# Package 'manCULTA'

July 30, 2025

Title Common and Unique Latent Transition Analysis (CULTA) as a Way to Examine the Trait-State Dynamics of Alcohol IntoxicationVersion 1.0.0

**Description** Research compendium for the manuscript
Pesigan, I. J. A., Russell, M. A., Chow, S.-M. (Under Review).
Common and Unique Latent Transition Analysis (CULTA) as a W

Common and Unique Latent Transition Analysis (CULTA) as a Way to Examine the Trait-State Dynamics of Alcohol Intoxication.

<doi:10.0000/0000000000>.

URL https://github.com/jeksterslab/manCULTA,

https://jeksterslab.github.io/manCULTA/, https://osf.io/gtdmr/,

https://doi.org/10.0000/0000000000

BugReports https://github.com/jeksterslab/manCULTA/issues

**License** GPL (>= 3)

**Encoding** UTF-8

LazyData true

**Roxygen** list(markdown = TRUE)

**Depends** R (>= 4.0.0)

LinkingTo Rcpp, RcppArmadillo

Imports Rcpp, ggplot2

Suggests knitr, rmarkdown, testthat, lavaan

RoxygenNote 7.3.2

**NeedsCompilation** yes

**Author** Ivan Jacob Agaloos Pesigan [aut, cre, cph] (ORCID:

<https://orcid.org/0000-0003-4818-8420>)

Maintainer Ivan Jacob Agaloos Pesigan < r. jeksterslab@gmail.com>

2 Contents

# **Contents**

AIC.fitculta	3
anova.fitculta	5
as.data.frame.simculta	8
as.matrix.simculta	9
	10
	13
1	15
confint.fitculta	16
E	18
1 0	20
e	23
	25
8	26
$\varepsilon$	28
8 17	29
FigIC	30
8	31
8	32
FigRMSECULTAEst	34
FigRMSELTAEst	35
FitCULTA1Profile	37
FitCULTA2Profiles	39
FitLTA2Profiles	42
FitRILTA2Profiles	45
GenCULTA2Profiles	48
GenData	55
InputCULTA1Profile	55
InputCULTA2Profiles	58
InputLTA2Profiles	61
InputRILTA2Profiles	63
logLik.fitculta	66
params	69
print.fitculta	<b>7</b> 0
print.simculta	73
results_culta_est	74
results_entropy	75
results_ic	76
results_lta_est	76
Sim	78
SimFitCULTA1Profile	<b>7</b> 9
SimFitCULTA2Profiles	80
SimFitLTA2Profiles	81
SimFitRILTA2Profiles	82
	83
	84
SimProj	85
SimProj	83

AIC.fitculta 3

muex	Ц	JU
Index	10	00
	WriteData	<del>)</del> 8
	vcov.fitculta	
	SumParamsCULTA2Profiles	
	summary.fitculta	<del>)</del> 3
	SumLTAParamsRILTA2Profiles	
	SumLTAParamsLTA2Profiles	<b>)</b> 1
	SumLTAParamsCULTA2Profiles	<del>)</del> 0
	SumFitRILTA2Profiles	39
	SumFitLTA2Profiles	38
	SumFitCULTA2Profiles	37
	SumFitCULTA1Profile	36
	Sum	35

AIC.fitculta

Akaike's Information Criterion

# Description

Akaike's Information Criterion

# Usage

```
## S3 method for class 'fitculta'
AIC(object, ...)
```

# Arguments

object Object of class fitculta.
... additional arguments.

# Value

Returns Akaike's information criterion (AIC).

# Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
```

4 AIC.fitculta

```
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 \# number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x < - 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 < -0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
 data = 1,
 nrow = p,
  ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
```

anova.fitculta 5

```
c(-0.278, -0.165, -0.199, -0.148)
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
AIC(fit)
## End(Not run)
```

anova.fitculta

Compare Two Nested fitculta Models Using Scaled Chi-Square Difference Test

# **Description**

This function compares two fitculta models using the Satorra-Bentler scaled chi-square difference test based on log-likelihoods, number of free parameters, and scaling correction factors. It also returns model fit indices for both models, including AIC, BIC, adjusted BIC, and entropy.

#### Usage

```
## S3 method for class 'fitculta'
anova(object, other, ...)
```

6 anova.fitculta

### **Arguments**

```
object Model object of class fitculta.

other Another model object of class fitculta.

... additional arguments.
```

#### Value

A list with two elements:

fit A matrix summarizing model fit indices for both models, including: logLik, df (number of free parameters), correction (scaling factor), AIC, BIC, aBIC (adjusted BIC), and entropy.

diff A named numeric vector with scaled chi-square difference, degrees of freedom difference, and p-value.

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
 psi_p_2,
```

anova.fitculta 7

```
psi_p_3,
 psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = q
)
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
  nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
  c(2.253, 1.493, 1.574, 1.117),
  c(-0.278, -0.165, -0.199, -0.148)
# data generation ------
data <- GenCULTA2Profiles(</pre>
  n = n,
  m = m,
  mu_x = mu_x
  sigma_x = sigma_x,
  nu_0 = nu_0,
  kappa_0 = kappa_0,
  alpha_0 = alpha_0,
  beta_00 = beta_00,
  gamma_00 = gamma_00,
  gamma_10 = gamma_10,
  mu_t = mu_t
  psi_t = psi_t,
  mu_p = mu_p,
  psi_p = psi_p,
  common_trait_loading = common_trait_loading,
  common_state_loading = common_state_loading,
  phi_0 = phi_0,
  phi_1 = phi_1,
  psi_s0 = psi_s0,
  psi_s = psi_s,
  theta = theta,
  mu_profile = mu_profile
```

8 as.data.frame.simculta

```
# model fitting -----
# NOTE: Model fitting takes time
one_profile <- FitCULTA1Profile(data = data)
two_profiles <- FitCULTA2Profiles(data = data)
anova(one_profile, two_profiles)
## End(Not run)</pre>
```

as.data.frame.simculta

Coerce an Object of Class simculta to a Data Frame

# **Description**

Coerce an Object of Class simculta to a Data Frame

# Usage

```
## S3 method for class 'simculta' as.data.frame(x, ...)
```

#### **Arguments**

x Object of class simculta.

... Additional arguments.

### Author(s)

Ivan Jacob Agaloos Pesigan

```
x <- GenCULTA2Profiles(
    n = 10,
    m = 6,
    common_trait_loading = matrix(
        data = c(1, 1.25, 1.50, 1.75),
        ncol = 1
    ),
    common_state_loading = matrix(
        data = c(1, 1.5, 1.75, 2.00),
        ncol = 1
    ),
    mu_t = NULL,
    psi_t = NULL,
    mu_p = NULL,
    psi_p = NULL,</pre>
```

as.matrix.simculta 9

```
theta = diag(4),
 mu_profile = cbind(
   c(-3, -3, -3, -3),
   c(3, 3, 3, 3)
 ),
 mu_x = 0,
 sigma_x = 1,
 nu_0 = -3.563,
 kappa_0 = 0.122,
 alpha_0 = -3.586,
 beta_00 = 2.250,
 gamma_00 = 0.063,
 gamma_10 = 0.094,
 phi_0 = 0.311,
 phi_1 = 0,
 psi_s0 = 0.151,
 psi_s = 0.290
)
as.data.frame(x)
```

as.matrix.simculta

Coerce an Object of Class simculta to a Matrix

# Description

Coerce an Object of Class simculta to a Matrix

# Usage

```
## S3 method for class 'simculta' as.matrix(x, ...)
```

# **Arguments**

x Object of class simculta.

... Additional arguments.

#### Author(s)

Ivan Jacob Agaloos Pesigan

```
x <- GenCULTA2Profiles(
    n = 10,
    m = 6,
    common_trait_loading = matrix(
        data = c(1, 1.25, 1.50, 1.75),</pre>
```

10 BIC.fitculta

```
ncol = 1
 ),
 common_state_loading = matrix(
   data = c(1, 1.5, 1.75, 2.00),
   ncol = 1
 ),
 mu_t = NULL,
 psi_t = NULL,
 mu_p = NULL
 psi_p = NULL,
 theta = diag(4),
 mu_profile = cbind(
   c(-3, -3, -3, -3),
   c(3, 3, 3, 3)
 ),
 mu_x = 0,
 sigma_x = 1,
 nu_0 = -3.563,
 kappa_0 = 0.122,
 alpha_0 = -3.586,
 beta_00 = 2.250,
 gamma_00 = 0.063,
 gamma_10 = 0.094,
 phi_0 = 0.311,
 phi_1 = 0,
 psi_s0 = 0.151,
 psi_s = 0.290
as.matrix(x)
```

BIC.fitculta

Bayesian Information Criterion

# Description

**Bayesian Information Criterion** 

# Usage

```
## S3 method for class 'fitculta'
BIC(object, adjust = FALSE, ...)
```

# **Arguments**

```
    object Object of class fitculta.
    adjust Logical. If adjust = TRUE, return the sample size adjusted BIC.
    additional arguments.
```

BIC.fitculta 11

# Value

Returns Bayesian information criterion (BIC).

#### Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
```

12 BIC.fitculta

```
)
# state parameters
common_state_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = 1
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
BIC(fit)
BIC(fit, adjust = TRUE)
## End(Not run)
```

coef.fitculta 13

coef.fitculta

Parameter Estimates

# **Description**

Parameter Estimates

# Usage

```
## S3 method for class 'fitculta'
coef(object, ...)
```

# **Arguments**

```
object Object of class fitculta.
... additional arguments.
```

#### Value

Returns a vector of parameter estimates.

# Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x < - 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
```

14 coef.fitculta

```
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
 nrow = p,
 ncol = q
)
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
  c(2.253, 1.493, 1.574, 1.117),
  c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
  n = n,
  m = m,
  mu_x = mu_x,
  sigma_x = sigma_x,
  nu_0 = nu_0,
  kappa_0 = kappa_0,
  alpha_0 = alpha_0,
  beta_00 = beta_00,
```

Compress 15

```
gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
coef(fit)
## End(Not run)
```

Compress

Compress Replication

# Description

Compress Replication

#### Usage

```
Compress(taskid, repid, output_folder)
```

### **Arguments**

taskid Positive integer. Task ID.
repid Positive integer. Replication ID.
output\_folder Character string. Output folder.

#### Value

The output is saved as an external file in output\_folder.

# Author(s)

Ivan Jacob Agaloos Pesigan

16 confint.fitculta

confint.fitculta

Confidence Intervals for Parameter Estimates

# Description

Confidence Intervals for Parameter Estimates

#### Usage

```
## S3 method for class 'fitculta'
confint(object, parm = NULL, level = 0.95, ...)
```

# Arguments

object Object of class fitculta.

parm a specification of which parameters are to be given confidence intervals, either

a vector of numbers or a vector of names. If missing, all parameters are consid-

ered.

level the confidence level required.

... additional arguments.

#### Value

Returns a matrix of confidence intervals.

### Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments ------
# random seed for reproducibility
set.seed(42)

# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension

# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566

# profile membership and transition parameters</pre>
```

confint.fitculta 17

```
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
 psi_p_2,
 psi_p_3,
 psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = q
# state parameters
common_state_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
```

18 converged

```
kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
confint(fit, level = 0.95)
## End(Not run)
```

converged

Convergence Status of a Model Fit

# Description

Checks whether the model fitting procedure for an object of class fitculta has successfully converged.

#### Usage

```
converged(object, ...)
```

#### **Arguments**

```
object Object of class fitculta.
... additional arguments.
```

#### Value

Logical. TRUE if the model has converged, FALSE otherwise.

converged 19

#### Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments ------
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
)
# state parameters
common_state_loading <- matrix(</pre>
```

20 entropy

```
data = 1,
  nrow = p,
  ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
  c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
  n = n,
  m = m,
  mu_x = mu_x
  sigma_x = sigma_x,
  nu_0 = nu_0,
  kappa_0 = kappa_0,
  alpha_0 = alpha_0,
  beta_00 = beta_00,
  gamma_00 = gamma_00,
  gamma_10 = gamma_10,
  mu_t = mu_t
  psi_t = psi_t,
  mu_p = mu_p,
  psi_p = psi_p,
  common_trait_loading = common_trait_loading,
  common_state_loading = common_state_loading,
  phi_0 = phi_0,
  phi_1 = phi_1,
  psi_s0 = psi_s0,
  psi_s = psi_s,
  theta = theta,
  mu_profile = mu_profile
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
converged(fit)
## End(Not run)
```

entropy

entropy 21

# **Description**

Entropy

# Usage

```
entropy(object, ...)
```

# **Arguments**

```
object Object of class fitculta. ... additional arguments.
```

# Value

Returns Akaike's information criterion (AIC).

#### Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments ------
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x < - 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
```

22 entropy

```
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
 ncol = q
)
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
  nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
  c(2.253, 1.493, 1.574, 1.117),
  c(-0.278, -0.165, -0.199, -0.148)
)
# data generation -------------
data <- GenCULTA2Profiles(</pre>
  n = n,
  m = m,
  mu_x = mu_x,
  sigma_x = sigma_x,
  nu_0 = nu_0,
  kappa_0 = kappa_0,
  alpha_0 = alpha_0,
  beta_00 = beta_00,
  gamma_00 = gamma_00,
  gamma_10 = gamma_10,
  mu_t = mu_t
  psi_t = psi_t,
  mu_p = mu_p,
  psi_p = psi_p,
  common_trait_loading = common_trait_loading,
  common_state_loading = common_state_loading,
  phi_0 = phi_0,
```

FigBiasCULTAEst 23

FigBiasCULTAEst

Plot Relative Bias (CULTA Estimates)

# **Description**

Plot relative bias for common CULTA estimates.

#### Usage

```
FigBiasCULTAEst(results_culta_est)
```

#### **Arguments**

```
results_culta_est
```

Summary CULTA estimates results data frame.

# **Details**

The parameters are indexed as follows:

- 1  $\phi_0$  parameter. Autoregressive coefficient for profile 0.
- 2  $\phi_1$  parameter. Autoregressive coefficient for profile 1.
- 3  $\psi_T$  parameter. Variance in the common trait; reflects stable between-person differences.
- 4  $\lambda_{t2}$  parameter. Factor loading for the common trait and item 2.
- 5  $\lambda_{t3}$  parameter. Factor loading for the common trait and item 3.
- **6**  $\lambda_{t4}$  parameter. Factor loading for the common trait and item 4.
- 7  $\psi_{p11}$  parameter. Trait-specific item 1 variance.
- 8  $\psi_{p22}$  parameter. Trait-specific item 2 variance.
- 9  $\psi_{p33}$  parameter. Trait-specific item 3 variance.
- 10  $\psi_{p44}$  parameter. Trait-specific item 4 variance.

24 FigBiasCULTAEst

11  $\psi_{s0}$  parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.

- 12  $\psi_s$  parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.
- 13  $\lambda_{s2}$  parameter. Factor loading for the common state and item 2.
- 14  $\lambda_{s3}$  parameter. Factor loading for the common state and item 3.
- 15  $\lambda_{s4}$  parameter. Factor loading for the common state and item 4.
- **16**  $\theta_{11}$  parameter. Unique state variance for item 1.
- 17  $\theta_{22}$  parameter. Unique state variance for item 2.
- **18**  $\theta_{33}$  parameter. Unique state variance for item 3.
- 19  $\theta_{44}$  parameter. Unique state variance for item 4.
- **20**  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- 21  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 22  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- 23  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 24  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 25  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- **26**  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 27  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- 28  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.
- **29**  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.
- **30**  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.
- **31**  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.
- 32  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.
- 33  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Figure Functions: FigBiasLTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigEntropy(), FigIC(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSECULTAEst(), FigRMSECULTAEst()
```

```
## Not run:
data(results_culta_est, package = "manCULTA")
FigBiasCULTAEst(results_culta_est)
## End(Not run)
```

FigBiasLTAEst 25

FigBiasLTAEst

Plot Relative Bias (LTA Estimates)

#### **Description**

Plot relative bias for common LTA estimates.

# Usage

```
FigBiasLTAEst(results_lta_est)
```

# Arguments

results\_lta\_est

Summary LTA estimates results data frame.

#### **Details**

The parameters are indexed as follows:

- 1  $\theta_{11}$  parameter. Unique state variance for item 1.
- 2  $\theta_{22}$  parameter. Unique state variance for item 2.
- 3  $\theta_{33}$  parameter. Unique state variance for item 3.
- **4**  $\theta_{44}$  parameter. Unique state variance for item 4.
- 5  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- **6**  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 7  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- **8**  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- **9**  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 10  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- 11  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 12  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- 13  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.
- 14  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.
- 15  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.
- 16  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.
- 17  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.
- 18  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigEntropy(), FigIC(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSECULTAEst(), FigRMSELTAEst()
```

### **Examples**

```
## Not run:
data(results_lta_est, package = "manCULTA")
FigBiasLTAEst(results_lta_est)
## End(Not run)
```

FigCoverageCULTAEst

Plot Coverage Probabilities (CULTA Estimates)

#### **Description**

Plot coverage probabilities for CULTA estimates.

#### Usage

```
FigCoverageCULTAEst(results_culta_est)
```

# Arguments

```
results_culta_est
```

Summary CULTA estimates results data frame.

#### **Details**

The parameters are indexed as follows:

- 1  $\phi_0$  parameter. Autoregressive coefficient for profile 0.
- **2**  $\phi_1$  parameter. Autoregressive coefficient for profile 1.
- 3  $\psi_T$  parameter. Variance in the common trait; reflects stable between-person differences.
- **4**  $\lambda_{t2}$  parameter. Factor loading for the common trait and item 2.
- 5  $\lambda_{t3}$  parameter. Factor loading for the common trait and item 3.
- **6**  $\lambda_{t4}$  parameter. Factor loading for the common trait and item 4.
- 7  $\psi_{p11}$  parameter. Trait-specific item 1 variance.
- 8  $\psi_{p22}$  parameter. Trait-specific item 2 variance.
- 9  $\psi_{v33}$  parameter. Trait-specific item 3 variance.

- 10  $\psi_{p44}$  parameter. Trait-specific item 4 variance.
- 11  $\psi_{s0}$  parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.
- 12  $\psi_s$  parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.
- 13  $\lambda_{s2}$  parameter. Factor loading for the common state and item 2.
- 14  $\lambda_{s3}$  parameter. Factor loading for the common state and item 3.
- 15  $\lambda_{s4}$  parameter. Factor loading for the common state and item 4.
- **16**  $\theta_{11}$  parameter. Unique state variance for item 1.
- 17  $\theta_{22}$  parameter. Unique state variance for item 2.
- **18**  $\theta_{33}$  parameter. Unique state variance for item 3.
- **19**  $\theta_{44}$  parameter. Unique state variance for item 4.
- **20**  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- 21  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 22  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- 23  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 24  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 25  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- **26**  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 27  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- **28**  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.
- **29**  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.
- **30**  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.
- 31  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.
- 32  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.
- 33  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageLTAEst(), FigEntropy(), FigIC(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSECULTAEst()

28 FigCoverageLTAEst

#### **Examples**

```
## Not run:
data(results_culta_est, package = "manCULTA")
FigCoverageCULTAEst(results_culta_est)
## End(Not run)
```

FigCoverageLTAEst

Plot Coverage Probabilities (LTA Estimates)

#### **Description**

Plot coverage probabilities for common LTA estimates.

#### Usage

```
FigCoverageLTAEst(results_lta_est)
```

#### **Arguments**

```
results_lta_est
```

Summary LTA estimates results data frame.

#### **Details**

The parameters are indexed as follows:

- 1  $\theta_{11}$  parameter. Unique state variance for item 1.
- 2  $\theta_{22}$  parameter. Unique state variance for item 2.
- **3**  $\theta_{33}$  parameter. Unique state variance for item 3.
- **4**  $\theta_{44}$  parameter. Unique state variance for item 4.
- 5  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- 6  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 7  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- 8  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 9  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 10  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- 11  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 12  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- 13  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.

FigEntropy 29

```
14 \mu_{40} parameter. Profile specific mean for profile 0 and item 4.
```

15  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.

16  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.

17  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.

18  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageCULTAEst(), FigEntropy(), FigIC(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSECULTAEst()
```

# Examples

```
## Not run:
data(results_lta_est, package = "manCULTA")
FigCoverageLTAEst(results_lta_est)
## End(Not run)
```

FigEntropy

Plot Entropy

#### **Description**

Plot average entropy.

#### Usage

FigEntropy(results\_entropy)

#### Arguments

results\_entropy

Summary entropy results data frame.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigIC(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSECULTAEst(), FigRMSECULTAEst()
```

30 FigIC

# **Examples**

```
## Not run:
data(results_entropy, package = "manCULTA")
FigEntropy(results_entropy)
## End(Not run)
```

FigIC

Plot Information Criteria

# Description

Plot average information criteria.

# Usage

```
FigIC(results_ic)
```

# **Arguments**

results\_ic Summary information criteria results data frame.

# Author(s)

Ivan Jacob Agaloos Pesigan

### See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigEntropy(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSECULTAEst(), FigRMSELTAEst()
```

```
## Not run:
data(results_ic, package = "manCULTA")
FigIC(results_ic)
## End(Not run)
```

FigPowerCULTAEst 31

FigPowerCULTAEst

Plot Statistical Power (CULTA Estimates)

### **Description**

Plot statistical power for CULTA estimates.

# Usage

```
FigPowerCULTAEst(results_culta_est)
```

### **Arguments**

results\_culta\_est

Summary CULTA estimates results data frame.

#### **Details**

The parameters are indexed as follows:

- 1  $\phi_0$  parameter. Autoregressive coefficient for profile 0.
- **2**  $\phi_1$  parameter. Autoregressive coefficient for profile 1.
- 3  $\psi_T$  parameter. Variance in the common trait; reflects stable between-person differences.
- 4  $\lambda_{t2}$  parameter. Factor loading for the common trait and item 2.
- 5  $\lambda_{t3}$  parameter. Factor loading for the common trait and item 3.
- **6**  $\lambda_{t4}$  parameter. Factor loading for the common trait and item 4.
- 7  $\psi_{p11}$  parameter. Trait-specific item 1 variance.
- 8  $\psi_{p22}$  parameter. Trait-specific item 2 variance.
- 9  $\psi_{p33}$  parameter. Trait-specific item 3 variance.
- 10  $\psi_{p44}$  parameter. Trait-specific item 4 variance.
- 11  $\psi_{s0}$  parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.
- 12  $\psi_s$  parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.
- 13  $\lambda_{s2}$  parameter. Factor loading for the common state and item 2.
- 14  $\lambda_{s3}$  parameter. Factor loading for the common state and item 3.
- 15  $\lambda_{s4}$  parameter. Factor loading for the common state and item 4.
- **16**  $\theta_{11}$  parameter. Unique state variance for item 1.
- 17  $\theta_{22}$  parameter. Unique state variance for item 2.
- **18**  $\theta_{33}$  parameter. Unique state variance for item 3.
- **19**  $\theta_{44}$  parameter. Unique state variance for item 4.
- **20**  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.

32 FigPowerLTAEst

- 21  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 22  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- 23  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 24  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 25  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- **26**  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 27  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- **28**  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.
- **29**  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.
- **30**  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.
- 31  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.
- **32**  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.
- 33  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigEntropy(), FigIC(), FigPowerLTAEst(), FigRMSECULTAEst(), FigRMSELTAEst()
```

### **Examples**

```
## Not run:
data(results_culta_est, package = "manCULTA")
FigPowerCULTAEst(results_culta_est)
## End(Not run)
```

FigPowerLTAEst

Plot Statistical Power (LTA Estimates)

#### **Description**

Plot statistical power for common LTA estimates.

### Usage

```
FigPowerLTAEst(results_lta_est)
```

FigPowerLTAEst 33

# **Arguments**

```
results_lta_est
```

Summary LTA estimates results data frame.

#### **Details**

The parameters are indexed as follows:

- 1  $\theta_{11}$  parameter. Unique state variance for item 1.
- 2  $\theta_{22}$  parameter. Unique state variance for item 2.
- 3  $\theta_{33}$  parameter. Unique state variance for item 3.
- **4**  $\theta_{44}$  parameter. Unique state variance for item 4.
- 5  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- **6**  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 7  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- **8**  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- **9**  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 10  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- 11  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 12  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- 13  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.
- 14  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.
- 15  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.
- 16  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.
- 17  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.
- 18  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigEntropy(), FigIC(), FigPowerCULTAEst(), FigRMSECULTAEst(), FigRMSELTAEst()
```

```
## Not run:
data(results_lta_est, package = "manCULTA")
FigPowerLTAEst(results_lta_est)
## End(Not run)
```

34 FigRMSECULTAEst

FigRMSECULTAEst

Plot Root Mean Square Error (CULTA Estimates)

### **Description**

Plot root mean square error for CULTA estimates.

#### Usage

```
FigRMSECULTAEst(results_culta_est)
```

#### **Arguments**

results\_culta\_est

Summary CULTA estimates results data frame.

#### **Details**

The parameters are indexed as follows:

- 1  $\phi_0$  parameter. Autoregressive coefficient for profile 0.
- **2**  $\phi_1$  parameter. Autoregressive coefficient for profile 1.
- 3  $\psi_T$  parameter. Variance in the common trait; reflects stable between-person differences.
- 4  $\lambda_{t2}$  parameter. Factor loading for the common trait and item 2.
- 5  $\lambda_{t3}$  parameter. Factor loading for the common trait and item 3.
- **6**  $\lambda_{t4}$  parameter. Factor loading for the common trait and item 4.
- 7  $\psi_{p11}$  parameter. Trait-specific item 1 variance.
- **8**  $\psi_{p22}$  parameter. Trait-specific item 2 variance.
- 9  $\psi_{p33}$  parameter. Trait-specific item 3 variance.
- 10  $\psi_{p44}$  parameter. Trait-specific item 4 variance.
- 11  $\psi_{s0}$  parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.
- 12  $\psi_s$  parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.
- 13  $\lambda_{s2}$  parameter. Factor loading for the common state and item 2.
- 14  $\lambda_{s3}$  parameter. Factor loading for the common state and item 3.
- 15  $\lambda_{s4}$  parameter. Factor loading for the common state and item 4.
- **16**  $\theta_{11}$  parameter. Unique state variance for item 1.
- 17  $\theta_{22}$  parameter. Unique state variance for item 2.
- **18**  $\theta_{33}$  parameter. Unique state variance for item 3.
- 19  $\theta_{44}$  parameter. Unique state variance for item 4.
- **20**  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.

FigRMSELTAEst 35

- 21  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 22  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- 23  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 24  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 25  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- **26**  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 27  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- **28**  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.
- **29**  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.
- **30**  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.
- **31**  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.
- **32**  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.
- 33  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigEntropy(), FigIC(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSELTAEst()
```

### **Examples**

```
## Not run:
data(results_culta_est, package = "manCULTA")
FigRMSECULTAEst(results_culta_est)
## End(Not run)
```

FigRMSELTAEst

Plot Root Mean Square Error (LTA Estimates)

#### **Description**

Plot root mean square error for common LTA estimates.

### Usage

```
FigRMSELTAEst(results_lta_est)
```

36 FigRMSELTAEst

### **Arguments**

```
results_lta_est
```

Summary LTA estimates results data frame.

#### **Details**

The parameters are indexed as follows:

- 1  $\theta_{11}$  parameter. Unique state variance for item 1.
- 2  $\theta_{22}$  parameter. Unique state variance for item 2.
- 3  $\theta_{33}$  parameter. Unique state variance for item 3.
- **4**  $\theta_{44}$  parameter. Unique state variance for item 4.
- 5  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- **6**  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 7  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.
- **8**  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 9  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 10  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- 11  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.
- 12  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.
- 13  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.
- 14  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.
- 15  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.
- 16  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.
- 17  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.
- 18  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

#### Author(s)

Ivan Jacob Agaloos Pesigan

# See Also

```
Other Figure Functions: FigBiasCULTAEst(), FigBiasLTAEst(), FigCoverageCULTAEst(), FigCoverageLTAEst(), FigEntropy(), FigIC(), FigPowerCULTAEst(), FigPowerLTAEst(), FigRMSECULTAEst()
```

```
## Not run:
data(results_lta_est, package = "manCULTA")
FigRMSELTAEst(results_lta_est)
## End(Not run)
```

FitCULTA1Profile 37

FitCULTA1Profile

Fit the One-Profile CULTA Model (CUTS Model with AR)

### **Description**

Fits the one-profile CULTA model using Mplus.

## Usage

```
FitCULTA1Profile(
  data,
  wd = ".",
  mplus_bin = NULL,
  starts = 10,
  starting_values = NULL
)
```

# **Arguments**

data R object. Object of class simculta.
wd Character string. Working directory.

mplus\_bin Character string. Path to Mplus binary. If mplus\_bin = NULL, the function will

try to find the appropriate binary.

starts Positive integer. Number of initial stage starting values.

starting\_values

Optional list of starting values.

### Value

Returns an object of class fitculta. which is a list with the following elements:

- call: Function call.
- fun: Function used ("FitCULTA1Profile").
- args: Function arguments.
- output: Mplus output files.
- elapsed: Elapsed time.

### Author(s)

Ivan Jacob Agaloos Pesigan

### See Also

Other Model Fitting Functions: FitCULTA2Profiles(), FitLTA2Profiles(), FitRILTA2Profiles(), InputCULTA1Profile(), InputCULTA2Profiles(), InputLTA2Profiles(), InputRILTA2Profiles()

38 FitCULTA1Profile

```
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q \leftarrow 1 \# common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
)
```

FitCULTA2Profiles 39

```
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
# model fitting ------
# NOTE: Model fitting takes time
FitCULTA1Profile(data = data)
## End(Not run)
```

FitCULTA2Profiles

Fit the Two-Profile CULTA Model

# Description

Fits the two-profile CULTA model using Mplus.

40 FitCULTA2Profiles

### Usage

```
FitCULTA2Profiles(
  data,
  wd = ".",
  ncores = 1L,
  mplus_bin = NULL,
  starts = c(20, 4),
  stiterations = 10,
  stscale = 5,
  starting_values = NULL
)
```

### **Arguments**

data R object. Object of class simculta. Character string. Working directory. wd Positive integer. Number of cores to use. ncores Character string. Path to Mplus binary. If mplus\_bin = NULL, the function will mplus\_bin try to find the appropriate binary. starts Vector of positive integer of length two. Number of initial stage starts and number of final stage optimizations. stiterations Positive integer. Number of initial stage iterations. stscale Positive integer. Random start scale. starting\_values

### Value

Returns an object of class fitculta. which is a list with the following elements:

Optional list of starting values.

- call: Function call.
- fun: Function used ("FitCULTA2Profiles").
- args: Function arguments.
- output: Mplus output files.
- elapsed: Elapsed time.

#### Author(s)

Ivan Jacob Agaloos Pesigan

# See Also

```
Other Model Fitting Functions: FitCULTA1Profile(), FitLTA2Profiles(), FitRILTA2Profiles(), InputCULTA1Profile(), InputCULTA2Profiles(), InputLTA2Profiles(), InputRILTA2Profiles()
```

FitCULTA2Profiles 41

# Examples

)

```
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q \leftarrow 1 \# common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
```

42 FitLTA2Profiles

```
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
# model fitting ------
# NOTE: Model fitting takes time
FitCULTA2Profiles(data = data)
## End(Not run)
```

FitLTA2Profiles

Fit the Two-Profile LTA Model

### **Description**

Fits the two-profile LTA model using Mplus.

FitLTA2Profiles 43

### Usage

```
FitLTA2Profiles(
  data,
  wd = ".",
  ncores = 1L,
  mplus_bin = NULL,
  starts = c(20, 4),
  stiterations = 10,
  stscale = 5,
  starting_values = NULL
)
```

### **Arguments**

data R object. Object of class simculta. Character string. Working directory. wd Positive integer. Number of cores to use. ncores Character string. Path to Mplus binary. If mplus\_bin = NULL, the function will mplus\_bin try to find the appropriate binary. starts Vector of positive integer of length two. Number of initial stage starts and number of final stage optimizations. stiterations Positive integer. Number of initial stage iterations. stscale Positive integer. Random start scale. starting\_values Optional list of starting values.

### Value

Returns an object of class fitculta. which is a list with the following elements:

- call: Function call.
- fun: Function used ("FitLTA2Profiles").
- args: Function arguments.
- output: Mplus output files.
- elapsed: Elapsed time.

#### Author(s)

Ivan Jacob Agaloos Pesigan

# See Also

```
Other Model Fitting Functions: FitCULTA1Profile(), FitCULTA2Profiles(), FitRILTA2Profiles(), InputCULTA1Profile(), InputCULTA2Profiles(), InputLTA2Profiles(), InputRILTA2Profiles()
```

44 FitLTA2Profiles

```
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q \leftarrow 1 \# common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
)
```

FitRILTA2Profiles 45

```
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
# model fitting ------
# NOTE: Model fitting takes time
FitLTA2Profiles(data = data)
## End(Not run)
```

FitRILTA2Profiles

Fit the Two-Profile RILTA Model

# Description

Fits the two-profile RILTA model using Mplus.

46 FitRILTA2Profiles

### Usage

```
FitRILTA2Profiles(
  data,
  wd = ".",
  ncores = 1L,
  mplus_bin = NULL,
  starts = c(20, 4),
  stiterations = 10,
  stscale = 5,
  starting_values = NULL
)
```

### **Arguments**

data R object. Object of class simculta. Character string. Working directory. wd Positive integer. Number of cores to use. ncores Character string. Path to Mplus binary. If mplus\_bin = NULL, the function will mplus\_bin try to find the appropriate binary. starts Vector of positive integer of length two. Number of initial stage starts and number of final stage optimizations. stiterations Positive integer. Number of initial stage iterations. stscale Positive integer. Random start scale. starting\_values

### Value

Returns an object of class fitculta. which is a list with the following elements:

Optional list of starting values.

- call: Function call.
- fun: Function used ("FitRILTA2Profiles").
- args: Function arguments.
- output: Mplus output files.
- elapsed: Elapsed time.

#### Author(s)

Ivan Jacob Agaloos Pesigan

### See Also

Other Model Fitting Functions: FitCULTA1Profile(), FitCULTA2Profiles(), FitLTA2Profiles(), InputCULTA1Profile(), InputCULTA2Profiles(), InputLTA2Profiles(), InputRILTA2Profiles()

FitRILTA2Profiles 47

## **Examples**

)

```
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q \leftarrow 1 \# common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
```

```
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
# model fitting ------
# NOTE: Model fitting takes time
FitRILTA2Profiles(data = data)
## End(Not run)
```

## **Description**

Generates data from a two-profile longitudinal CULTA model where profile membership, trait components, and state dynamics are influenced by an observed covariate. The function produces simulated responses incorporating covariate effects, individual differences, and time-varying fluctuations.

# Usage

```
GenCULTA2Profiles(
  n,
 mu_x,
  sigma_x,
  nu_0,
  kappa_0,
  alpha_0,
 beta_00,
  gamma_00,
  gamma_10,
 mu_t,
 psi_t,
 mu_p,
  psi_p,
  common_trait_loading,
  common_state_loading,
  phi_0,
  phi_1,
  psi_s0,
  psi_s,
  theta,
 mu_profile
)
```

# Arguments

n	Positive integer. Number of individuals.
m	Positive integer. Number of measurement occasions.
mu_x	Numeric. Mean of the covariate $\mu_X$ .
sigma_x	Numeric. Variance of the covariate $\sigma_X$ .
nu_0	Numeric. Intercept $\nu_0$ for the logistic model of initial profile membership.
kappa_0	Numeric. Covariate effect $\kappa_0$ on initial profile membership.
alpha_0	Numeric. Intercept $\alpha_0$ for the logistic model of profile transitions across time.
beta_00	Numeric. Effect $\beta_{00}$ for self-persistence in profile 0 transitions.
gamma_00	Numeric. Covariate effect $\gamma_{00}$ on remaining in profile 0.
gamma_10	Numeric. Covariate effect $\gamma_{10}$ on transitioning from profile 1 to profile 0.

mu_t	Numeric or vector of length $q$ . Mean $\mu_T$ of the common trait factor. If mu_t = NULL, defaults to zero.	
psi_t	Numeric matrix of size $q \times q$ . Positive definite covariance matrix $\psi_T$ for the common trait factor.	
mu_p	Numeric vector of length $p$ . Mean vector $\pmb{\mu}_p$ for unique trait components. If mu_p = NULL, defaults to zero.	
psi_p	Numeric matrix of size $p \times p$ . Positive definite covariance matrix $\Psi_p$ for unique trait components.	
common_trait_loading		
	Numeric matrix of size $p \times q$ . Factor loading matrix specifying the influence of the common trait on each observed item.	
common_state_loading		
	Numeric matrix of size $p \times 1$ . Factor loading matrix specifying the influence of the common state on each observed item.	
phi_0	Numeric. Autoregressive coefficient $\phi_0$ for the common state process in profile 0.	
phi_1	Numeric. Autoregressive coefficient $\phi_1$ for the common state process in profile 0.	
psi_s0	Numeric. Variance $\psi_{s0}$ of the initial common state.	
psi_s	Numeric. Innovation variance $\psi_s$ for the common state process.	
theta	Numeric matrix of size $p \times p$ . Positive definite covariance matrix $\Theta$ for unique state components.	
mu_profile	Numeric matrix of size $p \times 2$ . Profile-specific means for each observed item across two latent profiles.	

### **Details**

The GenCULTA2Profiles() function generates data for a two-profile CULTA model with a covariate. The CULTA model incorporates a covariate, latent categorical variables, trait components, state components, and profile-specific means to simulate longitudinal data with latent profile transitions.

Let  $i \in \{1,\ldots,n\}$  denote the index for individuals, let  $t \in \{0,\ldots,m-1\}$  denote the index measurement occasions, let  $k \in \{1, \dots, p\}$  denote the index items, and let  $c \in \{0, 1\}$  be the index of the two latent profiles (profile 0 and profile 1). Let q be the trait dimension, q = 1 in this context.

### Covariate

The covariate is generated from a normal distribution with mean  $\mu_X$  and variance  $\sigma_X$ .

### **Latent Categorical Variables**

Latent categorical variables represent profile membership for each individual at each measurement occasion. In a two-profile model, profile membership is influenced by a covariate and previous profile status, following a logistic formulation. We distinguish between:

- Initial profile membership (baseline time point)
- Profile transitions across subsequent time points

We describe both components below.

### **Initial Profile Membership**

For the first measurement occasion (t = 0), profile membership is determined by the following log-odds for belonging to profile 0 (with profile 1 as the reference category):

( 
$$\nu_0 + \kappa_0 \times \text{Covariate} \quad 0$$
 ).

The corresponding probability of belonging to each profile is given by:

$$\left(\begin{array}{c} \frac{\exp(\nu_0 + \kappa_0 \times \text{Covariate})}{\exp(\nu_0 + \kappa_0 \times \text{Covariate}) + 1} & \frac{1}{\exp(\nu_0 + \kappa_0 \times \text{Covariate}) + 1} \end{array}\right).$$

Profile membership at the first occasion is sampled based on these probabilities.

#### **Profile Transitions**

For subsequent occasions (t = 1, ..., m-1), profile transitions depend on the profile at the previous occasion and the covariate. The log-odds for transitioning to profile 0 at time t are given by:

$$\begin{pmatrix} \alpha_0 + \beta_{00} + \gamma_{00} \times \text{Covariate} & 0 \\ \alpha_0 + \gamma_{10} \times \text{Covariate} & 0 \end{pmatrix}.$$

The probability of transitioning to each profile is computed as:

$$\left( \begin{array}{c} \frac{\exp(\alpha_0 + \beta_{00} + \gamma_{00} \times \text{Covariate})}{\exp(\alpha_0 + \beta_{00} + \gamma_{00} \times \text{Covariate}) + 1} & \frac{1}{\exp(\alpha_0 + \beta_{00} + \gamma_{00} \times \text{Covariate}) + 1} \\ \frac{\exp(\alpha_0 + \gamma_{10} \times \text{Covariate})}{\exp(\alpha_0 + \gamma_{10} \times \text{Covariate}) + 1} & \frac{1}{\exp(\alpha_0 + \gamma_{10} \times \text{Covariate}) + 1} \end{array} \right).$$

Profile membership for each subsequent time point is sampled using these transition probabilities, based on the individual's covariate value and previous profile.

#### **Trait Components**

The trait variate captures between-person differences and is composed of a shared (common) component and item-specific (unique) components. The full decomposition is given by:

$$Trait_i = Common Trait Loading \times Common Trait_i + Unique Trait_i$$
.

We describe each component below.

### **Common Trait**

The common trait Common  $\operatorname{Trait}_i$  represents shared individual differences that influence all items uniformly. It is drawn from a normal distribution with mean  $\mu_T$  and variance  $\psi_T$ :

Common Trait<sub>i</sub> 
$$\sim \mathcal{N}\left(\mu_T, \psi_T\right)$$

The influence of the common trait on each item is determined by the  $p \times q$  common trait loading,

## **Unique Traits**

The unique trait component  $\operatorname{Unique} \operatorname{Trait}_{k,i}$  captures item-specific stable differences and is drawn from a multivariate normal distribution:

Unique Trait<sub>i</sub> 
$$\sim \mathcal{N}\left(\boldsymbol{\mu}_{p}, \boldsymbol{\Psi}_{p \times p}\right)$$

## **Combined Trait Variate**

The trait variate for item k and individual i is obtained by combining the common and unique trait components:

$$\operatorname{Trait}_{k,i} = \operatorname{Common} \operatorname{Trait} \operatorname{Loading}_k \times \operatorname{Common} \operatorname{Trait}_i + \operatorname{Unique} \operatorname{Trait}_{k,i}$$

The common trait component introduces shared variance across items, while the unique trait component allows for item-specific differences not explained by the common trait.

### **State Components**

The state variate is composed of two parts: a common state shared across items, and unique states specific to each item. The full decomposition is given by:

$$State_{k,i,t} = Common\ State\ Loading_k \times Common\ State_{i,t} + Unique\ State_{k,i,t}$$

We describe each component below.

### **Common State**

The common state Common  $State_{i,t}$  evolves over time following a first-order autoregressive process:

Common State<sub>i,t</sub> = 
$$\phi_c \times \text{Common State}_{i,t-1} + \zeta_{i,t}$$
.

The initial common state is drawn from a normal distribution:

Common State<sub>i,0</sub> 
$$\sim \mathcal{N}(0, \psi_{s_0})$$
.

The innovation term  $\zeta_{i,t}$  is normally distributed:

$$\zeta_{i,t} \sim \mathcal{N}\left(0, \psi_s\right)$$
.

The autoregressive parameter  $\phi_c$  depends on latent profile membership c:

$$\phi_c = \phi_0 + (\phi_1 - \phi_0) c.$$

Here,  $\phi_0$  and  $\phi_1$  represent the autoregressive coefficients for profiles coded as 0 and 1, respectively.

### **Unique State**

The unique  $\mathrm{State}_{k,i,t}$  captures item-specific deviations and is drawn from a multivariate normal distribution:

Unique State<sub>i,t</sub> 
$$\sim \mathcal{N}\left(0,\boldsymbol{\theta}\right)$$

where  $\theta$  is the item-level covariance matrix for the unique state component.

### **Combined State Variate**

The state variate for item k, individual i, and time t combines the common and unique state components:

$$\text{State}_{k,i,t} = \text{Common State Loading}_k \times \text{Common State}_{i,t} + \text{Unique State}_{k,i,t}$$

The common state loading parameter Common State Loading $_k$  controls the influence of the shared state on each item.

### Observed Variables

The observed variable is given by

$$Y_{k,i,t} = \mu_{k,c} + \text{Trait}_{k,i} + \text{State}_{k,i,t}$$

where  $\mu_{k,c}$  is the profile specific mean, while  $\operatorname{Trait}_{k,i}$  and  $\operatorname{State}_{k,i,t}$  correspond to the trait and state components of the model.

### Value

Returns an object of class simculta. which is a list with the following elements:

- call: Function call.
- fun: Function used ("GenCULTA2Profiles").
- args: Function arguments.
- id: Vector of ID numbers.
- covariate: Vector of covariate values.
- categorical: Latent profiles.
- common\_trait: Common trait.
- unique\_trait: Unique trait.
- common\_state: Common state.
- trait: Common trait + unique trait.
- state: Common state + unique state.
- data: Generated data which is a matrix of observed variables generated from the CULTA model with two-profiles.

### Author(s)

Ivan Jacob Agaloos Pesigan

### See Also

Other Data Generation Functions: GenData()

```
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 10 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
```

```
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- diag(1)</pre>
mu_t <- 0
psi_p <- diag(p)</pre>
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = q
# state parameters
common_state_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 0.151
psi_s <- 0.290
theta <- diag(p)</pre>
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
```

GenData 55

```
psi_s = psi_s,
theta = theta,
mu_profile = mu_profile
)
```

GenData

Simulate Data

# Description

The function simulates data using the GenCULTA2Profiles() function.

## Usage

```
GenData(taskid)
```

## **Arguments**

taskid

Positive integer. Task ID.

### See Also

Other Data Generation Functions: GenCULTA2Profiles()

# **Examples**

```
## Not run:
set.seed(42)
sim <- GenData(taskid = 1)
as.matrix(sim)
## End(Not run)</pre>
```

 ${\tt Input CULTA1Profile}$ 

Generate Mplus Inout file for the One-Profile CULTA Model (CUTS Model with AR)

# Description

Generates Mplus input file for the one-profile CULTA model.

## Usage

```
InputCULTA1Profile(data, wd = ".", starts = 10, starting_values = NULL)
```

56 InputCULTA1Profile

### **Arguments**

data R object. Object of class simculta.
wd Character string. Working directory.

starts Positive integer. Number of initial stage starting values.

starting\_values

Optional list of starting values.

#### Value

Writes data and input files in wd.

### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

Other Model Fitting Functions: FitCULTA1Profile(), FitCULTA2Profiles(), FitLTA2Profiles(), FitRILTA2Profiles(), InputCULTA2Profiles(), InputLTA2Profiles(), InputRILTA2Profiles()

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x < - 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
```

InputCULTA1Profile

57

```
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
 psi_p_2,
 psi_p_3,
 psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
 nrow = p,
 ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
  c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
  n = n,
  m = m,
  mu_x = mu_x
  sigma_x = sigma_x,
  nu_0 = nu_0,
  kappa_0 = kappa_0,
  alpha_0 = alpha_0,
  beta_00 = beta_00,
  gamma_00 = gamma_00,
  gamma_10 = gamma_10,
  mu_t = mu_t
  psi_t = psi_t,
  mu_p = mu_p,
  psi_p = psi_p,
  common_trait_loading = common_trait_loading,
  common_state_loading = common_state_loading,
```

```
phi_0 = phi_0,
phi_1 = phi_1,
psi_s0 = psi_s0,
psi_s = psi_s,
theta = theta,
mu_profile = mu_profile
)

# generate data and Mplus input files fitting ------
InputCULTA1Profile(data = data)

## End(Not run)
```

InputCULTA2Profiles

Generate Mplus Input file for the Two-Profile CULTA Model

### **Description**

Generates Mplus input file for the two-profile CULTA model.

### Usage

```
InputCULTA2Profiles(
  data,
  wd = ".",
  ncores = 1L,
  starts = c(20, 4),
  stiterations = 10,
  stscale = 5,
  starting_values = NULL
)
```

### **Arguments**

data R object. Object of class simculta.

wd Character string. Working directory.

ncores Positive integer. Number of cores to use.

starts Vector of positive integer of length two. Number of initial stage starts and num-

ber of final stage optimizations.

stiterations Positive integer. Number of initial stage iterations.

stscale Positive integer. Random start scale.

starting\_values

Optional list of starting values.

## Value

Writes data and input files in wd.

### Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

Other Model Fitting Functions: FitCULTA1Profile(), FitCULTA2Profiles(), FitLTA2Profiles(), FitRILTA2Profiles(), InputCULTA1Profile(), InputLTA2Profiles(), InputRILTA2Profiles()

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x < - 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
 psi_p_1,
  psi_p_2,
  psi_p_3,
  psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
```

```
ncol = q
# state parameters
common_state_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# generate data and Mplus input files fitting ------
InputCULTA2Profiles(data = data)
## End(Not run)
```

InputLTA2Profiles 61

InputLTA2Profiles

Generate Mplus Input file for the Two-Profile LTA Model

## **Description**

Generates Mplus input file for the two-profile LTA model.

### Usage

```
InputLTA2Profiles(
  data,
  wd = ".",
  ncores = 1L,
  starts = c(20, 4),
  stiterations = 10,
  stscale = 5,
  starting_values = NULL
)
```

### **Arguments**

data R object. Object of class simculta.

wd Character string. Working directory.

ncores Positive integer. Number of cores to use.

starts Vector of positive integer of length two. Number of initial stage starts and num-

ber of final stage optimizations.

stiterations Positive integer. Number of initial stage iterations.

stscale Positive integer. Random start scale.

starting\_values

Optional list of starting values.

### Value

Writes data and input files in wd.

### Author(s)

Ivan Jacob Agaloos Pesigan

### See Also

```
Other Model Fitting Functions: FitCULTA1Profile(), FitCULTA2Profiles(), FitLTA2Profiles(), FitRILTA2Profiles(), InputCULTA1Profile(), InputCULTA2Profiles(), InputRILTA2Profiles()
```

62 InputLTA2Profiles

```
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q \leftarrow 1 \# common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
)
```

InputRILTA2Profiles 63

```
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
# generate data and Mplus input files fitting ------
InputLTA2Profiles(data = data)
## End(Not run)
```

InputRILTA2Profiles Generate Mplus Input file for the Two-Profile RILTA Model

## **Description**

Generates Mplus input file for the two-profile RILTA model.

InputRILTA2Profiles

### Usage

64

```
InputRILTA2Profiles(
  data,
  wd = ".",
  ncores = 1L,
  starts = c(20, 4),
  stiterations = 10,
  stscale = 5,
  starting_values = NULL
)
```

# Arguments

data R object. Object of class simculta.

wd Character string. Working directory.

ncores Positive integer. Number of cores to use.

starts Vector of positive integer of length two. Number of initial stage starts and num-

ber of final stage optimizations.

stiterations Positive integer. Number of initial stage iterations.

stscale Positive integer. Random start scale.

starting\_values

Optional list of starting values.

### Value

Writes data and input files in wd.

## Author(s)

Ivan Jacob Agaloos Pesigan

#### See Also

```
Other Model Fitting Functions: FitCULTA1Profile(), FitCULTA2Profiles(), FitLTA2Profiles(), FitRILTA2Profiles(), InputCULTA1Profiles(), InputCULTA2Profiles()
```

```
## Not run:
# complete list of R function arguments ------
# random seed for reproducibility
set.seed(42)

# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items</pre>
```

InputRILTA2Profiles 65

```
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
  psi_p_2,
  psi_p_3,
 psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = q
)
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
  nrow = p,
 ncol = 1
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
  c(-0.278, -0.165, -0.199, -0.148)
```

logLik.fitculta

```
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# generate data and Mplus input files fitting ------
InputRILTA2Profiles(data = data)
## End(Not run)
```

logLik.fitculta

Extract Log-Likelihood

## **Description**

Extract Log-Likelihood

### Usage

```
## S3 method for class 'fitculta'
logLik(object, ...)
```

### **Arguments**

object Object of class fitculta.
... additional arguments.

logLik.fitculta 67

## Value

Returns an object of class logLik. This is a number with at the attribute, "df" (degrees of freedom), giving the number of (estimated) parameters in the model, and "correction" which is the scaling correction factor for MLR.

## Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x < - 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
  psi_p_2,
  psi_p_3,
  psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
```

68 logLik.fitculta

```
nrow = p,
 ncol = q
)
# state parameters
common_state_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
logLik(fit, level = 0.95)
```

params 69

## End(Not run)

params

Simulation Parameters

## **Description**

**Simulation Parameters** 

### Usage

data(params)

#### **Format**

A dataframe with 5 rows and 38 columns:

taskid Simulation Task ID.

**n** Sample size.

m Measurement occasions.

**mu\_x**  $\mu_x$  parameter. Mean of the covariate.

**sigma\_x**  $\sigma_x$  parameter. Variance of the covariate.

**mu\_10**  $\mu_{10}$  parameter. Profile specific mean for profile 0 and item 1.

**mu\_20**  $\mu_{20}$  parameter. Profile specific mean for profile 0 and item 2.

**mu\_30**  $\mu_{30}$  parameter. Profile specific mean for profile 0 and item 3.

**mu\_40**  $\mu_{40}$  parameter. Profile specific mean for profile 0 and item 4.

**lambda\_t2**  $\lambda_{t2}$  parameter. Factor loading for the common trait and item 2.

**lambda\_s2**  $\lambda_{s2}$  parameter. Factor loading for the common state and item 2.

**lambda\_t3**  $\lambda_{t3}$  parameter. Factor loading for the common trait and item 3.

**lambda\_s3**  $\lambda_{s3}$  parameter. Factor loading for the common state and item 3.

**lambda\_t4**  $\lambda_{t4}$  parameter. Factor loading for the common trait and item 4.

**lambda\_s4**  $\lambda_{s4}$  parameter. Factor loading for the common state and item 4.

**theta\_11**  $\theta_{11}$  parameter. Unique state variance for item 1.

**theta\_22**  $\theta_{22}$  parameter. Unique state variance for item 2.

**theta\_33**  $\theta_{33}$  parameter. Unique state variance for item 3.

**theta\_44**  $\theta_{44}$  parameter. Unique state variance for item 4.

**phi\_0**  $\phi_0$  parameter. Autoregressive coefficient for profile 0.

**psi\_t**  $\psi_T$  parameter. Variance in the common trait; reflects stable between-person differences.

**psi\_p\_11**  $\psi_{p11}$  parameter. Trait-specific item 1 variance.

**psi\_p\_22**  $\psi_{p22}$  parameter. Trait-specific item 2 variance.

70 print.fitculta

```
psi_p_33 \psi_{p33} parameter. Trait-specific item 3 variance.
```

**psi\_p\_44**  $\psi_{p44}$  parameter. Trait-specific item 4 variance.

**psi\_s0**  $\psi_{s0}$  parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.

**psi\_s**  $\psi_s$  parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.

**mu\_11**  $\mu_{11}$  parameter. Profile specific mean for profile 1 and item 1.

**mu\_21**  $\mu_{21}$  parameter. Profile specific mean for profile 1 and item 2.

**mu\_31**  $\mu_{31}$  parameter. Profile specific mean for profile 1 and item 3.

**mu\_41**  $\mu_{41}$  parameter. Profile specific mean for profile 1 and item 4.

**phi\_1**  $\phi_1$  parameter. Autoregressive coefficient for profile 1.

**nu\_0**  $\nu_0$  parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.

**alpha\_0**  $\alpha_0$  parameter. Baseline log-odds of being in profile 0 across days.

**kappa\_0**  $\kappa_0$  parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.

**beta\_00**  $\beta_{00}$  parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.

**gamma\_00**  $\gamma_{00}$  parameter. Covariate effect on staying in profile 0; higher X increases persistence.

**gamma\_10**  $\gamma_{10}$  parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.

## Author(s)

Ivan Jacob Agaloos Pesigan

print.fitculta

Print Method for an Object of Class fitculta

### **Description**

Print Method for an Object of Class fitculta

### Usage

```
## S3 method for class 'fitculta'
print(x, alpha = NULL, digits = 4, ...)
```

#### **Arguments**

```
x Object of class fitculta. alpha Numeric vector. Significance level \alpha. If alpha = NULL, use alpha = 0.05. digits Digits to print. additional arguments.
```

print.fitculta 71

## Value

Prints a matrix of standardized regression slopes, standard errors, test statistics, p-values, and confidence intervals.

#### Author(s)

Ivan Jacob Agaloos Pesigan

```
## Not run:
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x < - 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
  psi_p_2,
  psi_p_3,
  psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
```

72 print.fitculta

```
ncol = q
# state parameters
common_state_loading <- matrix(</pre>
 data = 1,
 nrow = p,
 ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
print(fit)
## End(Not run)
```

print.simculta 73

print.simculta

Print Method for an Object of Class simculta

# **Description**

Print Method for an Object of Class simculta

# Usage

```
## S3 method for class 'simculta' print(x, ...)
```

# **Arguments**

x Object of class simculta.

... Additional arguments.

## Author(s)

Ivan Jacob Agaloos Pesigan

# **Examples**

```
x <- GenCULTA2Profiles(</pre>
  n = 10,
  m = 6,
  common_trait_loading = matrix(
    data = c(1, 1.25, 1.50, 1.75),
    ncol = 1
  ),
  common_state_loading = matrix(
    data = c(1, 1.5, 1.75, 2.00),
    ncol = 1
  ),
  mu_t = NULL
  psi_t = NULL,
  mu_p = NULL,
  psi_p = NULL,
  theta = diag(4),
  mu_profile = cbind(
   c(-3, -3, -3, -3),
    c(3, 3, 3, 3)
  ),
  mu_x = 0,
  sigma_x = 1,
  nu_0 = -3.563,
  kappa_0 = 0.122,
```

74 results\_culta\_est

```
alpha_0 = -3.586,
beta_00 = 2.250,
gamma_00 = 0.063,
gamma_10 = 0.094,
phi_0 = 0.311,
phi_1 = 0,
psi_s0 = 0.151,
psi_s = 0.290
)
print(x)
```

results\_culta\_est

Simulation Results (CULTA Estimates)

#### **Description**

Simulation Results (CULTA Estimates)

# Usage

```
data(results_culta_est)
```

#### **Format**

A dataframe with 165 rows and 22 columns:

taskid Task ID.

replications Number of replications.

parnames Parameter names

parameter Population parameter value.

model Fitted model.

- n Sample size.
- est Mean parameter estimate.
- se Mean standard error.
- **z** Mean z statistic.
- **p** Mean p-value.
- **Il** Mean lower limit of the 95% confidence interval.
- ul Mean upper limit of the 95% confidence interval.
- sig Proportion of statistically significant results.

**theta\_hit** Proportion of replications where the confidence intervals contained the population parameter.

sq\_error Mean squared error.

bias Bias.

results\_entropy 75

```
rel_bias Relative bias.
```

se\_bias Bias in standard error estimate.

rel\_se\_bias Relative bias in standard error estimate.

rmse Root mean square error.

coverage Coverage probability.

power Statistical power.

# Author(s)

Ivan Jacob Agaloos Pesigan

results\_entropy

Simulation Results (Entropy)

# Description

Simulation Results (Entropy)

# Usage

```
data(results_entropy)
```

# **Format**

A dataframe with 14 rows and 4 columns:

TaskID Simulation Task ID.

N Sample size.

Model Fitted model.

Entropy Mean entropy.

#### Author(s)

76 results\_lta\_est

 $results\_ic$ 

Simulation Results (Information Criteria)

# Description

Simulation Results (Information Criteria)

# Usage

```
data(results_ic)
```

# **Format**

A dataframe with 60 rows and 5 columns:

TaskID Simulation Task ID.

N Sample size.

Model Fitted model.

IC Information criteria type.

Value Mean information criteria.

# Author(s)

Ivan Jacob Agaloos Pesigan

 $results\_lta\_est$ 

Simulation Results (LTA Estimates)

# Description

Simulation Results (LTA Estimates)

# Usage

```
data(results_lta_est)
```

results\_lta\_est 77

# **Format**

A dataframe with 270 rows and 22 columns:

taskid Task ID.

replications Number of replications.

parnames Parameter names

parameter Population parameter value.

model Fitted model.

n Sample size.

est Mean parameter estimate.

se Mean standard error.

**z** Mean z statistic.

**p** Mean p-value.

**ll** Mean lower limit of the 95% confidence interval.

ul Mean upper limit of the 95% confidence interval.

sig Proportion of statistically significant results.

**theta\_hit** Proportion of replications where the confidence intervals contained the population parameter.

**sq\_error** Mean squared error.

bias Bias.

rel\_bias Relative bias.

se\_bias Bias in standard error estimate.

rel\_se\_bias Relative bias in standard error estimate.

rmse Root mean square error.

coverage Coverage probability.

power Statistical power.

## Author(s)

78 Sim

Sim Simulation Replication

# Description

Simulation Replication

# Usage

```
Sim(
  taskid,
  repid,
  output_folder,
  overwrite,
  integrity,
  seed,
  mplus_bin,
  starts,
  stiterations,
  stscale,
  max_iter
)
```

# Arguments

taskid	Positive integer. Task ID.
repid	Positive integer. Replication ID.
output_folder	Character string. Output folder.
overwrite	Logical. Overwrite existing output in output_folder.
integrity	Logical. If integrity = TRUE, check for the output file integrity when overwrite = FALSE.
seed	Integer. Random seed.
mplus_bin	Character string. Path to Mplus binary. If mplus_bin = NULL, the function will try to find the appropriate binary.
starts	Vector of positive integer of length two. Number of initial stage starts and number of final stage optimizations.
stiterations	Positive integer. Number of initial stage iterations.
stscale	Positive integer. Random start scale.
max_iter	Positive integer. Maximum number of reruns.

# Value

The output is saved as an external file in output\_folder.

SimFitCULTA1Profile 79

#### Author(s)

Ivan Jacob Agaloos Pesigan

SimFitCULTA1Profile Simulation Replication - FitCULTA1Profile

#### **Description**

Simulation Replication - FitCULTA1Profile

# Usage

```
SimFitCULTA1Profile(
  taskid,
  repid,
  output_folder,
  seed,
  suffix,
  overwrite,
  integrity,
  mplus_bin,
  starts,
  max_iter
)
```

# Arguments

taskid Positive integer. Task ID. repid Positive integer. Replication ID. output\_folder Character string. Output folder. seed Integer. Random seed. Character string. Output of manCULTA:::.SimSuffix(). suffix overwrite Logical. Overwrite existing output in output\_folder. integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite = FALSE. mplus\_bin Character string. Path to Mplus binary. If mplus\_bin = NULL, the function will try to find the appropriate binary. starts Positive integer. Number of initial stage starting values. Positive integer. Maximum number of reruns. max\_iter

#### **Details**

This function is executed via the Sim function.

80 SimFitCULTA2Profiles

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

Ivan Jacob Agaloos Pesigan

 ${\tt SimFitCULTA2Profiles} \quad \textit{Simulation Replication - FitCULTA2Profiles}$ 

# Description

Simulation Replication - FitCULTA2Profiles

#### Usage

```
SimFitCULTA2Profiles(
  taskid,
  repid,
  output_folder,
  seed,
  suffix,
  overwrite,
  integrity,
  mplus_bin,
  starts,
  stiterations,
  stscale,
  max_iter
)
```

#### **Arguments**

taskid Positive integer. Task ID. Positive integer. Replication ID. repid output\_folder Character string. Output folder. seed Integer. Random seed. suffix Character string. Output of manCULTA:::.SimSuffix(). overwrite Logical. Overwrite existing output in output\_folder. Logical. If integrity = TRUE, check for the output file integrity when overwrite integrity = FALSE. mplus\_bin Character string. Path to Mplus binary. If mplus\_bin = NULL, the function will try to find the appropriate binary. Vector of positive integer of length two. Number of initial stage starts and numstarts ber of final stage optimizations.

SimFitLTA2Profiles 81

stiterations Positive integer. Number of initial stage iterations.

stscale Positive integer. Random start scale.

max\_iter Positive integer. Maximum number of reruns.

#### **Details**

This function is executed via the Sim function.

#### Value

The output is saved as an external file in output\_folder.

### Author(s)

Ivan Jacob Agaloos Pesigan

SimFitLTA2Profiles

Simulation Replication - FitLTA2Profiles

# **Description**

Simulation Replication - FitLTA2Profiles

# Usage

```
SimFitLTA2Profiles(
  taskid,
  repid,
  output_folder,
  seed,
  suffix,
  overwrite,
  integrity,
  mplus_bin,
  starts,
  stiterations,
  stscale,
  max_iter
)
```

# **Arguments**

taskid Positive integer. Task ID.

repid Positive integer. Replication ID. output\_folder Character string. Output folder.

seed Integer. Random seed.

82 SimFitRILTA2Profiles

suffix	Character string. Output of manCULTA:::.SimSuffix().
overwrite	Logical. Overwrite existing output in output_folder.
integrity	Logical. If integrity = TRUE, check for the output file integrity when overwrite = FALSE.
mplus_bin	Character string. Path to Mplus binary. If mplus_bin = NULL, the function will try to find the appropriate binary.
starts	Vector of positive integer of length two. Number of initial stage starts and number of final stage optimizations.
stiterations	Positive integer. Number of initial stage iterations.
stscale	Positive integer. Random start scale.
max_iter	Positive integer. Maximum number of reruns.

# **Details**

This function is executed via the Sim function.

# Value

The output is saved as an external file in output\_folder.

# Author(s)

Ivan Jacob Agaloos Pesigan

SimFitRILTA2Profiles Simulation Replication - FitRILTA2Profiles

# Description

Simulation Replication - FitRILTA2Profiles

# Usage

```
SimFitRILTA2Profiles(
  taskid,
  repid,
  output_folder,
  seed,
  suffix,
  overwrite,
  integrity,
  mplus_bin,
  starts,
  stiterations,
  stscale,
  max_iter
)
```

SimFN 83

# **Arguments**

taskid Positive integer. Task ID.
repid Positive integer. Replication ID.

output\_folder Character string. Output folder.

seed Integer. Random seed.

suffix Character string. Output of manCULTA:::.SimSuffix().

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

mplus\_bin Character string. Path to Mplus binary. If mplus\_bin = NULL, the function will

try to find the appropriate binary.

starts Vector of positive integer of length two. Number of initial stage starts and num-

ber of final stage optimizations.

stiterations Positive integer. Number of initial stage iterations.

stscale Positive integer. Random start scale.

max\_iter Positive integer. Maximum number of reruns.

#### **Details**

This function is executed via the Sim function.

#### Value

The output is saved as an external file in output\_folder.

# Author(s)

Ivan Jacob Agaloos Pesigan

SimFN	Simulation File Name	,

# Description

Simulation File Name

# Usage

```
SimFN(output_type, output_folder, suffix)
```

#### **Arguments**

output\_type Character string. Output type.
output\_folder Character string. Output folder.

suffix Character string. Output of manCULTA:::.SimSuffix().

84 SimGenData

# Value

Returns a character string file name with the output\_folder in the OS-specific format.

# Description

Simulation Replication - GenData

# Usage

```
SimGenData(taskid, repid, output_folder, seed, suffix, overwrite, integrity)
```

# **Arguments**

taskid Positive integer. Task ID.

repid Positive integer. Replication ID. output\_folder Character string. Output folder.

seed Integer. Random seed.

suffix Character string. Output of manCULTA:::.SimSuffix().

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

#### **Details**

This function is executed via the Sim function.

# Value

The output is saved as an external file in output\_folder.

#### Author(s)

SimProj 85

SimProj Simulation Project Name

# Description

Simulation Project Name

# Usage

SimProj()

#### Value

Returns the project name as a character string.

#### Author(s)

Ivan Jacob Agaloos Pesigan

Sum Summary

# **Description**

Summary

# Usage

```
Sum(taskid, reps, output_folder, overwrite, integrity, ncores = 1L)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# Value

The output is saved as an external file in output\_folder.

# Author(s)

86 SumFitCULTA1Profile

SumFitCULTA1Profile Summary (FitCULTA1Profile)

# **Description**

Summary (FitCULTA1Profile)

# Usage

```
SumFitCULTA1Profile(
  taskid,
  reps,
  output_folder,
  overwrite,
  integrity,
  ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

SumFitCULTA2Profiles 87

```
SumFitCULTA2Profiles Summary (FitCULTA2Profiles)
```

# **Description**

```
Summary (FitCULTA2Profiles)
```

# Usage

```
SumFitCULTA2Profiles(
  taskid,
  reps,
  output_folder,
  overwrite,
  integrity,
  ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

88 SumFitLTA2Profiles

SumFitLTA2Profiles Summary (FitLTA2Profiles)

# **Description**

Summary (FitLTA2Profiles)

# Usage

```
SumFitLTA2Profiles(
  taskid,
  reps,
  output_folder,
  overwrite,
  integrity,
  ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

SumFitRILTA2Profiles 89

SumFitRILTA2Profiles Summary (FitRILTA2Profiles)

# **Description**

Summary (FitRILTA2Profiles)

# Usage

```
SumFitRILTA2Profiles(
  taskid,
  reps,
  output_folder,
  overwrite,
  integrity,
  ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

SumLTAParamsCULTA2Profiles

Summary (LTAParamsCULTA2Profiles)

# Description

Summary (LTAParamsCULTA2Profiles)

# Usage

```
SumLTAParamsCULTA2Profiles(
  taskid,
  reps,
  output_folder,
  overwrite,
  integrity,
  ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

SumLTAParamsLTA2Profiles

Summary (LTAParamsLTA2Profiles)

# Description

Summary (LTAParamsLTA2Profiles)

# Usage

```
SumLTAParamsLTA2Profiles(
  taskid,
  reps,
  output_folder,
  overwrite,
  integrity,
  ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

SumLTAParamsRILTA2Profiles

Summary (LTAParamsRILTA2Profiles)

# Description

Summary (LTAParamsRILTA2Profiles)

# Usage

```
SumLTAParamsRILTA2Profiles(
   taskid,
   reps,
   output_folder,
   overwrite,
   integrity,
   ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

#### Value

The output is saved as an external file in output\_folder.

#### Author(s)

summary.fitculta 93

summary.fitculta Summa

Summary Method for an Object of Class fitculta

# **Description**

Summary Method for an Object of Class fitculta

#### Usage

```
## S3 method for class 'fitculta'
summary(object, alpha = NULL, digits = 4, ...)
```

# **Arguments**

object Object of class fitculta. 
alpha Numeric vector. Significance level  $\alpha$ . If alpha = NULL, use alpha = 0.05. 
digits Digits to print. 
additional arguments.

#### Value

Returns a matrix of standardized regression slopes, standard errors, test statistics, p-values, and confidence intervals.

#### Author(s)

Ivan Jacob Agaloos Pesigan

# **Examples**

```
## Not run:
# complete list of R function arguments ------
# random seed for reproducibility
set.seed(42)

# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q <- 1 # common trait dimension

# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566

# profile membership and transition parameters
nu_0 <- -3.563</pre>
```

94 summary.fitculta

```
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t <- 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
)
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
)
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
  c(2.253, 1.493, 1.574, 1.117),
  c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
  n = n,
  m = m,
  mu_x = mu_x,
  sigma_x = sigma_x,
  nu_0 = nu_0,
  kappa_0 = kappa_0,
```

```
alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
)
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
summary(fit)
## End(Not run)
```

SumParamsCULTA2Profiles

Summary (ParamsCULTA2Profiles)

# **Description**

Summary (ParamsCULTA2Profiles)

# Usage

```
SumParamsCULTA2Profiles(
  taskid,
  reps,
  output_folder,
  overwrite,
  integrity,
  ncores = 1L
)
```

# Arguments

taskid Positive integer. Task ID.

reps Positive integer. Number of replications.

96 vcov.fitculta

output\_folder Character string. Output folder.

overwrite Logical. Overwrite existing output in output\_folder.

integrity Logical. If integrity = TRUE, check for the output file integrity when overwrite

= FALSE.

ncores Positive integer. Number of cores to use.

# **Details**

This function is executed via the Sum function.

# Value

The output is saved as an external file in output\_folder.

#### Author(s)

Ivan Jacob Agaloos Pesigan

vcov.fitculta Sampling Covariance Matrix

# **Description**

Sampling Covariance Matrix

#### Usage

```
## S3 method for class 'fitculta'
vcov(object, ...)
```

# Arguments

object Object of class fitculta.
... additional arguments.

#### Value

Returns a matrix of the variance-covariance matrix of parameter estimates.

#### Author(s)

vcov.fitculta 97

# **Examples**

)

```
# complete list of R function arguments -----
# random seed for reproducibility
set.seed(42)
# dimensions
n <- 1000 # number of individuals
m <- 6 # measurement occasions
p <- 4 # number of items
q \leftarrow 1 \# common trait dimension
# covariate parameters
mu_x <- 11.4009
sigma_x <- 24.67566
# profile membership and transition parameters
nu_0 <- -3.563
kappa_0 <- 0.122
alpha_0 <- -3.586
beta_00 <- 2.250
gamma_00 <- 0.063
gamma_10 <- 0.094
# trait parameters
psi_t < 0.10 * diag(1)
mu_t <- 0
psi_p <- diag(p)</pre>
psi_p_1 <- 0.10
psi_p_2 <- 0.10
psi_p_3 <- 0.50
psi_p_4 <- 0.50
diag(psi_p) <- c(</pre>
  psi_p_1,
 psi_p_2,
 psi_p_3,
  psi_p_4
)
mu_p \leftarrow rep(x = 0, times = p)
common_trait_loading <- matrix(</pre>
  data = 1,
  nrow = p,
  ncol = q
# state parameters
common_state_loading <- matrix(</pre>
  data = 1,
 nrow = p,
  ncol = 1
```

98 WriteData

```
phi_0 <- 0.000
phi_1 <- 0.311
psi_s0 <- 1.00
psi_s <- 0.25
theta <- 0.15 * diag(p)
# profile-specific means
mu_profile <- cbind(</pre>
 c(2.253, 1.493, 1.574, 1.117),
 c(-0.278, -0.165, -0.199, -0.148)
)
# data generation ------
data <- GenCULTA2Profiles(</pre>
 n = n,
 m = m,
 mu_x = mu_x,
 sigma_x = sigma_x,
 nu_0 = nu_0,
 kappa_0 = kappa_0,
 alpha_0 = alpha_0,
 beta_00 = beta_00,
 gamma_00 = gamma_00,
 gamma_10 = gamma_10,
 mu_t = mu_t
 psi_t = psi_t,
 mu_p = mu_p,
 psi_p = psi_p,
 common_trait_loading = common_trait_loading,
 common_state_loading = common_state_loading,
 phi_0 = phi_0,
 phi_1 = phi_1,
 psi_s0 = psi_s0,
 psi_s = psi_s,
 theta = theta,
 mu_profile = mu_profile
# model fitting ------
# NOTE: Model fitting takes time
fit <- FitCULTA2Profiles(data = data)</pre>
vcov(fit)
## End(Not run)
```

WriteData

Write Data to File

#### **Description**

Generic function to write data to file.

WriteData 99

# Usage

```
WriteData(x, file, ...)
## S3 method for class 'simculta'
WriteData(x, file, ...)
```

# Arguments

x Object of class simculta.file Character string. File name.... Additional arguments.

# Value

Invisibly returns NULL. Writes data to file as a side effect.

# Methods (by class)

• WriteData(simculta): Method for objects of class simculta.

# Author(s)

# **Index**

* Compression Functions	results_culta_est,74
Compress, 15	results_entropy,75
* Data Generation Functions	results_ic,76
GenCULTA2Profiles,48	results_lta_est,76
GenData, 55	* figure
* Figure Functions	FigBiasCULTAEst, 23
FigBiasCULTAEst, 23	FigBiasLTAEst, 25
FigBiasLTAEst, 25	FigCoverageCULTAEst, 26
FigCoverageCULTAEst, 26	FigCoverageLTAEst, 28
FigCoverageLTAEst, 28	FigEntropy, 29
FigEntropy, 29	FigIC, 30
FigIC, 30	FigPowerCULTAEst, 31
FigPowerCULTAEst, 31	FigPowerLTAEst, 32
FigPowerLTAEst, 32	FigRMSECULTAEst, 34
FigRMSECULTAEst, 34	FigRMSELTAEst, 35
FigRMSELTAEst, 35	* fit
* Model Fitting Functions	FitCULTA1Profile, 37
FitCULTA1Profile, 37	FitCULTA2Profiles, 39
FitCULTA2Profiles, 39	FitLTA2Profiles, 42
FitLTA2Profiles, 42	FitRILTA2Profiles, 45
FitRILTA2Profiles, 45	SimFitCULTA1Profile, 79
<pre>InputCULTA1Profile, 55</pre>	SimFitCULTA2Profiles, 80
InputCULTA2Profiles, 58	SimFitLTA2Profiles, 81
<pre>InputLTA2Profiles, 61</pre>	SimFitRILTA2Profiles, 82
InputRILTA2Profiles, 63	* gendata
* compress	GenData, 55
Compress, 15	SimGenData, 84
* culta	* input
FitCULTA1Profile, 37	InputCULTA1Profile, 55
FitCULTA2Profiles, 39	InputCULTA2Profiles, 58
FitLTA2Profiles, 42	InputLTA2Profiles, 61
FitRILTA2Profiles, 45	InputRILTA2Profiles, 63
GenCULTA2Profiles, 48	* manCULTA
InputCULTA1Profile, 55	Compress, 15
InputCULTA2Profiles, 58	FigBiasCULTAEst, 23
InputLTA2Profiles, 61	FigBiasLTAEst, 25
InputRILTA2Profiles, 63	FigCoverageCULTAEst, 26
* data	FigCoverageLTAEst, 28
params, 69	FigEntropy, 29

INDEX 101

F:TC 20	
FigIC, 30	* mixture
FigPowert TAFat 33	FitCULTA1Profile, 37 FitCULTA2Profiles, 39
FigPowerLTAEst, 32	•
FigRMSECULTAEst, 34	FitLTA2Profiles, 42
FigRMSELTAEst, 35	FitRILTA2Profiles, 45
FitCULTA1Profile, 37	GenCULTA2Profiles, 48
FitCULTA2Profiles, 39	InputCULTA1Profile, 55
FitLTA2Profiles, 42	InputCULTA2Profiles, 58
FitRILTA2Profiles, 45	InputLTA2Profiles, 61
GenCULTA2Profiles, 48	InputRILTA2Profiles, 63
GenData, 55	* parameters
InputCULTA1Profile, 55	params, $69$
InputCULTA2Profiles, 58	results_culta_est,74
InputLTA2Profiles, 61	results_entropy, 75
InputRILTA2Profiles, 63	results_ic,76
Sim, 78	results_lta_est,76
SimFitCULTA1Profile, 79	* simulation
SimFitCULTA2Profiles, 80	Sim, 78
SimFitLTA2Profiles, 81	SimFitCULTA1Profile, 79
SimFitRILTA2Profiles, 82	SimFitCULTA2Profiles, 80
SimFN, 83	SimFitLTA2Profiles, 81
SimGenData, 84	SimFitRILTA2Profiles, 82
SimProj, 85	SimFN, 83
Sum, 85	SimGenData, 84
SumFitCULTA1Profile, 86	SimProj, 85
SumFitCULTA2Profiles, 87	Sum, 85
SumFitLTA2Profiles,88	SumFitCULTA1Profile, 86
SumFitRILTA2Profiles, 89	SumFitCULTA2Profiles, 87
SumLTAParamsCULTA2Profiles, 90	SumFitLTA2Profiles, 88
SumLTAParamsLTA2Profiles, 91	SumFitRILTA2Profiles, 89
SumLTAParamsRILTA2Profiles, 92	SumLTAParamsCULTA2Profiles, 90
SumParamsCULTA2Profiles, 95	SumLTAParamsLTA2Profiles, 91
* methods	SumLTAParamsRILTA2Profiles, 92
AIC.fitculta, 3	SumParamsCULTA2Profiles, 95
anova.fitculta, 5	* sim
as.data.frame.simculta,8	<del>~</del>
	GenCULTA2Profiles, 48
as.matrix.simculta,9	* state
BIC.fitculta, 10	FitCULTA1Profile, 37
coef.fitculta, 13	FitCULTA2Profiles, 39
confint.fitculta, 16	FitLTA2Profiles, 42
converged, 18	FitRILTA2Profiles, 45
entropy, 21	GenCULTA2Profiles, 48
logLik.fitculta,66	InputCULTA1Profile, 55
print.fitculta, 70	InputCULTA2Profiles, 58
print.simculta,73	InputLTA2Profiles, 61
summary.fitculta,93	InputRILTA2Profiles, 63
vcov.fitculta,96	* summary
WriteData,98	Sum, 85

INDEX

SumFitCULTA1Profile, 86	FigRMSELTAEst, 24, 26, 27, 29, 30, 32, 33, 35,
SumFitCULTA2Profiles, 87	35
SumFitLTA2Profiles, 88	FitCULTA1Profile, 37, 40, 43, 46, 56, 59, 61,
SumFitRILTA2Profiles, 89	64
SumLTAParamsCULTA2Profiles, 90	FitCULTA2Profiles, 37, 39, 43, 46, 56, 59,
SumLTAParamsLTA2Profiles, 91	61, 64
SumLTAParamsRILTA2Profiles, 92	FitLTA2Profiles, 37, 40, 42, 46, 56, 59, 61,
SumParamsCULTA2Profiles, 95	64
* trait	FitRILTA2Profiles, 37, 40, 43, 45, 56, 59,
FitCULTA1Profile, 37	61, 64
FitCULTA2Profiles, 39	7
FitLTA2Profiles, 42	GenCULTA2Profiles, 48, 55
FitRILTA2Profiles, 45	GenCULTA2Profiles(), $50$ , $55$
GenCULTA2Profiles, 48	GenData, <i>53</i> , <i>55</i>
InputCULTA1Profile, 55	
InputCULTA2Profiles, 58	InputCULTA1Profile, 37, 40, 43, 46, 55, 59,
InputLTA2Profiles, 61	61, 64
InputRILTA2Profiles, 63	InputCULTA2Profiles, 37, 40, 43, 46, 56, 58,
inputkiliazri of fles, 03	61, 64
AIC.fitculta, 3	InputLTA2Profiles, 37, 40, 43, 46, 56, 59,
anova.fitculta,5	61, 64
as.data.frame.simculta,8	InputRILTA2Profiles, 37, 40, 43, 46, 56, 59,
as.matrix.simculta, 9	61, 63
as.mati ix.simeuita, y	,
BIC.fitculta, 10	logLik.fitculta,66
coef.fitculta, 13	params, 69
Compress, 15	print.fitculta, 70
confint.fitculta, 16	print.simculta, 73
converged, 18	p. 1
converged, 10	results_culta_est,74
entropy, 20	results_entropy, 75
	results_ic, 76
FigBiasCULTAEst, 23, 26, 27, 29, 30, 32, 33,	results_lta_est, 76
35, 36	,
FigBiasLTAEst, 24, 25, 27, 29, 30, 32, 33, 35,	Sim, 78
36	SimFitCULTA1Profile, 79
FigCoverageCULTAEst, 24, 26, 26, 29, 30, 32,	SimFitCULTA2Profiles, 80
33, 35, 36	SimFitLTA2Profiles, 81
FigCoverageLTAEst, 24, 26, 27, 28, 29, 30,	SimFitRILTA2Profiles, 82
32, 33, 35, 36	SimFN, 83
FigEntropy, 24, 26, 27, 29, 29, 30, 32, 33, 35,	SimGenData, 84
36	SimProj, 85
FigIC, 24, 26, 27, 29, 30, 32, 33, 35, 36	Sum, 85
FigPowerCULTAEst, 24, 26, 27, 29, 30, 31, 33,	SumFitCULTA1Profile, 86
35, 36	SumFitCULTA2Profiles, 87
FigPowerLTAEst, 24, 26, 27, 29, 30, 32, 32,	SumFitLTA2Profiles, 88
35, 36	SumFitRILTA2Profiles, 89
FigRMSECULTAEst, 24, 26, 27, 29, 30, 32, 33,	SumLTAParamsCULTA2Profiles, 90
34, 36	SumLTAParamsLTA2Profiles, 91
5 1, 50	Jame I'll al ampe I'll I Of IICS, /I

INDEX 103

```
SumLTAParamsRILTA2Profiles, 92
summary.fitculta, 93
SumParamsCULTA2Profiles, 95
vcov.fitculta, 96
WriteData, 98
```