Package ‘CoTiMA’

Type Package
Title Continuous Time Meta-Analysis (‘CoTiMA’)
Version 0.6.2
Date 2022-11-08
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 ‘Co-
TiMA’, 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 Dorm-
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)
R topics documented:

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

Suggests  R.rsp
VignetteBuilder  R.rsp
RoxygenNote  7.2.1

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  2022-11-08 10:40:10 UTC

R topics documented:

A128 ................................................................. 5
A313 ................................................................. 6
addedByResearcher2 .............................................. 6
addedByResearcher3 .............................................. 7
addedByResearcher313 .......................................... 7
ageM128 ........................................................... 8
ageM18 ............................................................. 8
ageM2 ............................................................. 9
ageM201 .......................................................... 9
ageM3 ........................................................... 10
ageM313 .......................................................... 10
ageM32 ........................................................... 11
ageSD128 ........................................................ 11
ageSD18 .......................................................... 12
ageSD2 ........................................................... 12
ageSD201 .......................................................... 13
ageSD3 ........................................................... 13
ageSD313 ........................................................ 14
ageSD32 ........................................................... 14
alphas128 ........................................................ 15
alphas313 ........................................................ 15
burnout128 ....................................................... 16
burnout18 ........................................................ 16
burnout2 .......................................................... 17
burnout201 ........................................................ 17
burnout3 .......................................................... 18
burnout313 ....................................................... 18
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>burnout32</td>
<td>19</td>
</tr>
<tr>
<td>combineVariables128</td>
<td>19</td>
</tr>
<tr>
<td>combineVariablesNames128</td>
<td>20</td>
</tr>
<tr>
<td>CoTiMABiG_D_BO</td>
<td>20</td>
</tr>
<tr>
<td>CoTiMAFullFit_3</td>
<td>21</td>
</tr>
<tr>
<td>CoTiMAFullFit_6</td>
<td>21</td>
</tr>
<tr>
<td>CoTiMAFullFit_6_new</td>
<td>22</td>
</tr>
<tr>
<td>CoTiMAFullInv23Fit_6</td>
<td>22</td>
</tr>
<tr>
<td>CoTiMAFullInvEq23Fit_6</td>
<td>23</td>
</tr>
<tr>
<td>CoTiMAInitFit_3</td>
<td>23</td>
</tr>
<tr>
<td>CoTiMAInitFit_6</td>
<td>24</td>
</tr>
<tr>
<td>CoTiMAInitFit_6_new</td>
<td>24</td>
</tr>
<tr>
<td>CoTiMAInitFit_6_NUTS</td>
<td>25</td>
</tr>
<tr>
<td>CoTiMAInitFit_D_BO</td>
<td>25</td>
</tr>
<tr>
<td>CoTiMAInitFullFit_6</td>
<td>26</td>
</tr>
<tr>
<td>CoTiMAInitFullFit_6_cats12</td>
<td>26</td>
</tr>
<tr>
<td>CoTiMAInitFullFit_6</td>
<td>27</td>
</tr>
<tr>
<td>CoTiMAInitOptimFit313</td>
<td>27</td>
</tr>
<tr>
<td>CoTiMAPart134Inv3Fit_6</td>
<td>28</td>
</tr>
<tr>
<td>CoTiMAPower_D_BO</td>
<td>28</td>
</tr>
<tr>
<td>CoTiMAnInitStancetArgs</td>
<td>29</td>
</tr>
<tr>
<td>CoTiMAnStudyList_3</td>
<td>29</td>
</tr>
<tr>
<td>CoTiMAnStudyList_6</td>
<td>30</td>
</tr>
<tr>
<td>CoTiMAnStudyList_6_new</td>
<td>30</td>
</tr>
<tr>
<td>country128</td>
<td>31</td>
</tr>
<tr>
<td>country18</td>
<td>31</td>
</tr>
<tr>
<td>country2</td>
<td>32</td>
</tr>
<tr>
<td>country201</td>
<td>32</td>
</tr>
<tr>
<td>country3</td>
<td>33</td>
</tr>
<tr>
<td>country313</td>
<td>33</td>
</tr>
<tr>
<td>country32</td>
<td>34</td>
</tr>
<tr>
<td>ctmaAllInvFit</td>
<td>34</td>
</tr>
<tr>
<td>ctmaBiG</td>
<td>36</td>
</tr>
<tr>
<td>ctmaBiGOMX</td>
<td>37</td>
</tr>
<tr>
<td>ctmaCombPRaw</td>
<td>38</td>
</tr>
<tr>
<td>ctmaCompFit</td>
<td>39</td>
</tr>
<tr>
<td>ctmaCorRel</td>
<td>39</td>
</tr>
<tr>
<td>ctmaEmpCov</td>
<td>40</td>
</tr>
<tr>
<td>ctmaEqual</td>
<td>42</td>
</tr>
<tr>
<td>ctmaFit</td>
<td>43</td>
</tr>
<tr>
<td>ctmaFitList</td>
<td>47</td>
</tr>
<tr>
<td>ctmaFitToPrep</td>
<td>47</td>
</tr>
<tr>
<td>ctmaGetPub</td>
<td>48</td>
</tr>
<tr>
<td>ctmaInit</td>
<td>49</td>
</tr>
<tr>
<td>ctmaLabels</td>
<td>52</td>
</tr>
<tr>
<td>ctmaOptimizeFit</td>
<td>53</td>
</tr>
<tr>
<td>ctmaOptimizeInit</td>
<td>55</td>
</tr>
<tr>
<td>ctmaPlot</td>
<td>57</td>
</tr>
</tbody>
</table>
### R topics documented:

- `ctmaPower` ................................................. 59
- `ctmaPRaw` ................................................. 61
- `ctmaPrep` .................................................. 62
- `ctmaPub` .................................................... 65
- `ctmaSaveFile` .............................................. 66
- `ctmaScaleInits` ............................................ 67
- `ctmaShapeRawData` ....................................... 67
- `ctmaStanResample` ........................................ 71
- `ctmaSV` ...................................................... 71
- `delta_t128` ............................................... 72
- `delta_t118` ............................................... 72
- `delta_t12` .................................................. 73
- `delta_t1201` .............................................. 74
- `delta_t13` .................................................. 74
- `delta_t1313` .............................................. 75
- `delta_t132` ............................................... 75
- `demands128` .............................................. 76
- `demands18` ............................................... 76
- `demands2` .................................................. 77
- `demands201` .............................................. 77
- `demands3` .................................................. 78
- `demands313` .............................................. 78
- `demands32` ............................................... 79
- `dl_link` .................................................... 79
- `empcov128` .............................................. 80
- `empcov18` ................................................ 80
- `empcov2` .................................................. 81
- `empcov201` .............................................. 81
- `empcov3` .................................................. 82
- `empcov313` .............................................. 82
- `empcov32` ................................................ 82
- `malePercent128` ......................................... 83
- `malePercent18` .......................................... 83
- `malePercent2` ............................................ 84
- `malePercent201` ......................................... 84
- `malePercent3` ............................................ 85
- `malePercent313` ......................................... 85
- `malePercent32` .......................................... 86
- `moderator128` .......................................... 86
- `moderator18` ............................................ 87
- `moderator2` .............................................. 87
- `moderator201` .......................................... 88
- `moderator3` .............................................. 88
- `moderator313` .......................................... 89
- `moderator32` ............................................ 90
- `moderatorLabels` ........................................ 90
- `moderatorValues` ........................................ 91
- `occupation128` ......................................... 91
A128 example matrix

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

A313 example matrix

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 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 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

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

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>
**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>

---

**Description**

ageM201 example vector

**Usage**

ageM201

**Format**

An object of class numeric of length 1.

**Author(s)**

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
ageM3

**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

**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>
**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>

---

**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>
**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>

---

**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>
**ageSD201**  
*ageSD201 example vector*

**Description**  
ageSD201 example vector

**Usage**  
ageSD201

**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 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 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 | 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 | 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**

**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>

---

**burnout201**

**burnout201 example vector**

**Description**

burnout201 example vector

**Usage**

burnout201

**Format**

An object of class character of length 1.

**Author(s)**

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
burnout3  

**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  

**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 10.

**Author(s)**

C. Guthier, C. Dormann & J. Cortina <CoTiMA@uni-mainz.org>
CoTiMAFullFit_3

Description

cmFit-object with a ‘full’ CoTiMA of 3 studies

Usage

CoTiMAFullFit_3

Format

An object of class CoTiMAFit of length 16.

Author(s)

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

CoTiMAFullFit_6

cmFit-object with a ‘full’ CoTiMA of 6 studies

Usage

CoTiMAFullFit_6

Format

An object of class CoTiMAFit of length 12.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
CoTiMAFullInv23Fit_6

Description
cmaFit-object with a 'full' CoTiMA of 6 studies

Usage
CoTiMAFullFit_6_new

Format
An object of class CoTiMAFit of length 13.

Author(s)
C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAFullInv23Fit_6

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 14.

Author(s)
C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
**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 12.

**Author(s)**

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

---

**Description**

ctmaInit-object with of 3 primary studies

**Usage**

CoTiMAInitFit_3

**Format**

An object of class CoTiMAFit of length 15.

**Author(s)**

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
CoTiMAInitFit_6

cmaInit-object with 6 primary studies

Description
ctmaInit-object with 6 primary studies

Usage
CoTiMAInitFit_6

Format
An object of class CoTiMAFit of length 17.

Author(s)
C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

CoTiMAInitFit_6_new

cmaInit-object with 6 primary studies

Description
ctmaInit-object with 6 primary studies

Usage
CoTiMAInitFit_6_new

Format
An object of class CoTiMAFit of length 15.

Author(s)
C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
CoTiMAInitFit_6_NUTS

c DMAInitFit_6_NUTS  ctmaInit-object with a 'full' CoTiMA of 6 studies using NUTS sampler

Description

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

Usage

CoTiMAInitFit_6_NUTS

Format

An object of class CoTiMAFit of length 16.

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

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

Usage

CoTiMAInitFit_D_BO

Format

An object of class CoTiMAFit of length 12.

Author(s)

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

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

Usage

CoTiMAMod1onFullFit_6

Format

An object of class CoTiMAFit of length 15.

Author(s)

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

---

Description

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

Usage

CoTiMAMod1onFullFit_6_cats12

Format

An object of class CoTiMAFit of length 13.

Author(s)

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
Description
ctmaFit-object with a continuous moderator of 2 cross effects

Usage
CoTiMAMod2on23Fit_6

Format
An object of class CoTiMAfit of length 15.

Author(s)
C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>

Description
CoTiMAoptimFit313 example vector

Usage
CoTiMAoptimFit313

Format
An object of class CoTiMAfit of length 4.

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**

cAGMA-fit-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 16.

**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 10.

**Author(s)**

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

This are preset arguments

Usage

CoTiMAStanctArgs

Format

An object of class list of length 36.

CoTiMASudyList_3  ctmaPrep-object created with 3 primary studies

Description

ctmaPrep-object created with 3 primary studies

Usage

CoTiMASudyList_3

Format

An object of class CoTiMAfit of length 28.

Author(s)

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

cmaPrep-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>

---

**country201**

**country201 example vector**

**Description**

country201 example vector

**Usage**

country201

**Format**

An object of class character of length 1.

**Author(s)**

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
**country3**

### 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**

### 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>
ctmaAllInvFit

country32  country32 example vector

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>

cmmaAllInvFit  cmmaAllInvFit

Description

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

Usage

cmmaAllInvFit(
  ctmaInitFit = NULL,
  activeDirectory = NULL,
  activateRBP = 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 = FALSE,
  T0means = 0,
  manifestMeans = 0,
  CoTiMAStanctArgs = NULL,
  lambda = NULL,
  manifestVars = NULL
)

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).
ingVarying, Allows ct intercepts to vary at the individual level (random effects model, accounts for unobserved heterogeneity)
scaleTime, scaleTime
optimize, optimize
nopriors, nopriors
finishsamples, finishsamples
iter, iter
chains, chains
verbose, verbose
loadAllInvFit, loadAllInvFit
saveAllInvFit, saveAllInvFit
silentOverwrite, silentOverwrite
customPar, logical. If set TRUE (default) leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (helps since ctsem > 3.4)
T0means, Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
manifestMeans, Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
CoTiMAStanctArgs, parameters that can be set to improve model fitting of the ctStanFit Function
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 lower triangular matrix with nrow=n.manifest & ncol=n.manifest.

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

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
```r
cdmaBiG(
  ctmaInitFit = NULL,
  activeDirectory = NULL,
  PETPEESEalpha = 0.1,
  activateRPB = FALSE,
  digits = 4,
  zcurve = FALSE,
  undoTimeScaling = TRUE
)
```

Arguments
- `ctmaInitFit`: fit object created with `ctmaInit` 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 (cf. Stanley, 2017, p. 582, below Eq. 2; default p = .10)
- `activateRPB`: if TRUE, messages (warning, finished) could be send to smart phone (default = FALSE)
- `digits`: rounding (default = 4)
- `zcurve`: performs z-curve analysis. Could fail if too few studies (e.g. around 10) are supplied. default=FALSE
- `undoTimeScaling`: if TRUE, the original time scale is used (timeScale argument possibly used in `ctmaInit` is undone)
ctmaBiGOMX

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

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

Description

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

Usage

```r
c dmaBiGOMX(
    ctmaInitFit = NULL,
    activeDirectory = NULL,
    PETPEESEalpha = 0.1,
```
ctmaCombPRaw

Arguments

cdmaInitFit
fit object created with cdmaInti containing the fitted ctsem model of each primary study
activeDirectory
the directory where to save results (if not specified, it is taken from cdmaInitFit)
PETPEESEalpha
# probability level (condition) below which to switch from PET to PEESE (Stanley, 2017, SPSS, 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.

Description

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

Usage

cdmaCombPRaw(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
Description

Perform 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

```r
minus2llDiffTest <- ctmaCompFit(CoTiMAFullInv23Fit_6,
                                 CoTiMAFullInvEq23Fit_6)
summary(minus2llDiffTest)
```

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

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

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

```r
cdmaEmpCov(
  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
ctmaEmpCov

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
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

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

Arguments

cdmaInvariantFit  object to which a CoTiMA fit has been assigned to (i.e., what has been returned by ctmaFit). In most cases probably a model in which (only) two effects were specified with invariantDrift.

activeDirectory  defines another active directory than the one used in ctmaInvariantFit

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

digits  Number of digits used for rounding (in outputs)

coresToUse  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)
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

cdmaFit(
  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 = TRUE,
  scaleMod = NULL,
  transfMod = NULL,
  scaleClus = NULL,
  scaleTime = NULL,
  optimize = TRUE,
  nopriors = TRUE,
  finishesamples = NULL,
  iter = NULL,
  chains = NULL,
  verbose = NULL,
  allInvModel = FALSE,
  customPar = FALSE,
  inits = NULL,
  modsToCompare = NULL,
  catsToCompare = NULL,
driftsToCompare = NULL,
useSampleFraction = NULL,
T0means = 0,
manifestMeans = 0,
CoTiMAStanctArgs = NULL,
lambda = NULL,
manifestVars = NULL
)

Arguments

cDMAInitFit object to which all single ctsem fits of primary studies has been assigned to (i.e.,
what has been returned by ctmaInit)

primaryStudyList could be a list of primary studies compiled with ctmaPrep that defines the subset
of studies in ctmaInitFit that should actually be used

cluster vector with cluster variables (e.g., countries). Has to be set up carefully. Will be
included in ctmaPrep in later 'CoTiMA' versions.

activeDirectory defines another active directory than the one used in ctmaInitFit

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 mod-
erator 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 subracted from available cores, else value = cores to use

scaleTI scale TI predictors - not recommended until version 0.5.3.1. Does not change
aggregated results anyways, just interpretation of effects for dimmies representing primary studies.

scaleMod scale moderator variables - TRUE (default) recommended for continuous and
categorical moderators, to separate withing and between effects

transfMod more general option to change moderator values. A vector as long as number of
moderators analyzed (e.g., c("mean(x)", "x - median(x)"))
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
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
allInvModel  estimates a model with all parameters invariant (DRIFT, DIFFUSION, T0VAR) if set TRUE (default = FALSE)
customPar  logical. If set TRUE leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (helps since 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
useSampleFraction  to speed up debugging. Provided as fraction (e.g., 1/10).
T0means  Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
manifestMeans  Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
CoTiMAStanctArgs  parameters that can be set to improve model fitting of the ctStanFit Function
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 lower triangular matrix with nrow=n.manifest & ncol=n.manifest.

Value

camaFit 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_Coef, DIFFUSION=model_Diffusion_Coef, T0VAR=model_T0var_Coef, CINT=model_Cint_Coef, MOD=modelMODCoeff, and CLUS=modelCLUSCoeff). 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="drift" and model.type="stanct" ("omx" was deprecated).

Examples

```r
## Not run:
# 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)

## End(Not run)

## 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
CoTiMAMod1onFullFit_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)
```
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)

Description

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

Usage

ctmaFitToPrep(ctmaFitObject = NULL)

Arguments

c DMAFitObject  DMAFitObject
ctmaGetPub

Value

list that could be used for fitting new CoTiMA models with ctmaInit or ctmaFit.

Examples

newStudyList <- ctmaFitToPrep(CoTiMAInitFit_3)

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

dctmaGetPub

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 google 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=0q271ckAAAAJ"),
c("Arnold B.; Bakker", "user=FTl3bwUAAAAJ"),
c("Evangelia; Demerouti", "user=9mj5LvMAAAAJ"),
c("Joachim; Stoebber", "user=T9xdVusAAAAJ"),
c("Claude; Fernet", "user=KwzjP4sAAAAJ"),
c("Frederic; Guay", "user=99vnhX4AAAAJ"),
c("Caroline; Senecal", "user=64ArFWQAAAAJ"),
c("Stéphanie; Austin", "user=PPyTI7EAAAAJ")),
flush=FALSE)

summary(pubList_8)
Description

Fits ctsem models to each primary study in the supplied list of primary studies prepared by `ctmaPrep`.

Usage

cdmaInit(
  primaryStudies = NULL,
  activeDirectory = NULL,
  activateRPB = FALSE,
  checkSingleStudyResults = TRUE,
  digits = 4,
  n.latent = NULL,
  n.manifest = 0,
  lambda = NULL,
  manifestVars = NULL,
  drift = NULL,
  diff = NULL,
  indVarying = FALSE,
  saveRawData = list(),
  coresToUse = c(1),
  silentOverwrite = FALSE,
  saveSingleStudyModelFit = c(),
  loadSingleStudyModelFit = c(),
  scaleTI = NULL,
  scaleTime = NULL,
  optimize = TRUE,
  nopriors = TRUE,
  finishesamples = NULL,
  chains = NULL,
  iter = NULL,
  verbose = NULL,
  customPar = FALSE,
  doPar = 1,
  useSV = FALSE,
  experimental = FALSE,
  T0means = 0,
  manifestMeans = 0,
  CoTiMAStanctArgs = NULL,
  posLL = 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 (hast 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 within a single time point using R-type lower triangular matrix with nrow=n.manifest & ncol=n.manifest.
drift
labels for drift effects. Have to be either of the character strings of the type V1toV2 (= freely estimated) or values (e.g., 0 for effects to be excluded, which is usually not recommended)
diff
labels for diffusion effects. Have to be either of the character strings of the type "diff_etal" or "diff_etal2_etal1" (= freely estimated) or values (e.g., 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 date. 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 interation (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. If set TRUE leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (helps since ctsem > 3.4)

doPar  parallel and multiple fitting if single studies. A value > 1 will fit each study doPar times in parallel mode during which no output is generated (screen remains silent). Useful to obtain best fit.

useSV  if TRUE (default=FALSE) start values will be used if provided in the list of primary studies

experimental  set TRUE to try new pairwise N function

T0means  Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.

manifestMeans  Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.

CoTiMAStanctArgs  parameters that can be set to improve model fitting of the ctStanFit Function

posLL  logical. Allows (default = TRUE) of positive loglik (neg -2ll) values

Value

cmmaFit 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 ommited 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 emprawList (containing the pseudo raw data created), statisticsList (comprising bias 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="stanct" ("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)
## Description

used for consistent labeling of names and parameters

### Usage

```r
cdmaLabels(
  n.latent = NULL,
  n.manifest = 0,
  lambda = NULL,
  manifestVars = NULL,
  drift = NULL,
  diff = NULL,
  invariantDrift = NULL,
  moderatedDrift = NULL,
  equalDrift = NULL,
  T0means = 0,
  manifestMeans = 0
)
```

### Arguments

- `n.latent`: n.latent
- `n.manifest`: n.manifest
- `lambda`: lambda
- `manifestVars`: manifestVar
- `drift`: drift
- `diff`: diffusion
- `invariantDrift`: invariantDrift
- `moderatedDrift`: moderatedDrift
- `equalDrift`: equalDrift
- `T0means`: T0means
- `manifestMeans`: manifestMeans

### 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
Description

Replaces deprecated `ctmaOptimizeInit`, which was limited to initial fitting (i.e., applies `ctmaInit`) of a primary study rFits times to capitalize on chance for obtaining a hard-to-find optimal fit. Now, optimizing a CoTiMA model generated with `ctmaFit` can also be done. Using `ctmaOptimizeFit` could be helpful if a model 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 `ctmaOptimizeFit` is like gambling, hoping that at least one set of starting values (the number it tries is specified in the rFits 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

```r
ctmaOptimizeFit(
  primaryStudies = NULL,
  activeDirectory = NULL,
  problemStudy = NULL,
  rFits = NULL,
  finishesamples = NULL,
  n.latent = NULL,
  coresToUse = c(1),
  indVarying = FALSE,
  scaleTime = NULL,
  randomScaleTime = c(1, 1),
  activateRPB = FALSE,
  checkSingleStudyResults = FALSE,
  customPar = FALSE,
  T0means = 0,
  manifestMeans = 0,
  CoTiMAStanctArgs = NULL,
  CoTiMAFit = NULL,
  CoTiMAInitFit = NULL,
  randomPar = FALSE,
  posLL = TRUE,
  lambda = NULL,
  manifestVars = NULL
)
```

Arguments

- `primaryStudies`: list of primary study information created with `ctmaPrep` or `ctmaFitToPrep`
- `activeDirectory`: `activeDirectory`
- `problemStudy`: number (position in list) where the problem study in `primaryStudies` is found
refits  how many reFits should be done
finishsamples  number of samples to draw (either from hessian based covariance or posterior distribution) for final results computation (default = 1000).
n.latent  number of latent variables of the model (hast to be specified)!
coresToUse  if neg., the value is subtracted from available cores, else value = cores to use
indVarying  control for unobserved heterogeneity by having randomly (inter-individually) varying manifest means
scaleTime  scale time (interval) - sometimes desirable to improve fitting
randomScaleTime  lower and upper limit of uniform distribution from which timeScale argument for ctmaInit is uniformly shuffled (integer)
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.
customPar  logical. If set TRUE (default) leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (helps since ctsem > 3.4)
T0means  Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
manifestMeans  Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
CoTiMAStanctArgs  parameters that can be set to improve model fitting of the ctStanFit Function
CoTiMAFit  a object fitted with ctmaFit
CoTiMAInitFit  the ctmaInitFit object that was used to create the CoTiMAFit object with ctmaFit
randomPar  logical. Overrides arguments used fo customPar and randomly selects custom-Par either TRUE or FALSE
posLL  logical. Allows (default = TRUE) of positive loglik (neg -2ll) values
lambda  R-type matrix with pattern of fixed (=1) or free (any string) loadings.
manifestVars  define the error variances of the manifests within a single time point using R-type lower triangular matrix with nrow=n.manifest & ncol=n.manifest. Useful to check estimates before they are saved.

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

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

summary(optimFit313)

## End(Not run)
```

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 is 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

```r
cmmaOptimizeInit(  
  primaryStudies = NULL,  
  activeDirectory = NULL,  
  problemStudy = NULL,  
  reFits = NULL,  
  finishsamples = NULL,  
  n.latent = NULL,  
  coresToUse = c(1),  
  indVarying = FALSE,  
  randomScaleTime = c(1, 1),  
  activateRPB = FALSE,  
  checkSingleStudyResults = FALSE,  
  customPar = FALSE,  
  T0means = 0,  
  manifestMeans = 0,  
  manifestVars = NULL,  
  CoTiMAStanctArgs = NULL,  
  scaleTime = NULL  
)
```
Arguments

- **primaryStudies**: list of primary study information created with `ctmaPrep` or `ctmaFitToPrep`
- **activeDirectory**: activeDirectory
- **problemStudy**: number (position in list) where the problem study in primaryStudies is found
- **reFits**: how many reFits should be done
- **finishesamples**: number of samples to draw (either from hessian based covariance or posterior distribution) for final results computation (default = 1000).
- **n.latent**: number of latent variables of the model (has to be specified!)
- **coresToUse**: if neg., the value is subtracted from available cores, else value = cores to use
- **indVarying**: control for unobserved heterogeneity by having randomly (inter-individually) varying manifest means
- **randomScaleTime**: lower and upper limit of uniform distribution from which timeScale argument for ctmaInit is uniformly shuffled (integer)
- **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.
- **customPar**: logical. If set TRUE (default) leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (helps since ctsem > 3.4)
- **T0means**: Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
- **manifestMeans**: Default 0 (assuming standardized variables). Can be assigned labels to estimate them freely.
- **manifestVars**: define the error variances of the manifests with a single time point using R-type lower triangular matrix with nrow=n.manifest & ncol=n.manifest.
- **CoTiMAStanctArgs**: parameters that can be set to improve model fitting of the `ctStanFit` Function
- **scaleTime**: scale time (interval) - sometimes desirable to improve fitting

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

```r
## 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)
```

### Description

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

### Usage

```r
chmaPlot(
  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,
  undoTimeScaling = TRUE,
  ...
)
```

### 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

undoTimeScaling  if TRUE, the original time scale is used (timeScale argument possibly used in ctmaInit is undone)

...  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

```r
## 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_BO$activeDirectory <- "/Users/tmp/" # adapt!
plot(CoTiMABiG_D_BO)

## End(Not run)
```
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

cmmaPower(
cmmaInitFit = 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,
scaleTime = NULL
)

Arguments

cmmaInitFit 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
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
customPar  logical. If set TRUE (default) leverages the first pass using priors and ensure that the drift diagonal cannot easily go too negative (helps since ctsem > 3.4)
scaleTime  scale time (interval) - sometimes desirable to improve fitting
Value

cdmaPower 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="stanct" ("omx" was deprecated).

Examples

## Not run:
CoTiMAInitFit_D_BO$activeDirectory <- "/Users/tmp/" # adapt!
CoTiMAPower_D_BO <- cdmaPower(ctmaInitFit=CoTiMAInitFit_D_BO,
                                statisticalPower = c(.50, .80, .95),
                                finishsamples = 10000)

summary(CoTiMAPower_D_BO)

## End(Not run)

description

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

Usage

cdmaPRaw(  
  empCovMat = NULL,
  empNMat = matrix(0, 0, 0),
  empN = NULL,
  studyNumber = NULL,
  empMeanVector = NULL,
  empVarVector = NULL,
  activateRPB = FALSE,
ctmaPrep

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
experimental  set TRUE to try new pairwise N function

Description

Combines information of primary studies into a list object and returns this list. This list is then used as input to fit 'cseme' 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

cdmaPrep(
    selectedStudies = NULL,
    excludedElements = NULL,
    addElements = NULL,
    digits = 4,
    moderatorLabels = NULL,
    moderatorValues = NULL,
    summary = TRUE,
    activeDirectory = NULL
)
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).

addElements
User-added objects that are handled as the weakly predefined objects. The major purpose is to collect information a researcher regards as important.

digits
Rounding used for summary function

moderatorLabels
character vector of names

moderatorValues
list of character vectors

summary
if TRUE (default) creates summary table and xlsx sheets. Could be set to FALSE in case of errors.

activeDirectory
Mandatory. If subsequent fitting is done using different folders or on different computers, it can be changed so that raw data files can be loaded.

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)
             "& 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)
             "& 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")
                                    )
ctmaPub

"3 = medium", "4 = high", "5 = very high")

CoTiMAstudyList_3 <- ctmaPrep(selectedStudies = c(2, 3, 313),
activeDirectory="/user/",
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

chmaPub(
  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

```r
pubResults_6 <- ctmaPub(getPubObj=pubList_8,
                        primaryStudyList=CoTiMAstudyList_6)
summary(pubResults_6)
```

---

**Description**

Internal function to save files

**Usage**

```r
cdmaSaveFile(
    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

Description

This function rescales inits for drifts and sets all other inits to 0 (because it is too complicated to re-scale inits for diffusions). It uses the internal transformations of \texttt{ctStanFit} (i.e., tforms) to transform the raw estimates, then re-scale them, and finally use the inverse of tfrom to supply raw estimates as inits.

Usage

\begin{verbatim}
ctmaScaleInits(
  CoTiMAFit = NULL,
  ctsemFit = NULL,
  newTimeScale = NULL,
  autoRefit = FALSE
)
\end{verbatim}

Arguments

\begin{itemize}
  \item \texttt{CoTiMAFit} \hspace{1cm} Fit object created with \texttt{ctmaFit}
  \item \texttt{ctsemFit} \hspace{1cm} Fit object created with \texttt{ctStanFit}
  \item \texttt{newTimeScale} \hspace{1cm} New Time scale \texttt{ctStanFit}
  \item \texttt{autoRefit} \hspace{1cm} Whether to automatically refit the original model using the new inits
\end{itemize}

Description

Raw data objects are re-shaped (dealing with missing time points, wrong time intervals etc)
Usage

cdmaShapeRawData(
  dataFrame = NULL,
  inputDataFrameFormat = NULL,
  inputTimeFormat = "time",
  missingValues = NA,
  n.manifest = NULL,
  Tpoints = NULL,
  allInputVariablesNames = NULL,
  orderInputVariablesNames = NULL,
  targetInputVariablesNames = NULL,
  targetInputTDpredNames = NULL,
  targetInputTIpredNames = NULL,
  targetInputVariablesNames = NULL,
  outputDataFrameFormat = "long",
  outputVariablesNames = "Y",
  outputTDpredNames = NULL,
  outputTIpredNames = NULL,
  outputTimeVariablesNames = "time",
  outputTimeFormat = "time",
  scaleTime = 1,
  minInterval = 1e-04,
  minToIDelta = NULL,
  maxToIDelta = NULL,
  negToIDelta = FALSE,
  min.val.n.Vars = 1,
  min.val.Tpoints = 1,
  experimental = FALSE
)

Arguments

  dataFrame       an R object containing data
  inputDataFrameFormat   "wide" or "long"
  inputTimeFormat        "time" (default) or "delta"
  missingValues       Missing value indicator, e.g., -999 or NA (default)
  n.manifest           Number of process variables (e.g, 2 in a bivariate model)
  Tpoints               Number of time points in the data frame
  allInputVariablesNames vector of all process variable names, time dependent predictor names, time inde-
                           pendent predictor names, and names of times/deltas. Only required if the
                           dataFrame does not have column names.
  orderInputVariablesNames = "names" vs "time" (e.g., names: X1, X2, X3, Y1, Y2, X3 vs time: X1, Y1,
                                        X2, Y2, ... ). For ctesem/CoTiMA, the output file will order by time.
**targetInputVariablesNames**

= the process variables in the dataFrame that should be used (in "names" or in "times" order; e.g., c("X1", "X3", "Y1", "X3"). This is used to delete variables from the data frame that are not required.

**targetInputTDpredNames**

The actual time dependent (TD) predictor variable names, e.g. 3, or 6, or 9, ... names if Tpoints = 3. Internally, each of the 3, 6, etc represents one TDpred. One typically does NOT have TD predictors in a CoTiMA.

**targetInputTIpredNames**

time independent (TI) predictor names in the dataFrame. One typically does NOT have TI predictors in CoTiMA except it uses raw data only, where TIpreds are available for individual cases.

**targetTimeVariablesNames**

The time variables names in the dataFrame. They also define which Tpoints will be included in the output file, e.g., c("Time4", "Time9").

**outputDataFrameFormat**

"long" (default) or "wide"

**outputVariablesNames**

"Y" (default; creates Y1_T0, Y2_T0, Y1_T1, Y2_T1, etc.), but can also be, e.g., c("X", "Y"; creates X_T0, Y_T0, X_T1, Y_T1, etc.).

**outputTDpredNames**

Will become "TD" if not specified

**outputTIpredNames**

Will become "TI" if not specified

**outputTimeVariablesNames**

"time" (default)

**outputTimeFormat**

"time" (default) or "delta"

**scaleTime**

= A scalar that is used to multiply the time variable. Typical use is rescaling primary study time to the time scale used in other primary studies. For example, scaleTime=1/(60 x 60 x 24 x 365.25) rescales time provided in seconds (frequent case when imported from SPSS) into years (60sec x 60min x 24hrs x 365.25 days incl. leap years).

**minInterval**

A parameter (default = 0.0001) supplied to ctIntervalise. Set to smaller values than any possible observed measurement interval, but larger than 0.0001. The value is used for indicating unavailable time interval information (caused by missing values) because NA is technically not possible for time intervals.

**minTolDelta**

Set, e.g. to 1/24, to delete variables from time points that are too close (e.g., 1hr; or even before) after another time point. Could be useful to delete values generated by unreliable responding, e.g., in diary studies. Note that minTolDelta applies to the time intervals AFTER the scaleTime argument has applied (i.e., scaleTime may need adaptation for each primary study, but minTolDelta does not).

**maxTolDelta**

Set, e.g., to 7, to delete variables from time points that are too far after another time point (e.g., 7 days, if all participants should have responded within a week).
Note that maxTolDelta applies to the time intervals AFTER the scaleTime argument has applied (i.e., scaleTime may need adaptation for each primary study, but minTolDelta does not).

negTolDelta FALSE (default) or TRUE. Delete entire cases that have at least one negative delta (‘unreliable responding’; use minTolDelta to delete certain variables only)

min.val.n.Vars min.val.n.Vars = Minimum no. of valid variables. Default = 1 (retains cases with only 1 valid variable). 0 would retain cases with all variables missing (not very useful). Retaining participants who provide a single valid variable is technically possible, but these participants contribute to the estimation of the variance/mean of this variable only. Since variance/mean are 1/0 in most CoTiMA applications, this is not very informative but at the cost of additional computational burden. Setting min.val.n.Vars = 2 is recommended.

min.val.Tpoints Minimum no. of valid Tpoints (i.e. Tpoints where min.val.n.Vars is met). Default = 1 retains participants with full set of valid variables at least at one single Tpoint (which will become T0). Setting min.val.Tpoints = 2 or higher values retains participants which provide longitudinal information. Since T0 covariances are usually not too interesting, min.val.Tpoints = 2 may be more reasonable than the default = 1.

experimental FALSE (default) or TRUE. Deprecated.

Value
A reshaped raw data file

Examples

```r
## Not run:
tmpData <- data.frame(matrix(c(1, 2, 1, 2, 1, 2, 11, 26, 1,
                                NA, NA, 3, NA, 3, NA, 12, 27, 1,
                                1, 2, 1, 2, 1, 2, NA, 24, 0 ),
                             nrow=3, byrow=TRUE))
colnames(tmpData) <- c("first_T0", "second_T0", "first_T1", "second_T1",
                        "TD1_0", "TD1_1",
                        "time1", "time2", "sex")
shapedData <- ctmaShapeRawData(dataFrame=tmpData,
                               inputDataFrameFormat="wide",
                               inputTimeFormat="time",
                               n.manifest=2,
                               Tpoints=2,
                               orderInputVariablesNames="time",
                               targetInputVariablesNames=c("first_T0", "second_T0",
                                                             "first_T1", "second_T1"),
                               targetInputTIpredNames=c("TD1_0", "TD1_1"),
                               targetInputTDpredNames="sex",
                               targetTimeVariablesNames=c("time1", "time2"),
                               scaleTime=1/12,
                               maxTolDelta=1.2)
head(shapedData)
```
## End(Not run)

chmaStanResample

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

Usage
chmaStanResample(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

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

Usage
chmaSV(
  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 \texttt{ctmaInit})
activeDirectory defines another active directory than the one used in \texttt{ctmaInit}
primaryStudies if \texttt{ctmaInitFit} does not contain the primaryStudies object created with \texttt{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

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

---

\texttt{delta_t128} \hspace{1cm} \textit{delta_t128 example vector}

Description
delta_t128 example vector

Usage
delta_t128

Format
An object of class \texttt{numeric} of length 1.

Author(s)
C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
**delta_t18**

**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**

**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_t201**  
*delta_t201 example vector*

**Description**

*delta_t201 example vector*

**Usage**

`delta_t201`

**Format**

An object of class `numeric` of length 2.

**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**

**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**

**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  

**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  

**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  

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>

demands201  

demands201 example vector

Description  
demands201 example vector

Usage  
demands201

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

**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

**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  

**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  

**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>

empcov201

empcov201 example matrix

Description

empcov201 example matrix

Usage

empcov201

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>
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 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 example matrix

**Description**

empcov32 example matrix

**Usage**

```r
empcov32
```

**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>

---

malePercent128 example vector

**Description**

malePercent128 example vector

**Usage**

```r
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>
malePercent201

**malePercent201** example vector

**Description**

malePercent201 example vector

**Usage**

malePercent201

**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  

**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  

**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 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 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>
**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>

---

**Description**

moderator201 example vector

**Usage**

moderator201

**Format**

An object of class numeric of length 2.

**Author(s)**

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
**moderator3**  

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

**Usage**

moderator313

**Format**

An object of class numeric of length 2.

**Author(s)**

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

---

**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  occupation2 example vector

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>
**occupation201**

**Description**

occupation201 example vector

**Usage**

occupation201

**Format**

An object of class character of length 1.

**Author(s)**

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

---

**occupation3**

**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>
### 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>

### 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>
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>

Description

call \texttt{ctmaPlot} if a CoTiMAFit object is supplied to \texttt{plot()}

Usage

\begin{verbatim}
## S3 method for class 'CoTiMAFit'
plot(x, ...)
\end{verbatim}

Arguments

\begin{itemize}
\item \texttt{x} list
\item \texttt{...} further arguments to be passed through to \texttt{summary()}
\end{itemize}

Value

returns a call to \texttt{ctmaPlot}, which is used to plot CoTiMA fit objects
### pubList_8

**Description**

pubList_8 example list

**Usage**

`pubList_8`

**Format**

An object of class `CoTiMAFit` of length 9.

**Author(s)**

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

---

### rawData128

**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**

**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**

**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**

```r
sampleSize2
```

**Format**

An object of class `numeric` of length 1.

**Author(s)**

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

### sampleSize201

**sampleSize201 example vector**

**Description**

sampleSize201 example vector

**Usage**

```r
sampleSize201
```

**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>
**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>

**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>

---

**source201**

*source201 example vector*

**Description**

*source201 example vector*

**Usage**

*source201*

**Format**

An object of class `character` of length 6.

**Author(s)**

C. Dormann & M. Homburg <CoTiMA@uni-mainz.org>
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 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>
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 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>
<table>
<thead>
<tr>
<th>targetVariables2</th>
<th>targetVariables2 example vector</th>
</tr>
</thead>
</table>

**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>

<table>
<thead>
<tr>
<th>targetVariables3</th>
<th>targetVariables3 example vector</th>
</tr>
</thead>
</table>

**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, 29

* data
  A128, 5
  A313, 6
  addedByResearcher2, 6
  addedByResearcher3, 7
  addedByResearcher313, 7
  ageM128, 8
  ageM18, 8
  ageM2, 9
  ageM201, 9
  ageM3, 10
  ageM313, 10
  ageM32, 11
  ageSD128, 11
  ageSD18, 12
  ageSD2, 12
  ageSD201, 13
  ageSD3, 13
  ageSD313, 14
  ageSD32, 14
  alphas128, 15
  alphas313, 15
  burnout128, 16
  burnout18, 16
  burnout2, 17
  burnout201, 17
  burnout3, 18
  burnout313, 18
  burnout32, 19
  combineVariables128, 19
  combineVariablesNames128, 20
  CoTiMABiG_D_BO, 20
  CoTiMAFullFit_3, 21
  CoTiMAFullFit_6, 21
  CoTiMAFullFit_6_new, 22
  CoTiMAFullInv23Fit_6, 22
  CoTiMAFullInvEq23Fit_6, 23
  CoTiMAInitFit_3, 23
  CoTiMAInitFit_6, 24
  CoTiMAInitFit_6_new, 24
  CoTiMAInitFit_6_NUTS, 25
  CoTiMAInitFit_D_BO, 25
  CoTiMAMod1onFullFit_6, 26
  CoTiMAMod1onFullFit_6_cats12, 26
  CoTiMAMod2on23Fit_6, 27
  CoTiMAoptimFit313, 27
  CoTiMAPart134Inv3Fit_6, 28
  CoTiMAPower_D_BO, 28
  CoTiMAstudyList_3, 29
  CoTiMAstudyList_6, 30
  CoTiMAstudyList_6_new, 30
  country128, 31
  country18, 31
  country2, 32
  country201, 32
  country3, 33
  country313, 33
  country32, 34
  delta_t128, 72
  delta_t18, 73
  delta_t2, 73
  delta_t201, 74
  delta_t3, 74
  delta_t313, 75
  delta_t32, 75
  demands128, 76
  demands18, 76
  demands2, 77
  demands201, 77
  demands3, 78
  demands313, 78
  demands32, 79
dl_link, 79
  empcov128, 80
  empcov18, 80
  empcov2, 81
empcov201, 81
empcov3, 82
empcov313, 82
empcov32, 83
malePercent28, 83
malePercent18, 84
malePercent2, 84
malePercent201, 85
malePercent3, 85
malePercent313, 86
malePercent32, 86
moderator128, 87
moderator18, 87
moderator2, 88
moderator201, 88
moderator3, 89
moderator313, 89
moderator32, 89
moderatorLabels, 90
moderatorValues, 91
occupation128, 91
occupation18, 92
occupation2, 92
occupation201, 93
occupation3, 93
occupation313, 94
occupation32, 94
pairwiseN128, 95
pubList_8, 96
rawData128, 96
recodeVariables128, 96
results128, 97
sampleSize128, 98
sampleSize18, 98
sampleSize2, 99
sampleSize201, 99
sampleSize3, 100
sampleSize313, 100
sampleSize32, 101
source128, 101
source2, 102
source201, 102
source3, 103
source313, 103
targetVariables128, 104
targetVariables2, 105
targetVariables3, 105
targetVariables313, 106
variableNames128, 106
A128, 5
A313, 6
addedByResearcher2, 6
addedByResearcher3, 7
addedByResearcher313, 7
ageM128, 8
ageM18, 8
ageM2, 9
ageM201, 9
ageM3, 10
ageM313, 10
ageM32, 11
ageSD128, 11
ageSD18, 12
ageSD2, 12
ageSD201, 13
ageSD3, 13
ageSD313, 14
ageSD32, 14
alphas128, 15
alphas313, 15
burnout128, 16
burnout18, 16
burnout2, 17
burnout201, 17
burnout3, 18
burnout313, 18
burnout32, 19
combineVariables128, 19
combineVariablesNames128, 20
CoTimABiG_D_BO, 20
CoTIMAFullFit_3, 21
CoTIMAFullFit_6, 21
CoTIMAFullFit_6_new, 22
CoTIMAFullInv23Fit_6, 22
CoTIMAFullInvEq23Fit_6, 23
CoTIMAInitFit_3, 23
CoTIMAInitFit_6, 24
CoTIMAInitFit_6_new, 24
CoTIMAInitFit_6_NUTS, 25
CoTIMAInitFit_D_BO, 25
CoTIMAMod1onFullFit_6, 26
CoTIMAMod1onFullFit_6_cats12, 26
CoTIMAMod2on23Fit_6, 27
CoTIMAoptimFit313, 27
INDEX

CoTiMAPart134Inv3Fit_6, 28
CoTiMAPower_D_BO, 28
CoTiMAsanctArgs, 29
CoTiMAsudyList_3, 29
CoTiMAsudyList_6, 30
CoTiMAsudyList_6_new, 30
country128, 31
country18, 31
country2, 32
country201, 32
country3, 33
country313, 33
country32, 34
c DMAAllInvFit, 34
c DMABig, 36
c DMABiGOMX, 37
c DMACombPRaw, 38
c DMACompFit, 39
c DMACorRel, 39
c DMAEmpCov, 40
c DMAEqual, 39, 42
ctmaf it, 39, 42, 43, 48, 53, 54, 67
c DMAfitList, 47
c DMAfitToPrep, 47, 53, 56
c DMAGetPub, 48, 65
c DMAInit, 36, 44, 48, 49, 53, 55, 58, 59, 72
c DMALabels, 52
c DMAOptimizeFit, 53, 53
c DMAOptimizeInit, 53, 55
c DMAPlot, 57, 95
c DMAPower, 59
c DMAPr AW, 61
c DMAPrep, 44, 47, 49, 50, 53, 56, 62, 72
c DMAPub, 65
c DMASaveFile, 66
c DMAShapeInits, 67
c DMAShape RawData, 67
c DMAStanResample, 71
c DMAsv, 71
c DMAFit, 35, 45, 51, 54, 56, 67
del etat128, 72
del etat18, 73
del etat2, 73
del etat201, 74
del etat3, 74
del etat313, 75
del etat32, 75
d emands128, 76
d emands18, 76
d emands2, 77
d emands201, 77
d emands3, 78
d emands313, 78
d emands32, 79
dl_link, 79
d empcov128, 80
d empcov18, 80
d empcov2, 81
d empcov201, 81
d empcov3, 82
d empcov313, 82
d empcov32, 83
malePercent128, 83
malePercent18, 84
malePercent2, 84
malePercent201, 85
malePercent3, 85
malePercent313, 86
malePercent32, 86
malePercent313, 86
moderator128, 87
moderator18, 87
moderator2, 88
moderator201, 88
moderator3, 89
moderator313, 89
moderator32, 90
moderatorLabels, 90
moderatorValues, 91
occupation128, 91
occupation18, 92
occupation2, 92
occupation201, 93
occupation3, 93
occupation313, 94
occupation32, 94
pairwiseN128, 95
plot CoTiMAFit, 95
pubList_8, 96
rawData128, 96
recodeVariables128, 97
results128, 97
sampleSize128, 98
INDEX

sampleSize18, 98
sampleSize2, 99
sampleSize201, 99
sampleSize3, 100
sampleSize313, 100
sampleSize32, 101
source128, 101
source2, 102
source201, 102
source3, 103
source313, 103
summary.CoTiMAFit, 104

targetVariables128, 104
targetVariables2, 105
targetVariables3, 105
targetVariables313, 106

variableNames128, 106