Package 'manCULTA'

July 18, 2025

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

Version 0.9.3

```
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 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	
nova.fitculta	5
s.data.frame.simculta	8
s.matrix.simculta	9
BIC.fitculta	10
oef.fitculta	13
Compress	15
onfint.fitculta	16
onverged	
ntropy	
igBiasCULTAEst	
igBiasLTAEst	
igCoverageCULTAEst	
igCoverageLTAEst	
igEntropy	
igIC	
igPowerCULTAEst	
igPowerLTAEst	
itCULTA1Profile	
itCULTA2Profiles	
itLTA2Profiles	
itRILTA2Profiles	
GenCULTA2Profiles	
GenData	
nputCULTA1Profile	
nputCULTA2Profiles	
nputLTA2Profiles	
nputRILTA2Profiles	
ogLik.fitculta	
arams	
rint.fitculta	
rint.simculta	
esults_culta_est	
esults_entropy	
esults_ic	
esults_lta_est	
im	
limFitCULTA1Profile	
SimFitCULTA2Profiles	
limFitLTA2Profiles	
limFitRILTA2Profiles	
limFN	
imGenData	
imProj	
dum	
umFitCULTA1Profile	82

AIC.fitculta 3

	SumFitCULTA2Profiles	82
	SumFitLTA2Profiles	83
	SumFitRILTA2Profiles	84
	SumLTAParamsCULTA2Profiles	85
	SumLTAParamsLTA2Profiles	85
	SumLTAParamsRILTA2Profiles	86
	summary.fitculta	87
	SumParamsCULTA2Profiles	89
	vcov.fitculta	90
	WriteData	92
Index		94

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

4 AIC.fitculta

```
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)
```

anova.fitculta 5

```
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 <- 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 Bias (CULTA Estimates)

Description

Plot 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 ψ_T parameter. Variance in the common trait; reflects stable between-person differences.
- 2 λ_{t2} parameter. Factor loading for the common trait and item 2.
- 3 λ_{t3} parameter. Factor loading for the common trait and item 3.
- 4 λ_{t4} parameter. Factor loading for the common trait and item 4.
- 5 ψ_{p11} parameter. Trait-specific item 1 variance.
- 6 ψ_{p22} parameter. Trait-specific item 2 variance.
- 7 ψ_{p33} parameter. Trait-specific item 3 variance.
- 8 ψ_{p44} parameter. Trait-specific item 4 variance.
- 9 ψ_{s0} parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.

24 FigBiasCULTAEst

10 ψ_s parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.

- 11 λ_{s2} parameter. Factor loading for the common state and item 2.
- 12 λ_{s3} parameter. Factor loading for the common state and item 3.
- 13 λ_{s4} parameter. Factor loading for the common state and item 4.
- **14** θ_{11} parameter. Unique state variance for item 1.
- 15 θ_{22} parameter. Unique state variance for item 2.
- **16** θ_{33} parameter. Unique state variance for item 3.
- 17 θ_{44} parameter. Unique state variance for item 4.
- **18** ν_0 parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- 19 κ_0 parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- **20** α_0 parameter. Baseline log-odds of being in profile 0 across days.
- 21 β_{00} parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 22 γ_{00} parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 23 γ_{10} parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- **24** μ_{10} parameter. Profile specific mean for profile 0 and item 1.
- 25 μ_{20} parameter. Profile specific mean for profile 0 and item 2.
- **26** μ_{30} parameter. Profile specific mean for profile 0 and item 3.
- 27 μ_{40} parameter. Profile specific mean for profile 0 and item 4.
- **28** μ_{11} parameter. Profile specific mean for profile 1 and item 1.
- **29** μ_{21} parameter. Profile specific mean for profile 1 and item 2.
- **30** μ_{31} parameter. Profile specific mean for profile 1 and item 3.
- 31 μ_{41} parameter. Profile specific mean for profile 1 and item 4.
- **32** ϕ_0 parameter. Autoregressive coefficient for profile 0.
- **33** ϕ_1 parameter. Autoregressive coefficient for profile 1.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

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

FigBiasLTAEst 25

Examples

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

FigBiasLTAEst

Plot Bias (LTA Estimates)

Description

Plot 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 θ_{11} parameter. Unique state variance for item 1.
- 2 θ_{22} parameter. Unique state variance for item 2.
- **3** θ_{33} parameter. Unique state variance for item 3.
- **4** θ_{44} parameter. Unique state variance for item 4.
- 5 ν_0 parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- 6 κ_0 parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 7 α_0 parameter. Baseline log-odds of being in profile 0 across days.
- 8 β_{00} parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- **9** γ_{00} parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 10 γ_{10} parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- 11 μ_{10} parameter. Profile specific mean for profile 0 and item 1.
- 12 μ_{20} parameter. Profile specific mean for profile 0 and item 2.
- 13 μ_{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 μ_{11} parameter. Profile specific mean for profile 1 and item 1.
- 16 μ_{21} parameter. Profile specific mean for profile 1 and item 2.
- 17 μ_{31} parameter. Profile specific mean for profile 1 and item 3.
- 18 μ_{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()
```

Examples

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

 ${\tt FigCoverageCULTAEst}$

Plot Coverage Probabilities (CULTA Estimates)

Description

Plot coverage probabilities for common 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 ψ_T parameter. Variance in the common trait; reflects stable between-person differences.
- 2 λ_{t2} parameter. Factor loading for the common trait and item 2.
- 3 λ_{t3} parameter. Factor loading for the common trait and item 3.

- λ_{t4} parameter. Factor loading for the common trait and item 4.
- ψ_{p11} parameter. Trait-specific item 1 variance.
- ψ_{p22} parameter. Trait-specific item 2 variance.
- ψ_{v33} parameter. Trait-specific item 3 variance.
- **8** ψ_{p44} parameter. Trait-specific item 4 variance.
- ψ_{s0} parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.
- ψ_s parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.
- λ_{s2} parameter. Factor loading for the common state and item 2.
- λ_{s3} parameter. Factor loading for the common state and item 3.
- λ_{s4} parameter. Factor loading for the common state and item 4.
- **14** θ_{11} parameter. Unique state variance for item 1.
- θ_{22} parameter. Unique state variance for item 2.
- θ_{33} parameter. Unique state variance for item 3.
- θ_{44} parameter. Unique state variance for item 4.
- **18** ν_0 parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- κ_0 parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- **20** α_0 parameter. Baseline log-odds of being in profile 0 across days.
- β_{00} parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- γ_{00} parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- γ_{10} parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- **24** μ_{10} parameter. Profile specific mean for profile 0 and item 1.
- μ_{20} parameter. Profile specific mean for profile 0 and item 2.
- **26** μ_{30} parameter. Profile specific mean for profile 0 and item 3.
- μ_{40} parameter. Profile specific mean for profile 0 and item 4.
- μ_{11} parameter. Profile specific mean for profile 1 and item 1.
- **29** μ_{21} parameter. Profile specific mean for profile 1 and item 2.
- **30** μ_{31} parameter. Profile specific mean for profile 1 and item 3.
- μ_{41} parameter. Profile specific mean for profile 1 and item 4.
- **32** ϕ_0 parameter. Autoregressive coefficient for profile 0.
- ϕ_1 parameter. Autoregressive coefficient for profile 1.

Author(s)

Ivan Jacob Agaloos Pesigan

28 FigCoverageLTAEst

See Also

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

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 θ_{11} parameter. Unique state variance for item 1.
- **2** θ_{22} parameter. Unique state variance for item 2.
- **3** θ_{33} parameter. Unique state variance for item 3.
- 4 θ_{44} parameter. Unique state variance for item 4.
- 5 ν_0 parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- 6 κ_0 parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 7 α_0 parameter. Baseline log-odds of being in profile 0 across days.
- **8** β_{00} parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 9 γ_{00} parameter. Covariate effect on staying in profile 0; higher X increases persistence.

FigEntropy 29

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

- 11 μ_{10} parameter. Profile specific mean for profile 0 and item 1.
- 12 μ_{20} parameter. Profile specific mean for profile 0 and item 2.
- 13 μ_{30} parameter. Profile specific mean for profile 0 and item 3.
- 14 μ_{40} parameter. Profile specific mean for profile 0 and item 4.
- 15 μ_{11} parameter. Profile specific mean for profile 1 and item 1.
- 16 μ_{21} parameter. Profile specific mean for profile 1 and item 2.
- 17 μ_{31} parameter. Profile specific mean for profile 1 and item 3.
- 18 μ_{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()
```

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.

30 FigIC

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

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

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()
```

```
## 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 common 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 ψ_T parameter. Variance in the common trait; reflects stable between-person differences.
- 2 λ_{t2} parameter. Factor loading for the common trait and item 2.
- 3 λ_{t3} parameter. Factor loading for the common trait and item 3.
- 4 λ_{t4} parameter. Factor loading for the common trait and item 4.
- 5 ψ_{n11} parameter. Trait-specific item 1 variance.
- 6 ψ_{p22} parameter. Trait-specific item 2 variance.
- 7 ψ_{p33} parameter. Trait-specific item 3 variance.
- 8 ψ_{p44} parameter. Trait-specific item 4 variance.
- 9 ψ_{s0} parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.
- 10 ψ_s parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.
- 11 λ_{s2} parameter. Factor loading for the common state and item 2.
- 12 λ_{s3} parameter. Factor loading for the common state and item 3.
- 13 λ_{s4} parameter. Factor loading for the common state and item 4.
- **14** θ_{11} parameter. Unique state variance for item 1.
- 15 θ_{22} parameter. Unique state variance for item 2.
- **16** θ_{33} parameter. Unique state variance for item 3.
- 17 θ_{44} parameter. Unique state variance for item 4.
- 18 ν_0 parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- 19 κ_0 parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.

32 FigPowerLTAEst

- **20** α_0 parameter. Baseline log-odds of being in profile 0 across days.
- 21 β_{00} parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 22 γ_{00} parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 23 γ_{10} parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- **24** μ_{10} parameter. Profile specific mean for profile 0 and item 1.
- 25 μ_{20} parameter. Profile specific mean for profile 0 and item 2.
- **26** μ_{30} parameter. Profile specific mean for profile 0 and item 3.
- 27 μ_{40} parameter. Profile specific mean for profile 0 and item 4.
- 28 μ_{11} parameter. Profile specific mean for profile 1 and item 1.
- **29** μ_{21} parameter. Profile specific mean for profile 1 and item 2.
- **30** μ_{31} parameter. Profile specific mean for profile 1 and item 3.
- 31 μ_{41} parameter. Profile specific mean for profile 1 and item 4.
- **32** ϕ_0 parameter. Autoregressive coefficient for profile 0.
- **33** ϕ_1 parameter. Autoregressive coefficient for profile 1.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

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

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 θ_{11} parameter. Unique state variance for item 1.
- 2 θ_{22} parameter. Unique state variance for item 2.
- 3 θ_{33} parameter. Unique state variance for item 3.
- 4 θ_{44} parameter. Unique state variance for item 4.
- 5 ν_0 parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- **6** κ_0 parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- 7 α_0 parameter. Baseline log-odds of being in profile 0 across days.
- **8** β_{00} parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- 9 γ_{00} parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- 10 γ_{10} parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.
- 11 μ_{10} parameter. Profile specific mean for profile 0 and item 1.
- 12 μ_{20} parameter. Profile specific mean for profile 0 and item 2.
- 13 μ_{30} parameter. Profile specific mean for profile 0 and item 3.
- 14 μ_{40} parameter. Profile specific mean for profile 0 and item 4.
- 15 μ_{11} parameter. Profile specific mean for profile 1 and item 1.
- 16 μ_{21} parameter. Profile specific mean for profile 1 and item 2.
- 17 μ_{31} parameter. Profile specific mean for profile 1 and item 3.
- 18 μ_{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()
```

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

34 FitCULTA1Profile

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)
```

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.

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()

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

FitCULTA1Profile 35

```
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),
```

36 FitCULTA2Profiles

```
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
FitCULTA1Profile(data = data)
## End(Not run)
```

FitCULTA2Profiles

Fit the Two-Profile CULTA Model

Description

Fits the two-profile CULTA model using Mplus.

Usage

```
FitCULTA2Profiles(
  data,
  wd = ".",
  ncores = 1L,
  mplus_bin = NULL,
```

FitCULTA2Profiles 37

```
starts = c(20, 4),
stiterations = 10,
stscale = 5
)
```

Arguments

data R object. Object of class simculta.

wd Character string. Working directory.

ncores Positive integer. Number of cores to use.

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.

Positive integer. Number of initial stage iterations.

stscale Positive integer. Random start scale.

Value

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

• call: Function call.

stiterations

• 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()

```
## 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>
```

38 FitCULTA2Profiles

```
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
{\tt common\_state\_loading} {\tt <- matrix(}
  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)
```

FitLTA2Profiles 39

```
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.

Usage

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

40 FitLTA2Profiles

Arguments

R object. Object of class simculta. data wd Character string. Working directory. Positive integer. Number of cores to use. ncores 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. Positive integer. Number of initial stage iterations. stiterations Positive integer. Random start scale. stscale

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()

```
## 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</pre>
```

FitLTA2Profiles 41

```
# 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
```

42 FitRILTA2Profiles

```
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.

Usage

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

FitRILTA2Profiles 43

Arguments

data R object. Object of class simculta.

wd Character string. Working directory.

ncores Positive integer. Number of cores to use.

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.

Value

stscale

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

Positive integer. Random start scale.

• 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()

```
## 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</pre>
```

44 FitRILTA2Profiles

```
# 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)
```

GenCULTA2Profiles

Simulate Observed Data from a Two-Profile CULTA Model

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,
    m,
    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
)
```

common_state_loading

0.

phi_0

Arguments

	n	Positive integer. Number of individuals.
	m	Positive integer. Number of measurement occasions.
	mu_x	Numeric. Mean of the covariate μ_X .
	sigma_x	Numeric. Variance of the covariate σ_X .
	nu_0	Numeric. Intercept ν_0 for the logistic model of initial profile membership.
	kappa_0	Numeric. Covariate effect κ_0 on initial profile membership.
	alpha_0	Numeric. Intercept α_0 for the logistic model of profile transitions across time.
	beta_00	Numeric. Effect β_{00} for self-persistence in profile 0 transitions.
	gamma_00	Numeric. Covariate effect γ_{00} on remaining in profile 0.
	gamma_10	Numeric. Covariate effect γ_{10} on transitioning from profile 1 to profile 0.
	mu_t	Numeric or vector of length q . Mean μ_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 ψ_T for the common trait factor.
	mu_p	Numeric vector of length p . Mean vector μ_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 Ψ_p for unique trait components.
common_trait_loading		
		Numeric matrix of size $p \times q$. Factor loading matrix specifying the influence of

Numeric matrix of size $p \times 1$. Factor loading matrix specifying the influence of

Numeric. Autoregressive coefficient ϕ_0 for the common state process in profile

the common trait on each observed item.

the common state on each observed item.

phi_1	Numeric. Autoregressive coefficient ϕ_1 for the common state process in profile 0.
psi_s0	Numeric. Variance ψ_{s0} of the initial common state.
psi_s	Numeric. Innovation variance ψ_s for the common state process.
theta	Numeric matrix of size $p \times p$. Positive definite covariance matrix Θ 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 $i \in \{1, ..., n\}$ denote the index for individuals, let $t \in \{0, ..., m-1\}$ denote the index measurement occasions, let $k \in \{1, ..., p\}$ 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 μ_X and variance σ_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):

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

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 $(t = 1, \dots, m-1)$, 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:

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

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 Trait_i represents shared individual differences that influence all items uniformly. It is drawn from a normal distribution with mean μ_T and variance ψ_T :

Common Trait_i
$$\sim \mathcal{N}\left(\mu_T, \psi_T\right)$$

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_i
$$\sim \mathcal{N}\left(\boldsymbol{\mu}_{p}, \boldsymbol{\Psi}_{p \times p}\right)$$

Combined Trait Variate

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

$$\operatorname{Trait}_{k,i} = \operatorname{Common} \operatorname{Trait}_{k,i} + \operatorname{Unique} \operatorname{Trait}_{k,i}$$

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 $\mathrm{State}_{i,t}$ evolves over time following a first-order autoregressive process:

Common State_{i,t} =
$$\phi_c \times \text{Common State}_{i,t-1} + \zeta_{i,t}$$
.

The initial common state is drawn from a normal distribution:

Common State_{i,0}
$$\sim \mathcal{N}(0, \psi_{s_0})$$
.

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

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

The autoregressive parameter ϕ_c depends on latent profile membership c:

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

Here, ϕ_0 and ϕ_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_{i,t}
$$\sim \mathcal{N}\left(0,\boldsymbol{\theta}\right)$$

where θ 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:

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

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
```

GenData 51

```
psi_s <- 0.290
theta <- 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
)
```

GenData

Simulate Data

Description

The function simulates data using the GenCULTA2Profiles() function.

Usage

```
GenData(taskid)
```

Arguments

taskid

Positive integer. Task ID.

52 InputCULTA1Profile

See Also

Other Data Generation Functions: GenCULTA2Profiles()

Examples

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

InputCULTA1Profile

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)
```

Arguments

data R object. Object of class simculta.

wd Character string. Working directory.

starts Positive integer. Number of initial stage 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()
```

InputCULTA1Profile 53

```
# 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 <- 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.

InputCULTA2Profiles 55

Usage

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

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.

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</pre>
```

```
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,
```

InputLTA2Profiles 57

```
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

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
)
```

Arguments

data

R object. Object of class simculta.

58 InputLTA2Profiles

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.

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()

```
## 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
```

InputLTA2Profiles 59

```
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,
 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,
```

60 InputRILTA2Profiles

```
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.

Usage

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

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.

Value

Writes data and input files in wd.

Author(s)

Ivan Jacob Agaloos Pesigan

InputRILTA2Profiles 61

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
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
```

62 logLik.fitculta

```
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 ------
InputRILTA2Profiles(data = data)
## End(Not run)
```

logLik.fitculta 63

Description

Extract Log-Likelihood

Usage

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

Arguments

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

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)
```

64 logLik.fitculta

```
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,
```

params 65

params

Simulation Parameters

Description

Simulation Parameters

Usage

data(params)

Format

A dataframe with 9 rows and 39 columns:

taskid Simulation Task ID.

n Sample size.

separation Level of separation. 0 for moderate, -1 for low, and 1 for strong.

m Measurement occasions.

mu_x μ_x parameter. Mean of the covariate.

sigma_x σ_x parameter. Variance of the covariate.

mu_10 μ_{10} parameter. Profile specific mean for profile 0 and item 1.

mu_20 μ_{20} parameter. Profile specific mean for profile 0 and item 2.

mu_30 μ_{30} parameter. Profile specific mean for profile 0 and item 3.

mu_40 μ_{40} parameter. Profile specific mean for profile 0 and item 4.

lambda_t2 λ_{t2} parameter. Factor loading for the common trait and item 2.

lambda_s2 λ_{s2} parameter. Factor loading for the common state and item 2.

66 params

- **lambda_t3** λ_{t3} parameter. Factor loading for the common trait and item 3.
- **lambda_s3** λ_{s3} parameter. Factor loading for the common state and item 3.
- **lambda_t4** λ_{t4} parameter. Factor loading for the common trait and item 4.
- **lambda_s4** λ_{s4} parameter. Factor loading for the common state and item 4.
- **theta_11** θ_{11} parameter. Unique state variance for item 1.
- theta_22 θ_{22} parameter. Unique state variance for item 2.
- **theta_33** θ_{33} parameter. Unique state variance for item 3.
- **theta_44** θ_{44} parameter. Unique state variance for item 4.
- **phi_0** ϕ_0 parameter. Autoregressive coefficient for profile 0.
- **psi_t** ψ_T parameter. Variance in the common trait; reflects stable between-person differences.
- $\mathbf{psi}_{-}\mathbf{p}_{-}\mathbf{11} \ \psi_{p11}$ parameter. Trait-specific item 1 variance.
- **psi_p_22** ψ_{p22} parameter. Trait-specific item 2 variance.
- **psi_p_33** ψ_{p33} parameter. Trait-specific item 3 variance.
- **psi_p_44** ψ_{p44} parameter. Trait-specific item 4 variance.
- **psi_s0** ψ_{s0} parameter. Initial-day variance of the common state; reflects variability in intoxication levels at observation start.
- **psi_s** ψ_s parameter. Residual state variance over days; captures within-person daily fluctuations not explained by trait or AR effects.
- **mu_11** μ_{11} parameter. Profile specific mean for profile 1 and item 1.
- **mu_21** μ_{21} parameter. Profile specific mean for profile 1 and item 2.
- **mu_31** μ_{31} parameter. Profile specific mean for profile 1 and item 3.
- **mu_41** μ_{41} parameter. Profile specific mean for profile 1 and item 4.
- **phi_1** ϕ_1 parameter. Autoregressive coefficient for profile 1.
- **nu_0** ν_0 parameter. Intercept for initial log-odds of profile 0 (vs. profile 1) when X=0.
- **alpha_0** α_0 parameter. Baseline log-odds of being in profile 0 across days.
- **kappa_0** κ_0 parameter. Covariate effect on initial profile membership; higher X increases odds of profile 0.
- **beta_00** β_{00} parameter. Increased odds of staying in profile 0 if previously in that profile; reflects persistence.
- **gamma_00** γ_{00} parameter. Covariate effect on staying in profile 0; higher X increases persistence.
- **gamma_10** γ_{10} parameter. Covariate effect on switching from state to profile 0; higher X increases transition odds.

Author(s)

Ivan Jacob Agaloos Pesigan

print.fitculta 67

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

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

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</pre>
```

68 print.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,
```

print.simculta 69

```
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

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

70 results_culta_est

Examples

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

results_culta_est

Simulation Results (CULTA Estimates)

Description

Simulation Results (CULTA Estimates)

Usage

```
data(results_culta_est)
```

results_entropy 71

Format

A dataframe with 486 rows and 20 columns:

taskid Task ID.

replications Number of replications.

parnames Parameter names

parameter Population parameter value.

model Fitted model.

separation Level of separation.

- 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.

se_bias Bias in standard error estimate.

coverage Coverage probability.

power Statistical power.

bias Bias.

Author(s)

Ivan Jacob Agaloos Pesigan

results_entropy

Simulation Results (Entropy)

Description

Simulation Results (Entropy)

Usage

data(results_entropy)

72 results_ic

Format

A dataframe with 27 rows and 5 columns:

TaskID Simulation Task ID.

N Sample size.

Separation Level of separation.

Model Fitted model.

Entropy Mean entropy.

Author(s)

Ivan Jacob Agaloos Pesigan

results_ic

Simulation Results (Information Criteria)

Description

Simulation Results (Information Criteria)

Usage

data(results_ic)

Format

A dataframe with 108 rows and 6 columns:

TaskID Simulation Task ID.

N Sample size.

Separation Level of separation.

Model Fitted model.

IC Information criteria type.

Value Mean information criteria.

Author(s)

Ivan Jacob Agaloos Pesigan

results_lta_est 73

results_lta_est

Simulation Results (LTA Estimates)

Description

Simulation Results (LTA Estimates)

Usage

```
data(results_lta_est)
```

Format

A dataframe with 486 rows and 20 columns:

taskid Task ID.

replications Number of replications.

parnames Parameter names

parameter Population parameter value.

model Fitted model.

separation Level of separation.

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.

se_bias Bias in standard error estimate.

coverage Coverage probability.

power Statistical power.

bias Bias.

Author(s)

74 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 75

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.

76 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 77

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.

78 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 79

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().

80 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 81

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)
```

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.

Value

The output is saved as an external file in output_folder.

Author(s)

82 SumFitCULTA2Profiles

SumFitCULTA1Profile Summary (FitCULTA1Profile)

Description

Summary (FitCULTA1Profile)

Usage

SumFitCULTA1Profile(taskid, reps, output_folder, overwrite, integrity)

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.

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

SumFitCULTA2Profiles Summary (FitCULTA2Profiles)

Description

Summary (FitCULTA2Profiles)

Usage

SumFitCULTA2Profiles(taskid, reps, output_folder, overwrite, integrity)

SumFitLTA2Profiles 83

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.

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

SumFitLTA2Profiles Summary (FitLTA2Profiles)

Description

Summary (FitLTA2Profiles)

Usage

SumFitLTA2Profiles(taskid, reps, output_folder, overwrite, integrity)

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.

Details

This function is executed via the Sum function.

84 SumFitRILTA2Profiles

Value

The output is saved as an external file in output_folder.

Author(s)

Ivan Jacob Agaloos Pesigan

SumFitRILTA2Profiles Summary (FitRILTA2Profiles)

Description

Summary (FitRILTA2Profiles)

Usage

SumFitRILTA2Profiles(taskid, reps, output_folder, overwrite, integrity)

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.

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)

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.

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

SumLTAParamsLTA2Profiles

Summary (LTAParamsLTA2Profiles)

Description

Summary (LTAParamsLTA2Profiles)

Usage

SumLTAParamsLTA2Profiles(taskid, reps, output_folder, overwrite, integrity)

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.

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

SumLTAParamsRILTA2Profiles

Summary (LTAParamsRILTA2Profiles)

Description

Summary (LTAParamsRILTA2Profiles)

Usage

SumLTAParamsRILTA2Profiles(taskid, reps, output_folder, overwrite, integrity)

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.

Details

This function is executed via the Sum function.

summary.fitculta 87

Value

The output is saved as an external file in output_folder.

Author(s)

Ivan Jacob Agaloos Pesigan

summary.fitculta

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 α . 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</pre>
```

88 summary.fitculta

```
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
# 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)
```

Arguments

taskid Positive integer. Task ID.

90 vcov.fitculta

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.

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 91

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
)
```

92 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 93

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_ic,72
Compress, 15	results_lta_est,73
* Data Generation Functions	* figure
GenCULTA2Profiles, 45	FigBiasCULTAEst, 23
GenData, 51	FigBiasLTAEst, 25
* Figure Functions	FigCoverageCULTAEst, 26
FigBiasCULTAEst, 23	FigCoverageLTAEst, 28
FigBiasLTAEst, 25	FigEntropy, 29
FigCoverageCULTAEst, 26	FigIC, 30
FigCoverageLTAEst, 28	FigPowerCULTAEst, 31
FigEntropy, 29	FigPowerLTAEst, 32
FigIC, 30	* fit
FigPowerCULTAEst, 31	FitCULTA1Profile, 34
FigPowerLTAEst, 32	FitCULTA2Profiles, 36
* Model Fitting Functions	FitLTA2Profiles, 39
FitCULTA1Profile, 34	FitRILTA2Profiles, 42
FitCULTA2Profiles, 36	SimFitCULTA1Profile, 75
FitLTA2Profiles, 39	SimFitCULTA2Profiles, 76
FitRILTA2Profiles, 42	SimFitLTA2Profiles, 77
InputCULTA1Profile, 52	SimFitRILTA2Profiles, 78
InputCULTA2Profiles, 54	* gendata
InputLTA2Profiles, 57	GenData, 51
<pre>InputRILTA2Profiles, 60</pre>	SimGenData, 80
* compress	* input
Compress, 15	<pre>InputCULTA1Profile, 52</pre>
* culta	<pre>InputCULTA2Profiles, 54</pre>
FitCULTA1Profile, 34	<pre>InputLTA2Profiles, 57</pre>
FitCULTA2Profiles, 36	<pre>InputRILTA2Profiles, 60</pre>
FitLTA2Profiles, 39	* manCULTA
FitRILTA2Profiles, 42	Compress, 15
GenCULTA2Profiles, 45	FigBiasCULTAEst, 23
InputCULTA1Profile, 52	FigBiasLTAEst, 25
InputCULTA2Profiles, 54	FigCoverageCULTAEst, 26
InputLTA2Profiles, 57	FigCoverageLTAEst, 28
InputRILTA2Profiles, 60	FigEntropy, 29
* data	FigIC, 30
params, 65	FigPowerCULTAEst, 31
results_culta_est,70	FigPowerLTAEst, 32
results_entropy, 71	FitCULTA1Profile, 34

INDEX 95

FitCULTA2Profiles, 36	InputCULTA1Profile, 52
FitLTA2Profiles, 39	<pre>InputCULTA2Profiles, 54</pre>
FitRILTA2Profiles, 42	<pre>InputLTA2Profiles, 57</pre>
GenCULTA2Profiles, 45	<pre>InputRILTA2Profiles, 60</pre>
GenData, 51	* parameters
<pre>InputCULTA1Profile, 52</pre>	params, 65
InputCULTA2Profiles, 54	results_culta_est,70
InputLTA2Profiles, 57	results_entropy,71
InputRILTA2Profiles, 60	results_ic, 72
Sim, 74	results_lta_est, 73
SimFitCULTA1Profile, 75	* simulation
SimFitCULTA2Profiles, 76	Sim, 74
SimFitLTA2Profiles, 77	SimFitCULTA1Profile, 75
SimFitRILTA2Profiles, 78	SimFitCULTA2Profiles, 76
SimFN, 79	SimFitLTA2Profiles, 77
SimGenData, 80	SimFitRILTA2Profiles, 78
SimProj, 81	SimFN, 79
Sum, 81	SimGenData, 80
SumFitCULTA1Profile, 82	SimProj, 81
SumFitCULTA2Profiles, 82	Sum, 81
SumFitLTA2Profiles, 83	SumFitCULTA1Profile, 82
SumFitRILTA2Profiles, 84	SumFitCULTA2Profiles, 82
SumLTAParamsCULTA2Profiles, 85	SumFitLTA2Profiles, 83
SumLTAParamsLTA2Profiles, 85	SumFitRILTA2Profiles, 84
SumLTAParamsRILTA2Profiles, 86	SumLTAParamsCULTA2Profiles, 85
SumParamsCULTA2Profiles, 89	SumLTAParamsLTA2Profiles, 85
* methods	SumLTAParamsRILTA2Profiles, 86
AIC.fitculta,3	SumParamsCULTA2Profiles, 89
anova.fitculta,5	* sim
as.data.frame.simculta,8	GenCULTA2Profiles, 45
as.matrix.simculta,9	* state
BIC.fitculta, 10	FitCULTA1Profile, 34
coef.fitculta, 13	FitCULTA2Profiles, 36
confint.fitculta, 16	FitLTA2Profiles, 39
converged, 18	FitRILTA2Profiles, 42
entropy, 21	GenCULTA2Profiles, 45
logLik.fitculta, 62	InputCULTA1Profile, 52
print.fitculta, 67	InputCULTA2Profiles, 54
print.simculta, 69	InputLTA2Profiles, 57
summary.fitculta, 87	InputRILTA2Profiles, 60
vcov.fitculta, 90	* summary
WriteData, 92	Sum, 81
* mixture	SumFitCULTA1Profile, 82
FitCULTA1Profile, 34	SumFitCULTA2Profiles, 82
FitCULTA2Profiles, 36	SumFitLTA2Profiles, 83
FitLTA2Profiles, 39	SumFitRILTA2Profiles, 83
FitRILTA2Profiles, 42	SumLTAParamsCULTA2Profiles, 85
GenCULTA2Profiles, 45	SumLTAParamsLTA2Profiles, 85

96 INDEX

SumLTAParamsRILTA2Profiles, 86 SumParamsCULTA2Profiles, 89	InputCULTA1Profile, 34, 37, 40, 43, 52, 55,
* trait	58,61 InputCULTA2Profiles, 34,37,40,43,52,54
* trait FitCULTA1Profile, 34	58, 61
FitCULTA2Profiles, 36	InputLTA2Profiles, 34, 37, 40, 43, 52, 55,
FitLTA2Profiles, 39	57, 61
FitRILTA2Profiles, 42	InputRILTA2Profiles, 34, 37, 40, 43, 52, 55
GenCULTA2Profiles, 45	58, 60
InputCULTA1Profile, 52	23,00
InputCULTA2Profiles, 54	logLik.fitculta,62
InputLTA2Profiles, 57	
InputRILTA2Profiles, 60	params, 65
inputation ites, ou	print.fitculta,67
AIC.fitculta, 3	print.simculta,69
anova.fitculta,5	1
as.data.frame.simculta, 8	results_culta_est, 70
as.matrix.simculta,9	results_entropy, 71
do.mati ix.oimedita,	results_ic, 72
BIC.fitculta, 10	results_lta_est, 73
bic. Titcuita, 10	Sim, 74
coef.fitculta, 13	SimFitCULTA1Profile, 75
Compress, 15	SimFitCULTA2Profiles, 76
confint.fitculta, 16	SimFitLTA2Profiles, 77
converged, 18	SimFitRILTA2Profiles, 78
converged, 10	SimFN, 79
ontropy 20	SimGenData, 80
entropy, 20	SimProj, 81
FigDiag(III TAFa+ 22 26 29 20 22 22	Sum, 81
FigBiasCULTAEst, 23, 26, 28–30, 32, 33 FigBiasLTAEst, 24, 25, 28–30, 32, 33	SumFitCULTA1Profile, 82
FigCoverageCULTAEst, 24, 26, 26, 29, 30, 32,	SumFitCULTA2Profiles, 82
33	SumFitLTA2Profiles, 83
FigCoverageLTAEst, 24, 26, 28, 28, 30, 32, 33	SumFitRILTA2Profiles, 84
FigEntropy, 24, 26, 28, 29, 29, 30, 32, 33	SumLTAParamsCULTA2Profiles, 85
FigIC, 24, 26, 28, 29, 30, 30, 32, 33	SumLTAParamsLTA2Profiles, 85
FigPowerCULTAEst, 24, 26, 28–30, 31, 33	SumLTAParamsRILTA2Profiles, 86
FigPowerLTAEst, 24, 26, 28–30, 31, 33	summary.fitculta,87
FitCULTA1Profile, 34, 37, 40, 43, 52, 55, 58,	SumParamsCULTA2Profiles, 89
61	
FitCULTA2Profiles, 34, 36, 40, 43, 52, 55,	vcov.fitculta,90
58, 61	
FitLTA2Profiles, 34, 37, 39, 43, 52, 55, 58,	WriteData,92
61	
FitRILTA2Profiles, 34, 37, 40, 42, 52, 55,	
58, 61	
GenCULTA2Profiles, 45, 52	
GenCULTA2Profiles(), 47, 51	
GenData, 50, 51	