRPB = FALSE,
  digits = 4,
  zcurve = FALSE
)
```

Arguments

<code>ctmaInitFit</code>	fit object created with <code>ctmaInit</code> containing the fitted ctsem model of each primary study
<code>activeDirectory</code>	the directory where to save results (if not specified, it is taken from <code>ctmaInitFit</code>)
<code>PETPEESEalpha</code>	probability level (condition) below which to switch from PET to PEESE (cf. Stanley, 2017, p. 582, below Eq. 2; default $p = .10$)
<code>activateRPB</code>	if TRUE, messages (warning, finished) could be send to smart phone (default = FALSE)
<code>digits</code>	rounding (default = 4)
<code>zcurve</code>	performs z-curve analysis. Could fail if too few studies (e.g. around 10) are supplied. default=FALSE

Value

`ctmaBiG` returns a list containing some arguments supplied, the results of analyses of publication bias and generalizability, model type, and the type of plot that could be performed with the returned object. The arguments in the returned object are `activeDirectory`, and `coresToUse`. Further arguments, which are just copied from the init-fit object supplied, are, `n.studies`, `n.latent`, `studyList`, `statisticsList`, `modelResults` (all parameter estimates and their standard error), and parameter names. All new results are returned as the list element "summary", which is printed if the summary function is applied to the returned object. The summary list element comprises a title (model='Analysis of Publication Bias & Generalizability') and "estimates", which is another list comprising "Fixed Effects of Drift Coefficients", "Heterogeneity", "Random Effects of Drift Coefficients", "PET-PEESE

corrections", "Egger's tests" (constant of the WLS regression of drift coefficients on their standard errors (SE) with $1/SE^2$ as weights), "Egger's tests Alt. Version" (constant of the OLS regression of the standard normal deviates of the drift coefficients on their precision), and "Z-Curve 2.0 Results". Plot type is `plot.type=c("funnel", "forest")` and `model.type="BiG"`.

Examples

```
## Not run:
# perform analyses of publication bias and generalizability
CoTiMABiG_D_BO$activeDirectory <- "/Users/tmp/" # adapt!
CoTiMABiG_D_BO <- ctmaBiG(ctmaInitFit=CoTiMABiG_D_BO, zcurve=FALSE)

## End(Not run)

# display results
summary(CoTiMABiG_D_BO)

## Not run:
# get funnel & forest plots
CoTiMABiG_D_BO$activeDirectory <- "/Users/tmp/" # adapt!
plot(CoTiMABiG_D_BO)

## End(Not run)
```

ctmaBiGOMX

ctmaBiGOMX

Description

Analysis of publication bias and fixed and random effects analysis of single drift coefficients if OLD OpenMx fit files are supplied

Usage

```
ctmaBiGOMX(
  ctmaInitFit = NULL,
  activeDirectory = NULL,
  PETPEESEalpha = 0.1,
  activateRPB = FALSE,
  digits = 4
)
```

Arguments

`ctmaInitFit` fit object created with `ctmaInti` containing the fitted `ctsem` model of each primary study

`activeDirectory` the directory where to save results (if not specified, it is taken from `ctmaInitFit`)

PETPEESEalpha	# probability level (condition) below which to switch from PET to PEESE (Stanley, 2017, SPPS,p. 582, below Eq. 2; (default p = .10)
activateRPB	if TRUE, messages (warning, finishes) could be send to smart phone (default = FALSE)
digits	rounding (default = 4)

Value

returns a CoTiMA fit object with results of publication bias analysis, fixed and random effect analysis, Egger's tests, PET-PEESE corrections.

ctmaCombPRaw	<i>ctmaCombPRaw</i>
--------------	---------------------

Description

Combine Pseudo Raw Data (extract them from 'CoTiMAFit object'\$studyFitList)

Usage

```
ctmaCombPRaw(listOfStudyFits = NULL, moderatorValues = NULL)
```

Arguments

listOfStudyFits	"Listobject of Studyfits"
moderatorValues	"Moderators"

Value

returns a pseudo raw data set that combines pseudo raw data and moderators of primary studies

ctmaCompFit	<i>ctmaCompFit</i>
-------------	--------------------

Description

Performs log-likelihood ratio tests to compare the fit of 2 models (CoTiMAFit objects created with [ctmaFit](#) or [ctmaEqual](#)), i.e., the difference between the two -2 times LLs between the first model and the more constrained second model. The nested structure of the two models is assumed to be given and not checked.

Usage

```
ctmaCompFit(model1 = NULL, model2 = NULL)
```

Arguments

model1	Model 1
model2	Model 2

Value

Returns the the difference between the two -2 times LLs (Diff_Minus2LL), the associated difference in degrees of freedom (Diff_df (= Diff_n.params)), and the probability (prob).

Examples

```
minus2l1DiffTest <- ctmaCompFit(CoTiMAFullInv23Fit_6,
                               CoTiMAFullInvEq23Fit_6)
summary(minus2l1DiffTest)
```

 ctmaCorRel

ctmaCorRel

Description

Disattenuates the entries in a correlation matrix using a vector of reliabilities.

Usage

```
ctmaCorRel(empcov = NULL, alphas = NULL)
```

Arguments

empcov	Empirical correlation matrix
alphas	Vector reliabilities

Value

A corrected correlation matrix (corEmpcov). Corrections leading to $r > 1.0$ are set to 1.0.

Examples

```
empcov313new <- ctmaCorRel(empcov=empcov313, alphas=alphas313)
```

ctmaEmpCov	<i>ctmaEmpCov</i>
------------	-------------------

Description

changes a full covariance matrix by selecting target variables, recoding them, combining them (compute the mean of two or more variables), and by adding rows/columns with NA if focal variables are not available.

Usage

```
ctmaEmpCov(
  targetVariables = NULL,
  recodeVariables = c(),
  combineVariables = c(),
  combineVariablesNames = c(),
  missingVariables = c(),
  nlatents = NULL,
  Tpoints = NULL,
  sampleSize = NULL,
  pairwiseN = NULL,
  empcov = NULL
)
```

Arguments

targetVariables	(col-/row-) number or names of the target variables
recodeVariables	(col-/row-) number or names of the target variables require inverse coding
combineVariables	list of vectors, which put together the targeted variables that should be used for composite variables
combineVariablesNames	new names for combined variables - not really important
missingVariables	missing variables
nlatents	number of (latent) variables - actually it is the number of all variables
Tpoints	number of time points.
sampleSize	sample size
pairwiseN	matrix of same dimensions as empcov containing possible pairwiseN.
empcov	empirical correlation matrix

Value

returns a list with two elements. The first element (results\$r) contains the adapted correlation matrix, and the second element (results\$pairwiseNNew) an adapted version of a matrix of pairwise N if pairwiseN was provided for the original correlation matrix supplied.

Examples

```
source17 <- c()
delta_t17 <- c(12)
sampleSize17 <- 440
empcov17 <- matrix(
  c( 1.00, -0.60, -0.36,  0.20,  0.62, -0.47, -0.18,  0.20,
    -0.60,  1.00,  0.55, -0.38, -0.43,  0.52,  0.27, -0.21,
    -0.36,  0.55,  1.00, -0.47, -0.26,  0.37,  0.51, -0.28,
    0.20, -0.38, -0.47,  1.00,  0.15, -0.28, -0.35,  0.56,
    0.62, -0.43, -0.26,  0.15,  1.00, -0.63, -0.30,  0.27,
    -0.47,  0.52,  0.37, -0.28, -0.63,  1.00,  0.55, -0.37,
    -0.18,  0.27,  0.51, -0.35, -0.30,  0.55,  1.00, -0.51,
    0.20, -0.21, -0.28,  0.56,  0.27, -0.37, -0.51,  1.00),
  nrow=8, ncol=8)
moderator17 <- c(3, 2)
rownames(empcov17) <- colnames(empcov17) <-
  c("Workload_1", "Exhaustion_1", "Cynicism_1", "Values_1",
    "Workload_2", "Exhaustion_2", "Cynicism_2", "Values_2")
targetVariables17 <-
  c("Workload_1", "Exhaustion_1", "Cynicism_1",
    "Workload_2", "Exhaustion_2", "Cynicism_2")
recodeVariables17 <- c("Workload_1", "Workload_2")
combineVariables17 <- list("Workload_1", c("Exhaustion_1", "Cynicism_1"),
  "Workload_2", c("Exhaustion_2", "Cynicism_2"))
combineVariablesNames17 <- c("Demands_1", "Burnout_1",
  "Demands_2", "Burnout_2")
missingVariables17 <- c();
results17 <- ctmaEmpCov(targetVariables = targetVariables17,
  recodeVariables = recodeVariables17,
  combineVariables = combineVariables17,
  combineVariablesNames = combineVariablesNames17,
  missingVariables = missingVariables17,
  nlatents = 2, sampleSize = sampleSize17,
  Tpoints = 2, empcov = empcov17)
empcov17 <- results17$r
```

ctmaEqual

ctmaEqual

Description

test if the two or more invariant drift parameters in the CoTiMAFit object supplied are equal. The supplied CoTiMA fit-object (ctmaInvariantFit) has to be a model fitted with [ctmaFit](#) where at least

two parameters were set invariant across primary studies (e.g., 2 cross effects). All parameters that are set invariant in the supplied model are then constrained to be equal by `ctmaEqual` (no user action required), the model is fitted, and a log-likelihood ratio test is performed informing about the probability that equality applies.

Usage

```
ctmaEqual(
  ctmaInvariantFit = NULL,
  activeDirectory = NULL,
  activateRPB = FALSE,
  digits = 4,
  coresToUse = 1
)
```

Arguments

<code>ctmaInvariantFit</code>	object to which a CoTiMA fit has been assigned to (i.e., what has been returned by <code>ctmaFit</code>). In most cases probably a model in which (only) two effects were specified with <code>invariantDrift</code> .
<code>activeDirectory</code>	defines another active directory than the one used in <code>ctmaInvariantFit</code>
<code>activateRPB</code>	set to TRUE to receive push messages with CoTiMA notifications on your phone
<code>digits</code>	Number of digits used for rounding (in outputs)
<code>coresToUse</code>	If neg., the value is subtracted from available cores, else value = cores to use

Value

returns a model where two or more parameters were set equal across primary studies and a log-likelihood difference test informing about the probability that the equality assumption is correct.

Examples

```
# Fit a CoTiMA with a set of parameters set equal that were set
# invariant in a previous model (of which the fit object is
# supplied in argument ctmaInvariantFit)
## Not run:
CoTiMAFullInv23Fit_6$activeDirectory <- "/Users/tmp/" # adapt!
CoTiMAFullInvEq23Fit_6 <- ctmaEqual(ctmaInvariantFit=CoTiMAFullInv23Fit_6)

## End(Not run)
```

ctmaFit	<i>ctmaFit</i>
---------	----------------

Description

Fits a ctsem model with invariant drift effects across primary studies, possible multiple moderators (but all of them of the the same type, either "cont" or "cat"), and possible cluster (e.g., countries where primary studies were conducted).

Usage

```
ctmaFit(  
  ctmaInitFit = NULL,  
  primaryStudyList = NULL,  
  cluster = NULL,  
  activeDirectory = NULL,  
  activateRPB = FALSE,  
  digits = 4,  
  drift = NULL,  
  invariantDrift = NULL,  
  moderatedDrift = NULL,  
  equalDrift = NULL,  
  mod.number = NULL,  
  mod.type = "cont",  
  mod.names = NULL,  
  indVarying = FALSE,  
  coresToUse = c(1),  
  scaleTI = NULL,  
  scaleMod = NULL,  
  scaleClus = NULL,  
  scaleTime = NULL,  
  optimize = TRUE,  
  nopriors = TRUE,  
  finishsamples = NULL,  
  iter = NULL,  
  chains = NULL,  
  verbose = NULL,  
  allInvModel = FALSE,  
  customPar = FALSE,  
  inits = NULL,  
  modsToCompare = NULL,  
  catsToCompare = NULL,  
  driftsToCompare = NULL  
)
```

Arguments

ctmaInitFit	object to which all single ctsem fits of primary studies has been assigned to (i.e., what has been returned by <code>ctmaInit</code>)
primaryStudyList	could be a list of primary studies compiled with <code>ctmaPrep</code> that defines the subset of studies in <code>ctmaInitFit</code> that should actually be used
cluster	vector with cluster variables (e.g., countries). Has to be set up carefully. Will be included in <code>ctmaPrep</code> in later 'CoTiMA' versions.
activeDirectory	defines another active directory than the one used in <code>ctmaInitFit</code>
activateRPB	set to TRUE to receive push messages with 'CoTiMA' notifications on your phone
digits	Number of digits used for rounding (in outputs)
drift	labels for drift effects. Have to be either of the type 'V1toV2' or '0' for effects to be excluded.
invariantDrift	drift labels for drift effects that are set invariant across primary studies (default = all drift effects).
moderatedDrift	labels for drift effects that are moderated (default = all drift effects)
equalDrift	Not enabled
mod.number	which in the vector of moderator values shall be used (e.g., 2 for a single moderator or 1:3 for 3 moderators simultaneously)
mod.type	'cont' or 'cat' (mixing them in a single model not yet possible)
mod.names	vector of names for moderators used in output
indVarying	allows continuous time intercepts to vary at the individual level (random effects model, accounts for unobserved heterogeneity)
coresToUse	if negative, the value is subtracted from available cores, else value = cores to use
scaleTI	scale TI predictors - not recommended if TI are dummies representing primary studies, which would be the usual case
scaleMod	scale moderator variables - FALSE (default) highly recommended for categorical moderators, TRUE highly recommended for continuous moderators
scaleClus	scale vector of cluster indicators - TRUE (default) yields avg. drift estimates, FALSE yields drift estimates of last cluster
scaleTime	scale time (interval) - sometimes desirable to improve fitting
optimize	if set to FALSE, Stan's Hamiltonian Monte Carlo sampler is used (default = TRUE = maximum a posteriori / importance sampling) .
nopriors	if TRUE, any priors are disabled – sometimes desirable for optimization
finishesamples	number of samples to draw (either from hessian based covariance or posterior distribution) for final results computation (default = 1000).
iter	number of iterations (default = 1000). Sometimes larger values could be required from Bayesian estimation
chains	number of chains to sample, during HMC or post-optimization importance sampling.

verbose	integer from 0 to 2. Higher values print more information during model fit – for debugging
allInvModel	estimates a model with all parameters invariant (DRIFT, DIFFUSION, T0VAR) if set TRUE (default = FALSE)
customPar	logical. Leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (could help with ctsem > 3.4)
inits	vector of start values
modsToCompare	when performing contrasts for categorical moderators, the moderator numbers (position in mod.number) that is used
catsToCompare	when performing contrasts for categorical moderators, the categories (values, not positions) for which effects are set equal
driftsToCompare	when performing contrasts for categorical moderators, the (subset of) drift effects analyzed

Value

ctmaFit returns a list containing some arguments supplied, the fitted model, different elements summarizing the main results, model type, and the type of plot that could be performed with the returned object. The arguments in the returned object are activeDirectory, coresToUse, moderator names (mod.names), and moderator type (mod.type). Further arguments, which are just copied from the init-fit object supplied, are, n.latent, studyList, parameterNames, and statisticsList. The fitted model is found in studyFitList, which is a large list with many elements (e.g., the ctsem model specified by CoTiMA, the rstan model created by ctsem, the fitted rstan model etc.). Further results returned are n.studies = 1 (required for proper plotting), data (created pseudo raw data), and a list with modelResults (i.e., DRIFT=model_Drift_Coeff, DIFFUSION=model_Diffusion_Coeff, T0VAR=model_T0var_Coeff, CINT=model_Cint_Coeff, MOD=modTI_Coeff, and CLUS=clusTI_Coeff). Possible invariance constraints are included in invariantDrift. The number of moderators simultaneously analyzed are included in 'n.moderators'. The most important new results are returned as the list element "summary", which is printed if the summary function is applied to the returned object. The summary list element comprises "estimates" (the aggregated effects), possible randomEffects (not yet fully working), the minus2ll value and its n.parameters, the opt.lag sensu Dormann & Griffin (2015) and the max.effects that occur at the opt.lag, clus.effects and mod.effects, and possible warning messages (message). Plot type is plot.type=c("drift") and model.type="stanct" ("omx" was deprecated).

Examples

```
# Example 1. Fit a CoTiMA to all primary studies previously fitted one by one
# with the fits assigned to CoTiMAInitFit_6
CoTiMAFullFit_6 <- ctmaFit(ctmaInitFit=CoTiMAInitFit_6)
summary(CoTiMAFullFit_6)

## Not run:
# Example 2. Fit a CoTiMA with only 2 cross effects invariant (not the auto
# effects) to all primary studies previously fitted one by one with the fits
```



```

# assigned to CoTiMAInitFit_6
CoTiMAInitFit_6$activeDirectory <- "/Users/tmp/" # adapt!
CoTiMAFullInv23Fit_6 <- ctmaFit(ctmaInitFit=CoTiMAInitFit_6,
                               invariantDrift=c("V1toV2", "V2toV1"))
summary(CoTiMAFullInv23Fit_6)

## End(Not run)

## Not run:
# Example 3. Fit a moderated CoTiMA
CoTiMAInitFit_6$activeDirectory <- "/Users/tmp/" # adapt!
CoTiMAMod1onFullFit_6 <- ctmaFit(ctmaInitFit=CoTiMAInitFit_6,
                                mod.number=1, mod.type="cont",
                                mod.names=c("Control"))
summary(CoTiMAMod1onFullFit_6)

## End(Not run)

```

ctmaFitList

ctmaFitList

Description

Combines CoTiMAFit objects into a list with class CoTiMAFit to inform generic functions what to do

Usage

```
ctmaFitList(...)
```

Arguments

... any number of CoTiMAFit objects

Value

a list that combines all objects supplied and is assigned the class 'CoTiMAFit'

Examples

```

## Not run:
CoTiMAInitFit_3$activeDirectory <- "/Users/tmp/" # adapt!
CoTiMAFullFit_3$activeDirectory <- "/Users/tmp/" # adapt!
plot(ctmaFitList(CoTiMAInitFit_3, CoTiMAFullFit_3),
     timeUnit="Months",
     timeRange=c(1, 144, 1) )

## End(Not run)

```

ctmaFitToPrep	<i>ctmaFitToPrep</i>
---------------	----------------------

Description

Extracts information from fitted CoTiMA objects to (re-)create list of primary studies originally created with [ctmaPrep](#)

Usage

```
ctmaFitToPrep(ctmaFitObject = NULL)
```

Arguments

ctmaFitObject ctmaFitObject

Value

list that could be used for fitting new CoTiMA models with [ctmaInit](#) or [ctmaFit](#).

Examples

```
newStudyList <- ctmaFitToPrep(CoTiMAInitFit_3)
```

ctmaGetPub	<i>ctmaGetPub</i>
------------	-------------------

Description

Retrieves publication and citation information from google scholar based on the supplied author names and their google ID (user)

Usage

```
ctmaGetPub(authorList = NULL, flush = FALSE, yearsToExclude = NULL)
```

Arguments

authorList list of authors and googe scholar addresses
flush if TRUE, the cache will be cleared and the data reloaded from Google.
yearsToExclude the years to be excluded (default = current year)

Value

list with (cumulative) frequencies and (cumulative) citations in google scholar

Note

Set flush=TRUE only if retrieving is necessary (e.g., first retrieval on a day)

Examples

```
pubList_8 <- ctmaGetPub(authorList = list( c("J; de Jonge",
  "https://scholar.google.de/citations?hl=de&user=0q27IckAAAAJ"),
  c("Arnold B.; Bakker", "user=FT13bwUAAAAJ"),
  c("Evangelia; Demerouti", "user=9mj5LvMAAAAJ"),
  c("Joachim; Stoeber", "user=T9xdVusAAAAJ"),
  c("Claude; Fernet", "user=KwzjP4sAAAAJ"),
  c("Frederic; Guay", "user=99vnhX4AAAAJ"),
  c("Caroline; Senecal", "user=64ArFWQAAAAJ"),
  c("St  phanie; Austin", "user=PPyTI7EAAAAJ")),
  flush=FALSE)
summary(pubList_8)
```

ctmaInit

ctmaInit

Description

Fits ctsem models to each primary study in the supplied list of primary studies prepared by [ctmaPrep](#).

Usage

```
ctmaInit(
  primaryStudies = NULL,
  activeDirectory = NULL,
  activateRPB = FALSE,
  checkSingleStudyResults = TRUE,
  digits = 4,
  n.latent = NULL,
  n.manifest = 0,
  lambda = NULL,
  manifestVars = NULL,
  drift = NULL,
  indVarying = FALSE,
  saveRawData = list(),
  coresToUse = c(1),
  silentOverwrite = FALSE,
  saveSingleStudyModelFit = c(),
  loadSingleStudyModelFit = c(),
  scaleTI = NULL,
  scaleTime = NULL,
```

```

optimize = TRUE,
nopriors = TRUE,
finishsamples = NULL,
chains = NULL,
iter = NULL,
verbose = NULL,
customPar = FALSE,
doPar = 1,
useSV = TRUE
)

```

Arguments

primaryStudies list of primary study information created with [ctmaPrep](#)

activeDirectory defines another active directory than the one used in [ctmaPrep](#)

activateRPB set to TRUE to receive push messages with 'CoTiMA' notifications on your phone

checkSingleStudyResults Displays estimates from single study ctsem models and waits for user input to continue. Useful to check estimates before they are saved.

digits number of digits used for rounding (in outputs)

n.latent number of latent variables of the model (has to be specified)!

n.manifest number of manifest variables of the model (if left empty it will assumed to be identical with n.latent).

lambda R-type matrix with pattern of fixed (=1) or free (any string) loadings.

manifestVars define the error variances of the manifests with a single time point using R-type matrix with nrow=n.manifest & ncol=n.manifest.

drift labels for drift effects. Have to be either of the type V1toV2 or 0 for effects to be excluded, which is usually not recommended)

indVarying control for unobserved heterogeneity by having randomly (inter-individually) varying manifest means

saveRawData save (created pseudo) raw data. List: saveRawData\$studyNumbers, \$fileName, \$row.names, col.names, \$sep, \$dec

coresToUse if neg., the value is subtracted from available cores, else value = cores to use

silentOverwrite overwrite old files without asking

saveSingleStudyModelFit save the fit of single study ctsem models (could save a lot of time afterwards if the fit is loaded)

loadSingleStudyModelFit load the fit of single study ctsem models

scaleTI scale TI predictors

scaleTime scale time (interval) - sometimes desirable to improve fitting

optimize	if set to FALSE, Stan's Hamiltonian Monte Carlo sampler is used (default = TRUE = maximum a posteriori / importance sampling) .
nopriors	if TRUE, any priors are disabled - sometimes desirable for optimization
finishesamples	number of samples to draw (either from hessian based covariance or posterior distribution) for final results computation (default = 1000).
chains	number of chains to sample, during HMC or post-optimization importance sampling.
iter	number of iteration (default = 1000). Sometimes larger values could be required from Bayesian estimation
verbose	integer from 0 to 2. Higher values print more information during model fit - for debugging
customPar	logical. Leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (could help with ctsem > 3.4)
doPar	parallel and multiple fitting if single studies
useSV	if TRUE (default) start values will be used if provided in the list of primary studies

Value

ctmaFit returns a list containing some arguments supplied, the fitted models, different elements summarizing the main results, model type, and the type of plot that could be performed with the returned object. The arguments in the returned object are activeDirectory, coresToUse, n.latent, n.manifest, and primaryStudyList. The study count is returned as n.studies, the created matrix of loadings of manifest on latent factors is returned as lambda, and a re-organized list of primary studies with some information omitted is returned as studyList. The fitted models for each primary study are found in studyFitList, which is a large list with many elements (e.g., the ctsem model specified by CoTiMA, the rstan model created by ctsem, the fitted rstan model etc.). Further results returned are emprowList (containing the pseudo raw data created), statisticsList (comprising basic stats such as average sample size, no. of measurement points, etc.), a list with modelResults (i.e., DRIFT=model_Drift_Coef, DIFFUSION=model_Diffusion_Coef, T0VAR=model_T0var_Coef, CINT=model_Cint_Coef), and the parameter names internally used. The summary list, which is printed if the summary function is applied to the returned object, comprises "estimates" (the aggregated effects), possible randomEffects (not yet fully working), confidenceIntervals, the minus2ll value and its n.parameters, and possible warning messages (message). Plot type is plot.type=c("drift") and model.type="stanc" ("omx" was deprecated).

Examples

```
# Fit a ctsem model to all three primary studies summarized in
# CoTiMAstudyList_3 and save the three fitted models
## Not run:
CoTiMAInitFit_3 <- ctmaInit(primaryStudies=CoTiMAstudyList_3,
                           n.latent=2,
                           checkSingleStudyResults=FALSE,
                           activeDirectory="/Users/tmp/") # adapt!

summary(CoTiMAInitFit_3)
```

```
## End(Not run)
```

```
ctmaLabels
```

```
ctmaLabels
```

Description

used for consistent labeling of names and parameters

Usage

```
ctmaLabels(
  n.latent = NULL,
  n.manifest = 0,
  lambda = NULL,
  manifestVar = NULL,
  drift = NULL,
  invariantDrift = NULL,
  moderatedDrift = NULL,
  equalDrift = NULL
)
```

Arguments

n.latent	n.latent
n.manifest	n.manifest
lambda	lambda
manifestVar	manifestVar
drift	drift
invariantDrift	invariantDrift
moderatedDrift	moderatedDrift
equalDrift	equalDrift

Value

returns consistently named parameters (e.g., "V1toV2") as well as their symbolic values, which are used to fix or free parameters when fitting a 'CoTiMA' model

ctmaOptimizeInit *ctmaOptimizeInit*

Description

Initial fitting (i.e., applies `ctmaInit`) to a primary study reFit times to capitalize on chance for obtaining a hard-to-find optimal fit. This could be very helpful if a primary yields out-of-range estimates, which could happen if the fitting algorithm unfortunately used random start values that resulted in a locally but not globally optimal fit. Essentially, using `ctmaOptimizeInit` is like gambling, hoping that at least one set of starting values (the number of tries is specified in the `reFits` argument) enables finding the global optimal fit. On unix-like machines (e.g. MacOS), this could be done in parallel mode if `coresToUse > 1`.

Usage

```
ctmaOptimizeInit(
  primaryStudies = NULL,
  activeDirectory = NULL,
  problemStudy = NULL,
  reFits = NULL,
  n.latent = NULL,
  coresToUse = c(1),
  activateRPB = FALSE,
  checkSingleStudyResults = FALSE,
  customPar = FALSE
)
```

Arguments

<code>primaryStudies</code>	list of primary study information created with <code>ctmaPrep</code> or <code>ctmaFitToPrep</code>
<code>activeDirectory</code>	<code>activeDirectory</code>
<code>problemStudy</code>	number (position in list) where the problem study in <code>primaryStudies</code> is found
<code>reFits</code>	how many reFits should be done
<code>n.latent</code>	number of latent variables of the model (has to be specified)!
<code>coresToUse</code>	if neg., the value is subtracted from available cores, else value = cores to use
<code>activateRPB</code>	set to TRUE to receive push messages with 'CoTiMA' notifications on your phone
<code>checkSingleStudyResults</code>	displays estimates from single study 'ctsem' models and waits for user input to continue. Useful to check estimates before they are saved.
<code>customPar</code>	logical. Leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (could help with <code>ctsem > 3.4</code>)

Value

returns a list with bestFit (= the best fit achieved), all_minus2ll (= all -2ll values for all fitted models), and summary, which is printed if the summary function is applied to the returned object, and which shows the summary information of the ctsem model with the best fit.

Note

All but one of multiple cores are used on unix-type machines for parallel fitting
During fitting, not output is generated. Be patient.

Examples

```
## Not run:
optimFit313 <- ctmaOptimizeInit(primaryStudies=CoTiMAstudyList_3,
                              activeDirectory="/Users/tmp/", # adapt!
                              problemStudy=which(CoTiMAstudyList_3$studyNumbers == 313),
                              reFits=10,
                              n.latent=2)

summary(optimFit313)

## End(Not run)
```

ctmaPlot

ctmaPlot

Description

Forest plot, funnel plots, plots of discrete time cross-lagged and autoregressive effect, and plots of required sample sizes

Usage

```
ctmaPlot(
  ctmaFitObject = NULL,
  activeDirectory = NULL,
  saveFilePrefix = "ctmaPlot",
  activateRPB = FALSE,
  plotCrossEffects = TRUE,
  plotAutoEffects = TRUE,
  timeUnit = "timeUnit (not specified)",
  timeRange = c(),
  yLimitsForEffects = c(),
  mod.number = 1,
  mod.values = -2:2,
  aggregateLabel = "",
  xLabels = NULL,
  ...
)
```


Arguments

ctmaFitObject 'CoTiMA' Fit object
 activeDirectory defines another active directory than the one used in ctmaInitFit
 saveFilePrefix Prefix used for saved plots
 activateRPB set to TRUE to receive push messages with 'CoTiMA' notifications on your phone
 plotCrossEffects logical
 plotAutoEffects logical
 timeUnit label for x-axis when plotting discrete time plots
 timeRange vector describing the time range for x-axis as sequence from/to/stepSize (e.g., c(1, 144, 1))
 yLimitsForEffects range for y-axis
 mod.number moderator number that should be used for plots
 mod.values moderator values that should be used for plots
 aggregateLabel label to indicate aggregated discrete time effects
 xLabels labels used for x-axis
 ... arguments passed through to plot()

Value

depending on the CoTiMA fit object supplied, generates funnel plots, forest plots, discrete time plots of autoregressive and cross-lagged effects, plots of required samples sizes across a range of discrete time intervals to achieve desired levels of statistical power, and post hoc power of primary studies. Plots are saved to disk.

Examples

```

## Not run:
# cannot run without proper activeDirectory specified. Adapt!
CoTiMAFullFit_3$activeDirectory <- "/Users/tmp/" # adapt!
plot(ctmaFitList(CoTiMAInitFit_3, CoTiMAFullFit_3),
     timeUnit="Months", timeRange=c(1, 144, 1),
     plotAutoEffects=FALSE)

## End(Not run)

## Not run:
# cannot run without proper activeDirectory specified. Adapt!
CoTiMABiG_D_B0$activeDirectory <- "/Users/tmp/" # adapt!
plot(CoTiMABiG_D_B0)

## End(Not run)

```

 ctmaPower

ctmaPower

Description

Fits a full invariant model to a list of primary studies and performs analyses of expected (post hoc) power and required sample sizes.

Usage

```
ctmaPower(
  ctmaInitFit = NULL,
  activeDirectory = NULL,
  statisticalPower = c(),
  failSafeN = NULL,
  failSafeP = NULL,
  timeRange = NULL,
  useMBESS = FALSE,
  coresToUse = 1,
  digits = 4,
  indVarying = FALSE,
  activateRPB = FALSE,
  silentOverwrite = FALSE,
  loadAllInvFit = c(),
  saveAllInvFit = c(),
  loadAllInvWOSingFit = c(),
  saveAllInvWOSingFit = c(),
  skipScaling = TRUE,
  useSampleFraction = NULL,
  optimize = TRUE,
  nopriors = TRUE,
  finishsamples = NULL,
  iter = NULL,
  chains = NULL,
  verbose = NULL,
  customPar = FALSE
)
```

Arguments

ctmaInitFit object to which all single 'ctsem' fits of primary studies has been assigned to (i.e., what has been returned by [ctmaInit](#))

activeDirectory defines another active directory than the one used in [ctmaInit](#)

statisticalPower vector of requested statistical power values

failSafeN	sample size used to determine across which time intervals effects become non-significant
failSafeP	p-value used to determine across which time intervals effects become non-significant
timeRange	vector describing the time range for x-axis as sequence from/to/stepSize (e.g., c(1, 144, 1))
useMBESS	use 'MBESS' package to calculate statistical power (slower)
coresToUse	if negative, the value is subtracted from available cores, else value = cores to use
digits	number of digits used for rounding (in outputs)
indVarying	Allows continuous time intercepts to vary at the individual level (random effects model, accounts for unobserved heterogeneity)
activateRPB	set to TRUE to receive push messages with 'CoTiMA' notifications on your phone
silentOverwrite	overwrite old files without asking
loadAllInvFit	load the fit of fully constrained 'CoTiMA' model
saveAllInvFit	save the fit of fully constrained 'CoTiMA' model
loadAllInvWOSingFit	load series of fits of fully constrained 'CoTiMA' model with single cross effects excluded, respectively
saveAllInvWOSingFit	save series of fits of fully constrained 'CoTiMA' model with single cross effects excluded, respectively
skipScaling	does not (re-)scale raw data (re-scaling of imported pseudo raw data achieves correlations = 1)
useSampleFraction	to speed up debugging. Provided as fraction (e.g., 1/10)
optimize	if set to FALSE, Stan's Hamiltonian Monte Carlo sampler is used (default = TRUE = maximum a posteriori / importance sampling) .
nopriors	if TRUE, any priors are disabled – sometimes desirable for optimization
finishsamples	number of samples to draw (either from hessian based covariance or posterior distribution) for final results computation (default = 1000).
iter	number of iterations (default = 1000). Sometimes larger values could be required from Bayesian estimation
chains	number of chains to sample, during HMC or post-optimization importance sampling.
verbose	integer from 0 to 2. Higher values print more information during model fit – for debugging
customPar	logical. Leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (could help with ctsem > 3.4)

Value

ctmaPower returns a list containing some arguments supplied, a fitted model with all (!) parameters invariant across primary studies, different elements summarizing the main results, model type, and the type of plot that could be performed with the returned object. The arguments in the returned object are activeDirectory, coresToUse, n.latent, n.manifest, and primaryStudyList. A further result returned is n.studies = 1 (required for proper plotting). Further arguments, which are just copied from the init-fit object supplied, are, n.latent, studyList, and the statisticsList. The fitted model is found in studyFitList, which is a large list with many elements (e.g., the ctsem model specified by CoTiMA, the rstan model created by ctsem, the fitted rstan model etc.). Further results returned are a list with modelResults (i.e., DRIFT=DRIFT, DIFFUSION=DIFFUSION, T0VAR=T0VAR, CINT=NULL) and the parameter names internally used. The summary list, which is printed if the summary function is applied to the returned object, contains "estimates", which is itself a list comprising "Estimates of Model with all Effects Invariant", "Requested Statistical Power" (which just returns the argument statisticalPower), "Power (post hoc) for Drift Effects", "Required Sample Sizes" "Effect Sizes (based on discrete-time calcs; used for power calcs.)", and "Range of significant effects" (across which intervals effects were significant). Plot type is plot.type=c("power") and model.type="stanc" ("omx" was deprecated).

Examples

```
## Not run:
CoTiMAInitFit_D_B0$activeDirectory <- "/Users/tmp/" # adapt!
CoTiMAPower_D_B0 <- ctmaPower(ctmaInitFit=CoTiMAInitFit_D_B0,
                             statisticalPower = c(.50, .80, .95),
                             finishsamples = 10000)

summary(CoTiMAPower_D_B0)

## End(Not run)
```

ctmaPraw

ctmaPraw

Description

Converts empirical correlation matrices to pseudo raw data (i.e. random data, that perfectly reproduce the correlations)

Usage

```
ctmaPraw(
  empCovMat = NULL,
  empNMat = matrix(0, 0, 0),
  empN = NULL,
  studyNumber = NULL,
  empMeanVector = NULL,
  empVarVector = NULL,
  activateRPB = FALSE
)
```

Arguments

empCovMat	empirical primary study covariance matrix
empNMat	matrix of (possibly pairwise) N
empN	N (in case of listwise N)
studyNumber	internal number
empMeanVector	vector of means for all variables, usually 0
empVarVector	vector of variances for all variables, usually 1
activateRPB	set TRUE to receive push messages with 'CoTiMA' notifications on your phone

 ctmaPrep

ctmaPrep

Description

Combines information of primary studies into a list object and returns this list. This list is then used as input to fit 'ctsem' models. Primary study information is expected to be assigned to 'numbered' objects. Some of these objects are pre-defined (e.g., 'empcov', 'ageM'). Most of the pre-defined objects could be empty, or they could be dropped by entering their names in the excludedElements-object (e.g., excludedElements = c('ageM')), but dropping them is not really necessary. Additional elements could also be added, which could be useful to put together all information about primary studies at the convenience of the researcher.

Usage

```
ctmaPrep(
  selectedStudies = NULL,
  excludedElements = NULL,
  addElements = NULL,
  digits = 4,
  moderatorLabels = NULL,
  moderatorValues = NULL,
  summary = TRUE
)
```

Arguments

selectedStudies	Vector of primary study numbers (numeric values with no leading 0; e.g., '2' but not '02')
excludedElements	Vector of predefined objects used to code primary study information. Some predefined objects are strongly defined; they have to be used in a special way because they are actually used in subsequent analyses. Some other objects could be used at the researcher's convenience (information is just collected). Strongly predefined objects are 'delta_t' (vector of time intervals; the only mandatory

requirement; should be of the type `c(NA, NA)` in cases when raw data are provided), `'sampleSize'` (single number), `'pairwiseN'` (matrix of pairwise N; could be used if correlation matrix is based on pairwise N), `'empcov'` (correlation matrix), `'moderator'` (vector of numbers; could be continuous or categorical), `'startValues'` (vector of start values), `'rawData'` (information about file name and structure of raw data), `'empMeans'` (means for variables; usually 0), and `'empVars'` (variances for variables; usually 1). Weakly predefined objects are `'studyNumber'` (intended as a special number used for the outputs of subsequently fitted CoTiMA models), `'source'` (intended as vector of authors' names and publication year), `'ageM'` (intended as value indicating the mean age of participants in a primary study), `'malePercent'` (intended as value indicating the percentage of male participants in a primary study), `'occupation'` (intended as vector of character strings representing the occupations of participants in a primary study), `'country'` (intended as single character string representing the country in which a primary study was conducted), `'alphas'` (intended as vector of Cronbach's alphas of the variables of a primary study; not yet functional), and `'targetVariables'` (intended as vector of character strings representing information about the variables used).'

<code>addElement</code>	User-added objects that are handled as the weakly predefined objects. The major purpose is to collect information a researcher regards as important.
<code>digits</code>	Rounding used for summary function
<code>moderatorLabels</code>	character vector of names
<code>moderatorValues</code>	list of character vectors
<code>summary</code>	if TRUE (default) creates summary table and xlsx sheets. Could be set to FALSE in case of errors.

Value

List of primary studies and parameters for the following CoTiMA (plus `StudyInformation` which could be saved to Excel)

Note

The following example shows information a researcher has about three studies, which have the numbers '2', '4' and '17'. All information about these studies are stored in objects ending with '2', '4', and '17', respectively. In most instances, one relevant piece of information is the empirical correlation (or covariance) matrix reported in this study, which is stored in the objects `'empcov2'`, `'empcov4'`, and `'empcov17'`. Note that full and symmetric matrices are required for `ctmaPrep`. Usually, sample sizes (`'sampleSize2'`, `'sampleSize4'`, & `'sampleSize17'`) and time lags (`'delta_t2'`, `'delta_t4'`, & `'delta_t17'`), are required, too.

Examples

```
# First Study
empcov2 <- matrix(c(1.00, 0.45, 0.57, 0.18,
                  0.45, 1.00, 0.31, 0.66,
```

```

                                0.57, 0.31, 1.00, 0.40,
                                0.18, 0.66, 0.40, 1.00), nrow=4, ncol=4)
delta_t2 <- 12
sampleSize2 <- 148
moderator2 <- c(1, 0.72)
source2 <- c("Houkes, I,", "Janssen, P, P, M,", "de Jonge, J",
            "& Bakker, A, B", "Study1", "2003")
addedByResearcher2 <- "something you want to add"

# Second Study
empcov3 <- matrix(c(1.00, 0.43, 0.71, 0.37,
                   0.43, 1.00, 0.34, 0.69,
                   0.71, 0.34, 1.00, 0.50,
                   0.37, 0.69, 0.50, 1.00), nrow=4, ncol=4)
delta_t3 <- 12
sampleSize3 <- 88
moderator3 <- c(1, 0.72)
source3 <- c("Houkes, I,", "Janssen, P, P, M,", "de Jonge, J",
            "& Bakker, A, B", "Study2", "2003")
addedByResearcher3 <- ""

# Third Study
empcov313 <- matrix(c(1.00, 0.38, 0.54, 0.34, 0.60, 0.28,
                    0.38, 1.00, 0.34, 0.68, 0.28, 0.68,
                    0.54, 0.34, 1.00, 0.47, 0.66, 0.39,
                    0.34, 0.68, 0.47, 1.00, 0.38, 0.72,
                    0.60, 0.28, 0.66, 0.38, 1.00, 0.38,
                    0.28, 0.68, 0.39, 0.72, 0.38, 1.00), nrow=6, ncol=6)
delta_t313 <- c(1.5, 1.5)
sampleSize313 <- 335
moderator313 <- c(0.8, 2.47)
source313 <- c("Demerouti", "Bakker", "& Bulters", "2004")
addedByResearcher313 <- "check correlation matrix"

# Add Labels and Values for Moderators (just for optional excel tables)
moderatorLabels <- c("Control", "Social Support")
moderatorValues <- list("continuous", c("1 = very low", "2 = low",
    "3 = medium", "4 = high", "5 = very high"))

CoTiMAstudyList_3 <- ctmaPrep(selectedStudies = c(2, 3, 313),
    excludedElements = "ageM",
    addElements = "addedByResearcher",
    moderatorLabels=moderatorLabels,
    moderatorValues=moderatorValues)

```

Description

Compute publication and citation scores for studies based on the (team of) authors' publication scores .

Usage

```
ctmaPub(
  getPubObj = NULL,
  primaryStudyList = NULL,
  yearsToExclude = 0,
  recency = 5,
  targetYear = NULL,
  indFUN = "sum",
  colFUN = "mean",
  addAsMod = FALSE
)
```

Arguments

getPubObj	publication information compiled with ctmaGetPub
primaryStudyList	vector with numbers of studies (e.g., c(1,3); requires source1 and source3 to be available)
yearsToExclude	years to exclude from publications
recency	years before targetYear that are considered for recency analysis
targetYear	year (default = last year) after which publications are ignored
indFUN	function (default = sum) how publications of each author within a collective (team) are summarized
colFUN	function (default = mean) how publications all authors of collective (team) are summarized
addAsMod	currently disabled. Add to existing moderator objects (or create them) in primaryStudyList, which is part of the returned object

Value

returns NEPP (= the `*number*` of studies published by the authors of the primary studies supplied UNTIL the year when the primary study was published), NEPPRecency (like NEPP, but limited to the number of years before the publication as specified with the recency argument), "Meaning of NEPP" and "Meaning of NEPPRecency" which explain what `*number*` exactly means (e.g., could be the mean of the sum of each author's publication, or the sum of the maximum publications per year of the authors), and "primaryStudyList(full)", which just returns the primaryStudyList supplied).

Examples

```
pubResults_6 <- ctmaPub(getPubObj=pubList_8,
```



```
summary(pubResults_6) primaryStudyList=CoTiMAstudyList_6)
```

ctmaSaveFile

ctmaSaveFile

Description

Internal function to save files

Usage

```
ctmaSaveFile(  
  activateRPB,  
  activeDirectory = activeDirectory,  
  SaveObject,  
  FileName,  
  Directory,  
  silentOverwrite = FALSE  
)
```

Arguments

activateRPB	set TRUE to receive push messages with 'CoTiMA' notifications on your phone
activeDirectory	directory name
SaveObject	object to save
FileName	filename
Directory	directory to save file in
silentOverwrite	override old files without asking

Value

No return value. Just saves files

ctmaStanResample	<i>ctmaStanResample</i>
------------------	-------------------------

Description

re-sample from a fitted stanct model to achieve desired number of finishsamples (could be useful to prevent exhausted memory)

Usage

```
ctmaStanResample(ctmaFittedModel = NULL, nsamples = 25, overallSamples = 500)
```

Arguments

ctmaFittedModel	a 'CoTiMA' fit object, usually with few 'finishsamples' to prevent memory exhaustion
nsamples	sample size per run
overallSamples	overall samples size to be achieved

Value

returns a CoTiMA fit object with an increased number of finish samples

ctmaSV	<i>ctmaSV</i>
--------	---------------

Description

derives start values by average discrete time SEM effects, converting them to continuous time, and inversely apply transformations used by 'ctsem'

Usage

```
ctmaSV(
  ctmaInitFit = NULL,
  activeDirectory = NULL,
  primaryStudies = NULL,
  coresToUse = 1,
  replaceSV = TRUE
)
```

Arguments

ctmaInitFit object to which all single 'ctsem' fits of primary studies has been assigned to (i.e., what has been returned by `ctmaInit`)
activeDirectory defines another active directory than the one used in `ctmaInit`
primaryStudies if `ctmaInitFit` does not contain the `primaryStudies` object created with `ctmaPrep` it could be added
coresToUse if negative, the value is subtracted from available cores, else value = cores to use
replaceSV if TRUE replaces `startValues` in `primaryStudies`, else it saves them as list element `inits`

Value

returns a modified list of primary studies with starting values added or replaced

Examples

```
## Not run:
newPrimaryStudyList <- ctmaSV(ctmaInitFit=CoTiMAInitFit_6)

## End(Not run)
```

delta_t128

delta_t128 example vector

Description

delta_t128 example vector

Usage

delta_t128

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

delta_t18

delta_t18 example vector

Description

delta_t18 example vector

Usage

delta_t18

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

delta_t2

delta_t2 example vector

Description

delta_t2 example vector

Usage

delta_t2

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

delta_t3 *delta_t3 example vector*

Description

delta_t3 example vector

Usage

delta_t3

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

delta_t313 *delta_t313 example vector*

Description

delta_t313 example vector

Usage

delta_t313

Format

An object of class numeric of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

delta_t32

delta_t32 example vector

Description

delta_t32 example vector

Usage

delta_t32

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

demands128

demands128 example vector

Description

demands128 example vector

Usage

demands128

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

demands18

demands18 example vector

Description

demands18 example vector

Usage

demands18

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

demands2

demands2 example vector

Description

demands2 example vector

Usage

demands2

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

demands3

demands3 example vector

Description

demands3 example vector

Usage

demands3

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

demands313

demands313 example vector

Description

demands313 example vector

Usage

demands313

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

`demands32`*demands32 example vector*

Description

demands32 example vector

Usage

demands32

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

`dl_link`*dl_link example path*

Description

dl_link example path

Usage

dl_link

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

empcov128

empcov128 example matrix

Description

empcov128 example matrix

Usage

empcov128

Format

An object of class `list` of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

empcov18

empcov18 example matrix

Description

empcov18 example matrix

Usage

empcov18

Format

An object of class `matrix` (inherits from `array`) with 4 rows and 4 columns.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

empcov2

empcov2 example matrix

Description

empcov2 example matrix

Usage

empcov2

Format

An object of class `matrix` (inherits from `array`) with 4 rows and 4 columns.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

empcov3

empcov3 example matrix

Description

empcov3 example matrix

Usage

empcov3

Format

An object of class `matrix` (inherits from `array`) with 4 rows and 4 columns.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

empcov313

empcov313 example matrix

Description

empcov313 example matrix

Usage

empcov313

Format

An object of class `matrix` (inherits from `array`) with 6 rows and 6 columns.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

empcov32

empcov32 example matrix

Description

empcov32 example matrix

Usage

empcov32

Format

An object of class `character` of length 4.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

malePercent128 *malePercent128 example vector*

Description

malePercent128 example vector

Usage

malePercent128

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

malePercent18 *malePercent18 example vector*

Description

malePercent18 example vector

Usage

malePercent18

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

malePercent2

malePercent2 example vector

Description

malePercent2 example vector

Usage

malePercent2

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

malePercent3

malePercent3 example vector

Description

malePercent3 example vector

Usage

malePercent3

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

malePercent313 *malePercent313 example vector*

Description

malePercent313 example vector

Usage

malePercent313

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

malePercent32 *malePercent32 example vector*

Description

malePercent32 example vector

Usage

malePercent32

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

moderator128

moderator128 example vector

Description

moderator128 example vector

Usage

moderator128

Format

An object of class `numeric` of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

moderator18

moderator18 example vector

Description

moderator18 example vector

Usage

moderator18

Format

An object of class `numeric` of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

moderator2

moderator2 example vector

Description

moderator2 example vector

Usage

moderator2

Format

An object of class `numeric` of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

moderator3

moderator3 example vector

Description

moderator3 example vector

Usage

moderator3

Format

An object of class `numeric` of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

moderator313

moderator313 example vector

Description

moderator313 example vector

Usage

moderator313

Format

An object of class `numeric` of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

moderator32

moderator32 example vector

Description

moderator32 example vector

Usage

moderator32

Format

An object of class `numeric` of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

`moderatorLabels` *moderatorLabels example vector*

Description

`moderatorLabels` example vector

Usage

`moderatorLabels`

Format

An object of class character of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

`moderatorValues` *moderatorValues example vector*

Description

`moderatorValues` example vector

Usage

`moderatorValues`

Format

An object of class list of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

occupation128 *occupation128 example vector*

Description

occupation128 example vector

Usage

occupation128

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

occupation18 *occupation18 example vector*

Description

occupation18 example vector

Usage

occupation18

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

occupation2	<i>occupation2 example vector</i>
-------------	-----------------------------------

Description

occupation2 example vector

Usage

occupation2

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

occupation3	<i>occupation3 example vector</i>
-------------	-----------------------------------

Description

occupation3 example vector

Usage

occupation3

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

occupation313 *occupation313 example vector*

Description

occupation313 example vector

Usage

occupation313

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

occupation32 *occupation32 example vector*

Description

occupation32 example vector

Usage

occupation32

Format

An object of class character of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

optimFit313

optimFit313 example vector

Description

optimFit313 example vector

Usage

optimFit313

Format

An object of class CoTiMAFit of length 3.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

pairwiseN128

pairwiseN128 example vector

Description

pairwiseN128 example vector

Usage

pairwiseN128

Format

An object of class list of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

plot.CoTiMAFit	<i>plot.CoTiMAFit</i>
----------------	-----------------------

Description

call `ctmaPlot` if a `CoTiMAFit` object is supplied to `plot()`

Usage

```
## S3 method for class 'CoTiMAFit'  
plot(x, ...)
```

Arguments

x	list
...	further arguments to be passed through to <code>summary()</code>

Value

returns a call to `'ctmaPlot'`, which is used to plot `CoTiMA` fit objects

pubList_8	<i>pubList_8 example list</i>
-----------	-------------------------------

Description

pubList_8 example list

Usage

```
pubList_8
```

Format

An object of class `CoTiMAFit` of length 10.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

pubResults_6

pubResults_6 example list

Description

pubResults_6 example list

Usage

pubResults_6

Format

An object of class CoTiMAFit of length 6.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

rawData128

rawData128 example list

Description

rawData128 example list

Usage

rawData128

Format

An object of class list of length 7.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

recodeVariables128 *recodeVariables128 example vector*

Description

recodeVariables128 example vector

Usage

recodeVariables128

Format

An object of class character of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

results128 *results128 example list*

Description

results128 example list

Usage

results128

Format

An object of class list of length 2.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

sampleSize128 *sampleSize128 example vector*

Description

sampleSize128 example vector

Usage

sampleSize128

Format

An object of class NULL of length 0.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

sampleSize18 *sampleSize18 example vector*

Description

sampleSize18 example vector

Usage

sampleSize18

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

sampleSize2

sampleSize2 example vector

Description

sampleSize2 example vector

Usage

sampleSize2

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

sampleSize3

sampleSize3 example vector

Description

sampleSize3 example vector

Usage

sampleSize3

Format

An object of class `numeric` of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

sampleSize313 *sampleSize313 example vector*

Description

sampleSize313 example vector

Usage

sampleSize313

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

sampleSize32 *sampleSize32 example vector*

Description

sampleSize32 example vector

Usage

sampleSize32

Format

An object of class numeric of length 1.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

source128

source128 example vector

Description

source128 example vector

Usage

source128

Format

An object of class character of length 4.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

source2

source2 example vector

Description

source2 example vector

Usage

source2

Format

An object of class character of length 6.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

source3

source3 example vector

Description

source3 example vector

Usage

source3

Format

An object of class character of length 6.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

source313

source313 example vector

Description

source313 example vector

Usage

source313

Format

An object of class character of length 4.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

summary.CoTiMAFit *summary.CoTiMAFit*

Description

defines summary for 'CoTiMA' fit objects

Usage

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

Arguments

object one CoTiMAFit object or more as ctmaFitList(object1, object2, ...)
... further arguments to be passed through to summary()

Value

returns a printed summary of a 'CoTiMA' fit object

targetVariables128 *targetVariables128 example vector*

Description

targetVariables128 example vector

Usage

```
targetVariables128
```

Format

An object of class character of length 7.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

targetVariables2 *targetVariables2 example vector*

Description

targetVariables2 example vector

Usage

targetVariables2

Format

An object of class character of length 4.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

targetVariables3 *targetVariables3 example vector*

Description

targetVariables3 example vector

Usage

targetVariables3

Format

An object of class character of length 4.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

targetVariables313 *targetVariables313 example vector*

Description

targetVariables313 example vector

Usage

targetVariables313

Format

An object of class character of length 6.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

variableNames128 *variableNames128 example vector*

Description

variableNames128 example vector

Usage

variableNames128

Format

An object of class character of length 9.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

Index

* datasets

CoTiMAStanctArgs, [25](#)

* data

A128, [5](#)

A313, [5](#)

addedByResearcher2, [6](#)

addedByResearcher3, [6](#)

addedByResearcher313, [7](#)

ageM128, [7](#)

ageM18, [8](#)

ageM2, [8](#)

ageM3, [9](#)

ageM313, [9](#)

ageM32, [10](#)

ageSD128, [10](#)

ageSD18, [11](#)

ageSD2, [11](#)

ageSD3, [12](#)

ageSD313, [12](#)

ageSD32, [13](#)

alphas128, [13](#)

alphas313, [14](#)

burnout128, [14](#)

burnout18, [15](#)

burnout2, [15](#)

burnout3, [16](#)

burnout313, [16](#)

burnout32, [17](#)

combineVariables128, [17](#)

combineVariablesNames128, [18](#)

CoTiMABiG_D_B0, [18](#)

CoTiMAFullFit_3, [19](#)

CoTiMAFullFit_6, [19](#)

CoTiMAFullInv23Fit_6, [20](#)

CoTiMAFullInvEq23Fit_6, [20](#)

CoTiMAInitFit_3, [21](#)

CoTiMAInitFit_6, [21](#)

CoTiMAInitFit_6_NUTS, [22](#)

CoTiMAInitFit_D_B0, [22](#)

CoTiMAMod1onFullFit_6, [23](#)

CoTiMAMod2on23Fit_6, [23](#)

CoTiMAPart134Inv3Fit_6, [24](#)

CoTiMAPower_D_B0, [24](#)

CoTiMAstudyList_3, [25](#)

CoTiMAstudyList_6, [26](#)

country128, [26](#)

country18, [27](#)

country2, [27](#)

country3, [28](#)

country313, [28](#)

country32, [29](#)

delta_t128, [59](#)

delta_t18, [60](#)

delta_t2, [60](#)

delta_t3, [61](#)

delta_t313, [61](#)

delta_t32, [62](#)

demands128, [62](#)

demands18, [63](#)

demands2, [63](#)

demands3, [64](#)

demands313, [64](#)

demands32, [65](#)

d1_link, [65](#)

empcov128, [66](#)

empcov18, [66](#)

empcov2, [67](#)

empcov3, [67](#)

empcov313, [68](#)

empcov32, [68](#)

malePercent128, [69](#)

malePercent18, [69](#)

malePercent2, [70](#)

malePercent3, [70](#)

malePercent313, [71](#)

malePercent32, [71](#)

moderator128, [72](#)

moderator18, [72](#)

- moderator2, [73](#)
 - moderator3, [73](#)
 - moderator313, [74](#)
 - moderator32, [74](#)
 - moderatorLabels, [75](#)
 - moderatorValues, [75](#)
 - occupation128, [76](#)
 - occupation18, [76](#)
 - occupation2, [77](#)
 - occupation3, [77](#)
 - occupation313, [78](#)
 - occupation32, [78](#)
 - optimFit313, [79](#)
 - pairwiseN128, [79](#)
 - pubList_8, [80](#)
 - pubResults_6, [81](#)
 - rawData128, [81](#)
 - recodeVariables128, [82](#)
 - results128, [82](#)
 - sampleSize128, [83](#)
 - sampleSize18, [83](#)
 - sampleSize2, [84](#)
 - sampleSize3, [84](#)
 - sampleSize313, [85](#)
 - sampleSize32, [85](#)
 - source128, [86](#)
 - source2, [86](#)
 - source3, [87](#)
 - source313, [87](#)
 - targetVariables128, [88](#)
 - targetVariables2, [89](#)
 - targetVariables3, [89](#)
 - targetVariables313, [90](#)
 - variableNames128, [90](#)
-
- A128, [5](#)
 - A313, [5](#)
 - addedByResearcher2, [6](#)
 - addedByResearcher3, [6](#)
 - addedByResearcher313, [7](#)
 - ageM128, [7](#)
 - ageM18, [8](#)
 - ageM2, [8](#)
 - ageM3, [9](#)
 - ageM313, [9](#)
 - ageM32, [10](#)
 - ageSD128, [10](#)
 - ageSD18, [11](#)
 - ageSD2, [11](#)
 - ageSD3, [12](#)
 - ageSD313, [12](#)
 - ageSD32, [13](#)
 - alphas128, [13](#)
 - alphas313, [14](#)
 - burnout128, [14](#)
 - burnout18, [15](#)
 - burnout2, [15](#)
 - burnout3, [16](#)
 - burnout313, [16](#)
 - burnout32, [17](#)
 - combineVariables128, [17](#)
 - combineVariablesNames128, [18](#)
 - CoTiMABiG_D_BO, [18](#)
 - CoTiMAFullFit_3, [19](#)
 - CoTiMAFullFit_6, [19](#)
 - CoTiMAFullInv23Fit_6, [20](#)
 - CoTiMAFullInvEq23Fit_6, [20](#)
 - CoTiMAInitFit_3, [21](#)
 - CoTiMAInitFit_6, [21](#)
 - CoTiMAInitFit_6_NUTS, [22](#)
 - CoTiMAInitFit_D_BO, [22](#)
 - CoTiMAMod1onFullFit_6, [23](#)
 - CoTiMAMod2on23Fit_6, [23](#)
 - CoTiMAPart134Inv3Fit_6, [24](#)
 - CoTiMAPower_D_BO, [24](#)
 - CoTiMAStantArgs, [25](#)
 - CoTiMAstudyList_3, [25](#)
 - CoTiMAstudyList_6, [26](#)
 - country128, [26](#)
 - country18, [27](#)
 - country2, [27](#)
 - country3, [28](#)
 - country313, [28](#)
 - country32, [29](#)
 - ctmaAllInvFit, [29](#)
 - ctmaBiG, [31](#)
 - ctmaBiGOMX, [32](#)
 - ctmaCombPRaw, [33](#)
 - ctmaCompFit, [33](#)
 - ctmaCorRel, [34](#)
 - ctmaEmpCov, [35](#)
 - ctmaEqual, [33](#), [36](#)
 - ctmaFit, [33](#), [36](#), [37](#), [38](#), [42](#)
 - ctmaFitList, [41](#)
 - ctmaFitToPrep, [42](#), [47](#)
 - ctmaGetPub, [42](#), [56](#)

- ctmaInit, [31](#), [39](#), [42](#), [43](#), [47](#), [50](#), [59](#)
- ctmaLabels, [46](#)
- ctmaOptimizeInit, [47](#)
- ctmaPlot, [48](#), [80](#)
- ctmaPower, [50](#)
- ctmaPRaw, [52](#)
- ctmaPrep, [39](#), [42–44](#), [47](#), [53](#), [59](#)
- ctmaPub, [55](#)
- ctmaSaveFile, [57](#)
- ctmaStanResample, [58](#)
- ctmaSV, [58](#)

- delta_t128, [59](#)
- delta_t18, [60](#)
- delta_t2, [60](#)
- delta_t3, [61](#)
- delta_t313, [61](#)
- delta_t32, [62](#)
- demands128, [62](#)
- demands18, [63](#)
- demands2, [63](#)
- demands3, [64](#)
- demands313, [64](#)
- demands32, [65](#)
- dl_link, [65](#)

- empcov128, [66](#)
- empcov18, [66](#)
- empcov2, [67](#)
- empcov3, [67](#)
- empcov313, [68](#)
- empcov32, [68](#)

- malePercent128, [69](#)
- malePercent18, [69](#)
- malePercent2, [70](#)
- malePercent3, [70](#)
- malePercent313, [71](#)
- malePercent32, [71](#)
- moderator128, [72](#)
- moderator18, [72](#)
- moderator2, [73](#)
- moderator3, [73](#)
- moderator313, [74](#)
- moderator32, [74](#)
- moderatorLabels, [75](#)
- moderatorValues, [75](#)

- occupation128, [76](#)
- occupation18, [76](#)
- occupation2, [77](#)
- occupation3, [77](#)
- occupation313, [78](#)
- occupation32, [78](#)
- optimFit313, [79](#)

- pairwiseN128, [79](#)
- plot.CoTiMAFit, [80](#)
- pubList_8, [80](#)
- pubResults_6, [81](#)

- rawData128, [81](#)
- recodeVariables128, [82](#)
- results128, [82](#)

- sampleSize128, [83](#)
- sampleSize18, [83](#)
- sampleSize2, [84](#)
- sampleSize3, [84](#)
- sampleSize313, [85](#)
- sampleSize32, [85](#)
- source128, [86](#)
- source2, [86](#)
- source3, [87](#)
- source313, [87](#)
- summary.CoTiMAFit, [88](#)

- targetVariables128, [88](#)
- targetVariables2, [89](#)
- targetVariables3, [89](#)
- targetVariables313, [90](#)

- variableNames128, [90](#)