Package ‘nlpsem’

June 4, 2023

Type Package

Title Linear and Nonlinear Longitudinal Process in Structural Equation Modeling Framework

Version 0.1.1

Description Provides computational tools for nonlinear longitudinal models, in particular the intrinsically nonlinear models, in four scenarios: (1) univariate longitudinal processes with growth factors, with or without covariates including time-invariant covariates (TICs) and time-varying covariates (TVCs); (2) multivariate longitudinal processes that facilitate the assessment of correlation or causation between multiple longitudinal variables; (3) multiple-group models for scenarios (1) and (2) to evaluate differences among manifested groups, and (4) longitudinal mixture models for scenarios (1) and (2), with an assumption that trajectories are from multiple latent classes. The methods implemented are introduced in Jin Liu (2023) <arXiv:2302.03237v2>.

License GPL (>= 3.0)

Encoding UTF-8

LazyData true

Depends R (>= 4.0.0)

Imports OpenMx, ggplot2, dplyr, tidyr, stringr, Matrix, nnet, readr

RoxygenNote 7.2.3

URL https://github.com/Veronica0206/nlpsem

BugReports https://github.com/Veronica0206/nlpsem/issues

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Jin Liu [aut, cre]

Maintainer Jin Liu <Veronica.Liu0206@gmail.com>

Repository CRAN

Date/Publication 2023-06-04 12:20:09 UTC
getEstimateStats: Calculate p-Values and Confidence Intervals of Parameters for a Fitted Model

Description
This function calculates p-values and confidence intervals (CIs) of parameters for a given model. It supports different types of CIs, including Wald CIs, likelihood-based CIs, bootstrap CIs, or all three.

Usage
getEstimateStats(model = NULL, est_in, p_values = TRUE, CI = TRUE, CI_type = "Wald", rep = NA, conf.level = 0.95)

Arguments
model A fitted mxModel object. This is the output from one of the estimation functions in this package. Default is NULL. This is only required when generating likelihood-based and bootstrap CIs.
est_in A data frame containing input estimates.
getEstimateStats

**p_values**  
A logical flag indicating whether to calculate p-values. Default is TRUE.

**CI**  
A logical flag indicating whether to compute confidence intervals. Default is TRUE.

**CI_type**  
A string specifying the type of confidence interval to compute. Supported options include "Wald", "likelihood", "bootstrap", or "all". Default is "Wald".

**rep**  
An integer specifying the number of replications for bootstrap. This is applicable if CI_type is "bootstrap" or "all". Default is NA.

**conf.level**  
A numeric value representing the confidence level for confidence interval calculation. Default is 0.95.

**Value**  
A data frame with calculated statistics (p-value, confidence intervals) added to the input estimates.

**References**


**Examples**

```r
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat
# Re-baseline the data so that the estimated initial status is for the starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
# Standardized time-invariant covariates
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)
# Fit bilinear spline latent growth curve model (fixed knots)
paraBLS_LGCM.r <- c(
  "mueta0", "mueta1", "mueta2", "knot",
```
BLS_LGCM_r <- getLGCM(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "BLS", intrinsic = FALSE,
  records = 1:9, res_scale = 0.1, paramOut = TRUE, names = paraBLS_LGCM.r)
## Output point estimate and standard errors
getEstimateStats(
  est_in = BLS_LGCM_r[[2]], CI_type = "Wald"
)

BLS_LGCM.TIC_f <- getLGCM(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "BLS", intrinsic = TRUE, records = 1:9,
  growth_TIC = c("ex1", "ex2"), res_scale = 0.1, paramOut = TRUE, names = paraBLS.TIC_LGCM.f)
## Output point estimate and standard errors
getEstimateStats(
  model = BLS_LGCM.TIC_f[[1]], est_in = BLS_LGCM.TIC_f[[2]], CI_type = "all", rep = 1000
)
getFigure

curveFun,
y_model = NULL,
t_var,
records,
m_var = NULL,
x_var = NULL,
x_type = NULL,
xstarts,
xlab = "Time",
outcome = "Process"
)

Arguments

model A fitted mxModel object. This is the output from one of the estimation functions in this package.
nClass An integer specifying the number of classes for the mixture model or multiple group model. Default is NULL, indicating a single-group model.
cluster_TIC A string or character vector representing the column name(s) for time-invariant covariate(s) indicating cluster formations. Default is NULL, indicating no such time-invariant covariates are present in the model.
grp_var A string specifying the column that indicates manifested classes when applicable.
sub_Model A string that specifies the sub-model for latent classes. Supported sub-models include "LGCM" (for latent growth curve models), "LCSM" (for latent change score models), "TVC" (for latent growth curve models or latent change score models with a time-varying covariate), "MGM" (for multivariate latent growth curve models or latent change score models), and "MED" (for longitudinal mediation models).
y_var A string or character vector representing the prefix of the column names for the outcome variable(s) at each study wave.
curveFun A string specifying the functional forms of the growth curve(s). Supported options for y_model = "LGCM" include: "linear" (or "LIN"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "bilinear spline" (or "BLS"). Supported options for y_model = "LCSM" include: "nonparametric" (or "NonP"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), and "Jenss-Bayley" (or "JB").
y_model A string that specifies how to fit longitudinal outcomes. Supported values are "LGCM" and "LCSM". By default, this is NULL as this argument only requires when sub_Model is "TVC" or "MGM".
t_var A string representing the prefix of the column names corresponding to the time variable at each study wave.
records A numeric vector representing the indices of the study waves.
m_var A string that specifies the prefix of the column names corresponding to the mediator variable at each time point. Default is NULL as this argument only requires when sub_Model is "MED".
getFigure

x_var A string specifying the baseline predictor if x_type = "baseline", or the prefix of the column names corresponding to the predictor variable at each study wave if x_type = "longitudinal". Default is NULL as this argument only requires when sub_Model is "MED".

x_type A string indicating the type of predictor variable used in the model. Supported values are "baseline" and "longitudinal". Default is NULL as this argument only requires when sub_Model is "MED".

xstarts A numeric value to indicate the starting time of the longitudinal process.

xlab A string representing the time unit (e.g., "Week", "Month", or "Year") for the x-axis. Default is "Time".

outcome A string or character vector representing the name(s) of the longitudinal process(es) under examination.

Value

A ggplot object or a list of ggplot objects, each representing a figure for the fitted model. If a list of ggplot objects is returned, it can be visualized using the print function.

Examples

```r
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat # Re-baseline the data so that the estimated initial status is for the starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
xstarts <- mean(baseT)

# Plot single group LGCM model
set.seed(20191029)
BLS_LGCM1 <- getLGCM(dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "BLS", intrinsic = FALSE, records = 1:9, res_scale = 0.1)
Figure1 <- getFigure(
  model = BLS_LGCM1, nClass = NULL, cluster_TIC = NULL, sub_Model = "LGCM",
  y_var = "M", curveFun = "BLS", y_model = "LGCM", t_var = "T", records = 1:9,
  m_var = NULL, x_var = NULL, x_type = NULL, xstarts = xstarts, xlab = "Month",
  outcome = "Mathematics"
)
print(Figure1)
# Plot mixture LGCM model
```

Derive Individual Factor Scores for Each Latent Variable Included in Model

Description

This function computes individual factor scores for each latent variable in a given model. It supports three types of factor scores: maximum likelihood, weighted maximum likelihood, and regression.

Usage

getIndFS(model, FS_type = "Regression")

Arguments

model

A fitted mxModel object. This is the output from one of the estimation functions in this package.

FS_type

A string specifying the type of factor scores to compute. Supported options include "ML" (for Maximum Likelihood), "WeightedML" (for Weighted Maximum Likelihood), and "Regression". Default is "Regression".

Value

A list containing two elements: scores_est, the factor score estimates, and scores_se, the standard errors of the factor score estimates.

References

Examples

OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat
# Re-baseline the data so that the estimated initial status is for the starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
# Standardized time-invariant covariates
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)

# Fit bilinear spline latent growth curve model (fixed knots)
LIN_LGCM <- getLGCM(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "linear",
  intrinsic = FALSE, records = 1:9, growth_TIC = NULL, res_scale = 0.1
)
getIndFS(model = LIN_LGCM, FS_type = "Regression")
# Fit bilinear spline latent growth curve model (random knots) with time-invariant covariates for
# mathematics development
## Fit the model
BLS_LGCM.TIC_f <- getLGCM(dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "BLS",
  intrinsic = TRUE, records = 1:9, growth_TIC = c("ex1", "ex2"),
  res_scale = 0.1)
## Output point estimate and standard errors
getIndFS(model = BLS_LGCM.TIC_f, FS_type = "Regression")

getLatentKappa(label1, label2, conf.level = 0.95)

getLatentKappa

Compute Latent Kappa Coefficient for Agreement between Two Latent Label Sets

Description

This function calculates the latent kappa, a measure of agreement between two sets of latent categorical labels. It also computes the confidence interval and provides a qualitative interpretation of the agreement level.

Usage

getLatentKappa(label1, label2, conf.level = 0.95)
getLatentKappa

Arguments

- **label1**: A factor vector representing the first set of latent categorical labels.
- **label2**: A factor vector representing the second set of latent categorical labels.
- **conf.level**: A numeric value representing the confidence level for the confidence interval of the kappa statistic. The default value is $0.95$.

Value

A list with two elements: The first is a string that provides the kappa statistic along with its confidence interval. The second is a string describing the level of agreement (such as "Perfect Agreement", "Slight Agreement", etc.).

References


Examples

```R
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
data("RMS_dat")
RMS_dat0 <- RMS_dat
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)
RMS_dat0$gx1 <- scale(RMS_dat0$INCOME)
RMS_dat0$gx2 <- scale(RMS_dat0$EDU)

## Fit a growth mixture model with no TICs
set.seed(20191029)
MIX_BLS_LGCM_r <- getMIX(
    dat = RMS_dat0, prop_starts = c(0.33, 0.34, 0.33), sub_Model = "LGCM",
    cluster_TIC = NULL, y_var = "M", t_var = "T", records = 1:9,
    curveFun = "BLS", intrinsic = FALSE, res_scale = list(0.3, 0.3, 0.3),
    growth_TIC = NULL, tries = 10)
```
## Membership of each individual from growth mixture model with no TICs

```r
label1 <- getPosterior(
  model = MIX_BLS_LGCM_r, nClass = 3, label = FALSE, cluster_TIC = NULL
)
```

```r
set.seed(20191029)
```

## Fit a growth mixture model with growth TICs and cluster TICs

```r
MIX_BLS_LGCM.TIC_r <- getMIX(
  dat = RMS_dat0, prop_starts = c(0.33, 0.34, 0.33), sub_Model = "LGCM",
  cluster_TIC = c("gx1", "gx2"), y_var = "M", t_var = "T", records = 1:9,
  curveFun = "BLS", intrinsic = FALSE, res_scale = list(0.3, 0.3, 0.3),
  growth_TIC = c("ex1", "ex2"), tries = 10
)
```

## Membership of each individual from growth mixture model with growth TICs and cluster TICs

```r
label2 <- getPosterior(
  model = MIX_BLS_LGCM.TIC_r, nClass = 3, label = FALSE,
  cluster_TIC = c("gx1", "gx2")
)
```

## Calculate membership between two sets of labels

```r
getLatentKappa(label1 = label1$membership, label2 = label2$membership)
```

---

**getLCSM**

*Fit a Latent Change Score Model with a Time-invariant Covariate If Any*

---

**Description**

This function fits a latent change score model with or without time-invariant covariates using the given data. It manages model setup, optimization, and if requested, outputs parameter estimates and standard errors.

**Usage**

```r
getLCSM(
  dat, 
  t_var, 
  y_var, 
  curveFun, 
  intrinsic = TRUE, 
  records, 
  growth_TIC = NULL, 
  starts = NULL, 
  res_scale = NULL, 
  tries = NULL, 
  OKStatus = 0, 
  jitterD = "runif", 
  loc = 1, 
  scale = 0.25,
)```
Arguments

dat A wide-format data frame, with each row corresponding to a unique ID. It contains the observed variables with repeated measurements and occasions, and time-invariant covariates (TICs) if any.
t_var A string specifying the prefix of the column names corresponding to the time variable at each study wave.
y_var A string specifying the prefix of the column names corresponding to the outcome variable at each study wave.
curveFun A string specifying the functional form of the growth curve. Supported options for latent change score models include: "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "nonparametric" (or "NonP").
intrinsic A logical flag indicating whether to build an intrinsically nonlinear longitudinal model. Default is TRUE.
records A numeric vector specifying indices of the study waves.
growth_TIC A string or character vector specifying the column name(s) of time-invariant covariate(s) contributing to the variability of growth factors if any. Default is NULL, indicating no growth TICs are included in the model.
starts A list containing initial values for the parameters. Default is NULL, indicating no user-specified initial values.
res_scale A numeric value representing the scaling factor for the initial calculation of the residual variance. This value should be between 0 and 1, exclusive. By default, this is NULL, as it is unnecessary when the user specifies the initial values using the starts argument.
tries An integer specifying the number of additional optimization attempts. Default is NULL.
OKStatus An integer (vector) specifying acceptable status codes for convergence. Default is 0.
jitterD A string specifying the distribution for jitter. Supported values are: "runif" (uniform distribution), "rnorm" (normal distribution), and "rcauchy" (Cauchy distribution). Default is "runif".
loc A numeric value representing the location parameter of the jitter distribution. Default is 1.
scale A numeric value representing the scale parameter of the jitter distribution. Default is 0.25.
paramOut A logical flag indicating whether to output the parameter estimates and standard errors. Default is FALSE.
names A character vector specifying parameter names. Default is NULL.
getLCSM

Value

A list containing the fitted latent change score model and, if paramOut = TRUE, a data frame with parameter estimates and standard errors.

References


Examples

```r
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat
# Re-baseline the data so that the estimated initial status is for the starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- (RMS_dat0$T1 - baseT)/12
RMS_dat0$T2 <- (RMS_dat0$T2 - baseT)/12
RMS_dat0$T3 <- (RMS_dat0$T3 - baseT)/12
RMS_dat0$T4 <- (RMS_dat0$T4 - baseT)/12
RMS_dat0$T5 <- (RMS_dat0$T5 - baseT)/12
RMS_dat0$T6 <- (RMS_dat0$T6 - baseT)/12
RMS_dat0$T7 <- (RMS_dat0$T7 - baseT)/12
RMS_dat0$T8 <- (RMS_dat0$T8 - baseT)/12
RMS_dat0$T9 <- (RMS_dat0$T9 - baseT)/12
# Standardized time-invariant covariates
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)
# Fit nonparametric change score model for reading development
## Fit model
NonP_LCSM <- getLCSM(
  dat = RMS_dat0, t_var = "T", y_var = "R", curveFun = "nonparametric",
  intrinsic = FALSE, records = 1:9, res_scale = 0.1
)
```

getLGCM

Fit a Latent Growth Curve Model with Time-invariant Covariate (If Any)

Description

This function fits a latent growth curve model with or without time-invariant covariates to the provided data. It manages model setup, optimization, and if requested, outputs parameter estimates and standard errors.

Usage

getLGCM(
  dat,
  t_var,
  y_var,
  curveFun,
  intrinsic = TRUE,
  records,
  growth_TIC = NULL,
  starts = NULL,
  res_scale = NULL,
  tries = NULL,
  OKStatus = 0,
  jitterD = "runif",
  loc = 1,
  scale = 0.25,
  paramOut = FALSE,
  names = NULL
)

Arguments

dat A wide-format data frame, with each row corresponding to a unique ID. It contains the observed variables with repeated measurements and occasions, and time-invariant covariates (TICs) if any.

t_var A string specifying the prefix of the column names corresponding to the time variable at each study wave.

y_var A string specifying the prefix of the column names corresponding to the outcome variable at each study wave.

curveFun A string specifying the functional form of the growth curve. Supported options for latent growth curve models are: "linear" (or "LIN"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "bilinear spline" (or "BLS").

intrinsic A logical flag indicating whether to build an intrinsically nonlinear longitudinal model. Default is TRUE.
getLGCM

records
A numeric vector specifying indices of the study waves.

growth_TIC
A string or character vector specifying the column name(s) of time-invariant
covariate(s) contributing to the variability of growth factors if any. Default is
NULL, indicating no growth TICs are included in the model.

starts
A list containing initial values for the parameters. Default is NULL, indicating no
user-specified initial values.

res_scale
A numeric value representing the scaling factor for the initial calculation of the
residual variance. This value should be between 0 and 1, exclusive. By default,
this is NULL, as it is unnecessary when the user specifies the initial values using
the starts argument.

tries
An integer specifying the number of additional optimization attempts. Default
is NULL.

OKStatus
An integer (vector) specifying acceptable status codes for convergence. Default
is 0.

jitterD
A string specifying the distribution for jitter. Supported values are: "runif"
(uniform distribution), "rnorm" (normal distribution), and "rcauchy" (Cauchy
distribution). Default is "runif".

loc
A numeric value representing the location parameter of the jitter distribution.
Default is 1.

scale
A numeric value representing the scale parameter of the jitter distribution. De-
fault is 0.25.

paramOut
A logical flag indicating whether to output the parameter estimates and standard
errors. Default is FALSE.

names
A character vector specifying parameter names. Default is NULL.

Value
A list containing the fitted latent growth curve model and, if paramOut = TRUE, a data frame with
parameter estimates and standard errors.

References
pretable Parameters from Reparameterizing Longitudinal Models: Transformation Matrices
between Growth Factors in Two Parameter Spaces". Journal of Educational and Behavioral
Statistics. doi:10.3102/10769986211052009
• Sterba, S. K. (2014). "Fitting Nonlinear Latent Growth Curve Models With Individually Vary-
doi:10.1080/10705511.2014.919828

Examples
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat
# Re-baseline the data so that the estimated initial status is for the
# starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
# Standardized time-invariant covariates
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)

# Fit bilinear spline latent growth curve model (fixed knots)
BLS_LGCM_r <- getLGCM(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "bilinear spline",
  intrinsic = FALSE, records = 1:9, growth_TIC = NULL, res_scale = 0.1
)
# Fit bilinear spline latent growth curve model (random knots) with
# time-invariant covariates for mathematics development
## Define parameter names
paraBLS.TIC_LGCM.f <- c(
  "alpha0", "alpha1", "alpha2", "alphag",
  paste(paste("psi", c("00", "01", "02", "0g", "11", "12", "1g", "22", "2g", "gg")),
  "residuals", paste(paste("bet1", c(0:2, "g")), paste(paste("beta2", c(0:2, "g")))),
  paste("mux", 1:2), paste(paste("phi", c("11", "12", "22")),
  "mueta0", "mueta1", "mueta2", "mu_knot"
)
## Fit the model
BLS_LGCM.TIC_f <- getLGCM(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "bilinear spline",
  intrinsic = TRUE, records = 1:9, growth_TIC = c("ex1", "ex2"), res_scale = 0.1,
  paramOut = TRUE, names = paraBLS.TIC_LGCM.f
)
## Output point estimate and standard errors
BLS_LGCM.TIC_f[[2]]

---

**getLRT**

Perform Likelihood Ratio Test (LRT) for Comparing Full and Reduced Models

**Description**

This function performs the likelihood ratio test (LRT) to compare a full model (an intrinsically nonlinear longitudinal model) with a corresponding parsimonious alternative (a non-intrinsically
nonlinear longitudinal model). It also provides an option to perform bootstrapping for the comparison.

Usage

getLRT(full, reduced, boot = FALSE, replications = NA)

Arguments

full A fitted mxModel object for the full model.
reduced A fitted mxModel object for the reduced model.
boot A logical flag indicating whether to perform bootstrapping for the comparison. Default is FALSE.
replications An integer specifying the number of bootstrap replications if boot is TRUE. Default is NA.

Value

A data frame containing the number of free parameters, estimated likelihood (-2ll), degrees of freedom, differences in log-likelihood and degrees of freedom, p-values, AIC, and BIC for both the full and reduced models.

Examples

```r
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat
# Re-baseline the data so that the estimated initial status is for the starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
# Fit bilinear spline growth model with random knot (intrinsically nonlinear model)
BLS_LGCM_f <- getLGCM(dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "bilinear spline",
    intrinsic = TRUE, records = 1:9, res_scale = 0.1)
# Fit bilinear spline growth model with fix knot (non-intrinsically nonlinear model)
BLS_LGCM_r <- getLGCM(dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "bilinear spline",
    intrinsic = FALSE, records = 1:9, res_scale = 0.1)
# Likelihood ratio test
getLRT(full = BLS_LGCM_f, reduced = BLS_LGCM_r, boot = FALSE, replications = NA)
```
**getMediation**

*Fit a Longitudinal Mediation Model*

**Description**

This function fits a longitudinal mediation model to the provided data. It manages model setup, optimization, and if requested, outputs parameter estimates and standard errors.

**Usage**

```r
getMediation(
  dat, t_var, y_var, m_var, x_type, x_var, curveFun, records, starts = NULL,
  res_scale = NULL, res_cor = NULL, tries = NULL, OKStatus = 0,
  jitterD = "runif", loc = 1, scale = 0.25, paramOut = FALSE, names = NULL
)
```

**Arguments**

- **dat**
  A wide-format data frame, with each row corresponding to a unique ID. It contains the observed variables with repeated measurements and occasions for multiple longitudinal processes and a baseline predictor when applicable.

- **t_var**
  A vector of strings, with each element representing the prefix for column names related to the time variable for the corresponding longitudinal variable at each study wave.

- **y_var**
  A string specifying the prefix of the column names corresponding to the outcome variable at each study wave.

- **m_var**
  A string specifying the prefix of the column names corresponding to the mediator variable at each study wave.

- **x_type**
  A string indicating the type of predictor variable used in the model. Supported values are "baseline" and "longitudinal".
### getMediation

**x_var**
A string specifying the baseline predictor if `x_type = "baseline"`, or the prefix of the column names corresponding to the predictor variable at each study wave if `x_type = "longitudinal"`.

**curveFun**
A string specifying the functional form of the growth curve. Supported options include: "linear" (or "LIN"), and "bilinear spline" (or "BLS").

**records**
A list of numeric vectors, with each vector specifying the indices of the observed study waves for the corresponding longitudinal variable.

**starts**
A list containing initial values for the parameters. Default is `NULL`, indicating no user-specified initial values.

**res_scale**
A numeric vector with each element representing the scaling factor for the initial calculation of the residual variance. These values should be between 0 and 1, exclusive. By default, this is `NULL`, as it is unnecessary when the user specifies the initial values using the `starts` argument.

**res_cor**
A numeric value or vector for user-specified residual correlation between any two longitudinal processes to calculate the corresponding initial value. By default, this is `NULL`, as it is unnecessary when the user specifies the initial values using the `starts` argument.

**tries**
An integer specifying the number of additional optimization attempts. Default is `NULL`.

**OKStatus**
An integer (vector) specifying acceptable status codes for convergence. Default is 0.

**jitterD**
A string specifying the distribution for jitter. Supported values are: "runif" (uniform distribution), "rnorm" (normal distribution), and "rcauchy" (Cauchy distribution). Default is "runif".

**loc**
A numeric value representing the location parameter of the jitter distribution. Default is 1.

**scale**
A numeric value representing the scale parameter of the jitter distribution. Default is 0.25.

**paramOut**
A logical flag indicating whether to output the parameter estimates and standard errors. Default is `FALSE`.

**names**
A character vector specifying parameter names. Default is `NULL`.

### Value
A list containing the fitted model and, if `paramOut = TRUE`, a data frame with parameter estimates and standard errors.

### References


Examples

```R
set.seed(20191029)
Med2_LGCM_LIN <- getMediation(
  dat = RMS_dat0, t_var = rep("T", 2), y_var = "M", m_var = "R", x_type = "baseline",
  x_var = "ex1", curveFun = "LIN", records = list(1:9, 1:9), res_scale = c(0.1, 0.1),
  res_cor = 0.3
)
```

```R
# Example 2: Longitudinal predictor, bilinear spline functional form
## Define parameter names
paraMed2_BLS <- c(
  "muetaX1", "muetaXr", "muetaX2", "mugX",
  paste0("psi", c("X1X1", "X1Xr", "X1X2", "XrXr", "XrX2", "X2X2")),
  "alphaM1", "alphaMr", "alphaM2", "mugM",
  paste0("psi", c("M1M1", "M1Mr", "M1M2", "MrMr", "MrM2", "M2M2"), ",", "r"),
  "alphaY1", "alphaYr", "alphaY2", "mugY",
  paste0("psi", c("Y1Y1", "Y1Yr", "Y1Y2", "YrYr", "YrY2", "Y2Y2"), ",", "r"),
  paste0("beta", c("X1Y1", "X1Yr", "X1Y2", "XrYr", "XrY2", "X2Y2",
    "X1M1", "X1Mr", "X1M2", "XrMr", "XrM2", "X2M2",
    "M1Y1", "M1Yr", "M1Y2", "MrYr", "MrY2", "M2Y2")),
  "muetaM1", "muetaMr", "muetaM2", "muetaY1", "muetaYr", "muetaY2",
  paste0("mediator", c("111", "11r", "112", "1rr", "1r2",
    "122", "rr2", "r22", "rrr", "222")),
```
getMGM

Fit a Multivariate Latent Growth Curve Model or Multivariate Latent Change Score Model

Description

This function fits a multivariate latent growth curve model or a multivariate latent change score model with the provided data. It manages model setup, optimization, and if requested, outputs parameter estimates and standard errors.

Usage

g MGM(
  dat,
  t_var,
  y_var,
  curveFun,
  intrinsic = TRUE,
  records,
  y_model,
  starts = NULL,
  res_scale = NULL,
  res_cor = NULL,
  tries = NULL,
  OKStatus = 0,
  jitterD = "runif",
  loc = 1,
  scale = 0.25,
  paramOut = FALSE,
  names = NULL
)

## Fit model
set.seed(20191029)
Med3_LGCM_BLS <- getMediation(
  dat = RMS_dat0, t_var = rep("T", 3), y_var = "S", m_var = "M", x_type = "longitudinal",
  x_var = "R", curveFun = "bilinear spline", records = list(2:9, 1:9, 1:9),
  res_scale = c(0.1, 0.1, 0.1), res_cor = c(0.3, 0.3), tries = 10, paramOut = TRUE,
  names = paraMed3_BLS
)
Med3_LGCM_BLS[[2]]
Arguments

dat  A wide-format data frame, with each row corresponding to a unique ID. It contains the observed variables with repeated measurements and occasions for multiple longitudinal outcomes.

t_var  A vector of strings, with each element representing the prefix for column names related to the time variable for the corresponding outcome variable at each study wave.

y_var  A vector of strings, with each element representing the prefix for column names corresponding to a particular outcome variable at each study wave.

curveFun  A string specifying the functional forms of the growth curve(s). Supported options for y_model = "LGCM" include: "linear" (or "LIN"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "bilinear spline" (or "BLS"). Supported options for y_model = "LCSM" include: "nonparametric" (or "NonP"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), and "Jenss-Bayley" (or "JB").

intrinsic  A logical flag indicating whether to build an intrinsically nonlinear longitudinal model. Default is TRUE.

records  A list of numeric vectors, with each vector specifying the indices of the observed study waves for the corresponding outcome variable.

y_model  A string specifying how to fit the longitudinal outcome. Supported values are "LGCM" and "LCSM".

starts  A list containing initial values for the parameters. Default is NULL, indicating no user-specified initial values.

res_scale  A numeric vector with each element representing the scaling factor for the initial calculation of the residual variance. These values should be between 0 and 1, exclusive. By default, this is NULL, as it is unnecessary when the user specifies the initial values using the starts argument.

res_cor  A numeric value or vector for user-specified residual correlation between any two longitudinal outcomes to calculate the corresponding initial value. By default, this is NULL, as it is unnecessary when the user specifies the initial values using the starts argument.

tries  An integer specifying the number of additional optimization attempts. Default is NULL.

OKStatus  An integer (vector) specifying acceptable status codes for convergence. Default is 0.

jitterD  A string specifying the distribution for jitter. Supported values are: "runif" (uniform distribution), "rnorm" (normal distribution), and "rcauchy" (Cauchy distribution). Default is "runif".

loc  A numeric value representing the location parameter of the jitter distribution. Default is 1.

scale  A numeric value representing the scale parameter of the jitter distribution. Default is 0.25.

paramOut  A logical flag indicating whether to output the parameter estimates and standard errors. Default is FALSE.

names  A character vector specifying parameter names. Default is NULL.
Value

A list containing the fitted model and, if paramOut = TRUE, a data frame with parameter estimates and standard errors.

References


Examples

```
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat
# Re-baseline the data so that the estimated initial status is for the starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT

# Fit linear multivariate latent growth curve model
LIN_PLGCM_f <- getMGM(
  dat = RMS_dat0, t_var = c("T", "T"), y_var = c("R", "M"), curveFun = "LIN",
  intrinsic = FALSE, records = list(1:9, 1:9), y_model = "LGCM", res_scale = c(0.1, 0.1),
  res_cor = 0.3
)

# Fit bilinear spline multivariate latent growth curve model (random knots)
## Define parameter names
paraBLS_PLGCM.f <- c(
  "Y_mueta0", "Y_mueta1", "Y_mueta2", "Y_knot",
paste0("Y_psi", c("00", "01", "02", "0g", "11", "12", "1g", "22", "2g", "gg")),
  "Y_res",
  "Z_mueta0", "Z_mueta1", "Z_mueta2", "Z_knot",
paste0("Z_psi", c("00", "01", "02", "0g", "11", "12", "1g", "22", "2g", "gg")),
  "Z_res",
paste0("YZ_psi", c("00", "10", "20", "g0", "01", "11", "21", "g1",
  "02", "12", "22", "g2", "0g", "1g", "2g", "gg")),
  "YZ_res"
)
## Fit model
BLS_PLGCM_f <- getMGM(
  dat = RMS_dat0, t_var = c("T", "T"), y_var = c("R", "M"), curveFun = "BLS",
  intrinsic = TRUE, records = list(1:9, 1:9), y_model = "LGCM", res_scale = c(0.1, 0.1),
  res_cor = 0.3, paramOut = TRUE, names = paraBLS_PLGCM.f
)```
getMGroup

Fit a Longitudinal Multiple Group Model

Description

This function fits a longitudinal multiple group model based on the specified sub-model. Supported submodels include:

- Latent growth curve models,
- Latent change score models,
- Latent growth curve models or latent change score models with a time-varying covariate,
- Multivariate latent growth curve models or multivariate latent change score models,
- Longitudinal mediation models.

For the first three submodels, time-invariant covariates are allowed.

Usage

getMGroup(
  dat,
  grp_var,
  sub_Model,
  t_var,
  records,
  y_var,
  curveFun,
  intrinsic = NULL,
  y_model = NULL,
  m_var = NULL,
  x_var = NULL,
  x_type = NULL,
  TVC = NULL,
  decompose = NULL,
  growth_TIC = NULL,
  starts = NULL,
  res_scale = NULL,
  res_cor = NULL,
  tries = NULL,
  OKStatus = 0,
  jitterD = "runif",
  loc = 1,
  scale = 0.25,
getMGroup

\[ \text{paramOut} = \text{FALSE}, \]
\[ \text{names} = \text{NULL} \]

Arguments

**dat**
A wide-format data frame, with each row corresponding to a unique ID. It contains the observed variables with repeated measurements and occasions for each longitudinal process, and time-invariant covariates (TICs) if any.

**grp_var**
A string specifying the column that indicates manifested classes.

**sub_Model**
A string that specifies the sub-model for manifested classes. Supported sub-models include "LGCM" (for latent growth curve models), "LCSM" (for latent change score models), "TVC" (for latent growth curve models or latent change score models with a time-varying covariate), "MGM" (for multivariate latent growth curve models or latent change score models), and "MED" (for longitudinal mediation models).

**t_var**
A string specifying the prefix of the column names corresponding to the time variable for each study wave. This applies when sub_Model is "LGCM", "LCSM" or "TVC". For sub_Model being "MGM" or "MED", t_var should be a string vector where each element corresponds to the time variable prefix for each respective longitudinal process.

**records**
A numeric vector denoting the indices of the observed study waves. This applies when sub_Model is "LGCM", "LCSM" or "TVC". For sub_Model being "MGM" or "MED", records should be a list of numeric vectors, where each vector provides the indices of the observed study waves for each longitudinal process.

**y_var**
A string defining the prefix of the column names corresponding to the outcome variable for each study wave. This is applicable when sub_Model is not "MGM". For sub_Model being "MGM", y_var should be a string vector where each element corresponds to the prefix of the column names for each outcome variable across the study waves.

**curveFun**
A string specifying the functional forms of the growth curve(s). Supported options for y_model = "LGCM" include: "linear" (or "LIN"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "bilinear spline" (or "BLS"). Supported options for y_model = "LCSM" include: "nonparametric" (or "NonP"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), and "Jenss-Bayley" (or "JB").

**intrinsic**
A logical flag indicating whether to build an intrinsically nonlinear longitudinal model. By default, this is NULL, as it is unnecessary when sub_Model is "MED".

**y_model**
A string that specifies how to fit longitudinal outcomes. Supported values are "LGCM" and "LCSM". By default, this is NULL as this argument only requires when sub_Model is "TVC" or "MGM".

**m_var**
A string that specifies the prefix of the column names corresponding to the mediator variable at each study wave. By default, this is NULL as this argument only requires when sub_Model is "MED".

**x_var**
A string specifying the baseline predictor if x_type = "baseline", or the prefix of the column names corresponding to the predictor variable at each study wave
if \( x\_type = \text{"longitudinal"} \). By default, this is NULL as this argument only requires when \( \text{sub\_Model} \) is \"MED\".

\[ x\_type \]
A string indicating the type of predictor variable used in the model. Supported values are \"baseline\" and \"longitudinal\". By default, this is NULL as this argument only requires when \( \text{sub\_Model} \) is \"MED\".

\[ TVC \]
A string that specifies the prefix of the column names corresponding to the time-varying covariate at each time point. By default, this is NULL as this argument only requires when \( \text{sub\_Model} \) is \"TVC\".

\[ \text{decompose} \]
An integer specifying the decomposition option for temporal states. Supported values include 0 (no decomposition), 1 (decomposition with interval-specific slopes as temporal states), 2 (decomposition with interval-specific changes as temporal states), and 3 (decomposition with change-from-baseline as temporal states). By default, this is NULL as this argument only requires when \( \text{sub\_Model} \) is \"TVC\".

\[ \text{growth\_TIC} \]
A string or character vector of column names of time-invariant covariate(s) accounting for the variability of growth factors if any. Default is NULL, indicating no growth TICs present in the model.

\[ \text{starts} \]
A list containing initial values for the parameters. Default is NULL, indicating no user-specified initial values.

\[ \text{res\_scale} \]
A list where each element is a (vector of) numeric scaling factor(s) for residual variance to calculate the corresponding initial value for a latent class, between 0 and 1 exclusive. By default, this is NULL, as it is unnecessary when the user specifies the initial values using the \( \text{starts} \) argument.

\[ \text{res\_cor} \]
A list where each element is a (vector of) numeric initial value(s) for residual correlation in each class. It needs to be specified if the \( \text{sub\_Model} \) is \"TVC\" (when \( \text{decompose} \neq 0 \)), \"MGM\", or \"MED\". By default, this is NULL, as it is unnecessary when the user specifies the initial values using the \( \text{starts} \) argument.

\[ \text{tries} \]
An integer specifying the number of additional optimization attempts. Default is NULL.

\[ \text{OKStatus} \]
An integer (vector) specifying acceptable status codes for convergence. Default is 0.

\[ \text{jitterD} \]
A string specifying the distribution for jitter. Supported values are: \"runif\" (uniform distribution), \"rnorm\" (normal distribution), and \"rcauchy\" (Cauchy distribution). Default is \"runif\".

\[ \text{loc} \]
A numeric value representing the location parameter of the jitter distribution. Default is 1.

\[ \text{scale} \]
A numeric value representing the scale parameter of the jitter distribution. Default is 0.25.

\[ \text{paramOut} \]
A logical flag indicating whether to output the parameter estimates and standard errors. Default is FALSE.

\[ \text{names} \]
A character vector specifying parameter names. Default is NULL.

Value

A list containing the fitted latent change score model and, if \( \text{paramOut} = \text{TRUE} \), a data frame with parameter estimates and standard errors.
Examples

```r
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
data("RMS_dat")
RMS_dat0 <- RMS_dat
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)
MGroup_BLS_LGCM.TIC_f <- getMGroup(
  dat = RMS_dat0, grp_var = "SEX", sub_Model = "LGCM", y_var = "M", t_var = "T",
  records = 1:9, curveFun = "BLS", intrinsic = FALSE, res_scale = list(0.3, 0.3)
)
paraBLS.TIC_LGCM.f <- c("alpha0", "alpha1", "alpha2", "alphag",
  paste0("psi", c("00", "01", "02", "0g", "11", "12", "1g", "22", "2g", "gg")),
  "residuals", paste0("beta1", c(0:2, "g")), paste0("beta2", c(0:2, "g")),
  paste0("mux", 1:2), paste0("phi", c("11", "12", "22")),
  "mueta0", "mueta1", "mueta2", "mu_knot"
)
set.seed(20191029)
MGroup_BLS_LGCM.TIC_f <- getMGroup(
  dat = RMS_dat0, grp_var = "SEX", sub_Model = "LGCM", y_var = "M", t_var = "T",
  records = 1:9, curveFun = "BLS", intrinsic = TRUE, res_scale = list(0.3, 0.3),
  growth_TIC = c("ex1", "ex2"), tries = 10, paramOut = TRUE, names = paraBLS.TIC_LGCM.f
)
MGroup_BLS_LGCM.TIC_f[[2]]
```

---

getMIX  
*Fit a Longitudinal Mixture Model*

Description

This function fits a longitudinal mixture model based on the specified sub-model. Supported sub-models include:

- Latent growth curve models,
- Latent change score models,
- Latent growth curve models or latent change score models with a time-varying covariate,
Multivariate latent growth curve models or multivariate latent change score models,
• Longitudinal mediation models.

Time-invariant covariates are allowed for the first three submodels.

Usage

getMIX(
  dat,  
  prop_starts,  
  sub_Model,  
  cluster_TIC = NULL,  
  t_var,  
  records,  
  y_var,  
  curveFun,  
  intrinsic = NULL,  
  y_model = NULL,  
  m_var = NULL,  
  x_type = NULL,  
  x_var = NULL,  
  TVC = NULL,  
  decompose = NULL,  
  growth_TIC = NULL,  
  starts = NULL,  
  res_scale = NULL,  
  res_cor = NULL,  
  tries = NULL,  
  OKStatus = 0,  
  jitterD = "runif",  
  loc = 1,  
  scale = 0.25,  
  paramOut = FALSE,  
  names = NULL
)

Arguments

dat  
A wide-format data frame, with each row corresponding to a unique ID. It contains the observed variables with repeated measurements and occasions for each longitudinal process, and time-invariant covariates (TICs) if any.

prop_starts  
A numeric vector of user-specified initial component proportions of latent classes.

sub_Model  
A string that specifies the sub-model for latent classes. Supported sub-models include "LGCM" (for latent growth curve models), "LCM" (for latent change score models), "TVM" (for latent growth curve models or latent change score models with a time-varying covariate), "MGM" (for multivariate latent growth curve models or latent change score models), and "MED" (for longitudinal mediation models).
cluster_TIC: A string or character vector representing the column name(s) for time-invariant covariate(s) indicating cluster formations. Default is NULL, indicating no such time-invariant covariates are present in the model.

t_var: A string specifying the prefix of the column names corresponding to the time variable for each study wave. This applies when sub_Model is "LGCM", "LCSM" or "TVC". For sub_Model being "MGM" or "MED", t_var should be a string vector where each element corresponds to the time variable prefix for each respective longitudinal process.

records: A numeric vector denoting the indices of the observed study waves. This applies when sub_Model is "LGCM", "LCSM" or "TVC". For sub_Model being "MGM" or "MED", records should be a list of numeric vectors, where each vector provides the indices of the observed study waves for each longitudinal process.

y_var: A string defining the prefix of the column names corresponding to the outcome variable for each study wave. This is applicable when sub_Model is not "MGM". For sub_Model being "MGM", y_var should be a string vector where each element corresponds to the prefix of the column names for each outcome variable across the study waves.

curveFun: A string specifying the functional forms of the growth curve(s). Supported options for y_model = "LGCM" include: "linear" (or "LIN"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "bilinear spline" (or "BLS"). Supported options for y_model = "LCSM" include: "nonparametric" (or "NonP"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), and "Jenss-Bayley" (or "JB").

intrinsic: A logical flag indicating whether to build an intrinsically nonlinear longitudinal model. By default, this is NULL as it is unnecessary when sub_Model is "MED".

y_model: A string that specifies how to fit longitudinal outcomes. Supported values are "LGCM" and "LCSM". By default, this is NULL as this argument only requires when sub_Model is "TVC" or "MGM".

m_var: A string that specifies the prefix of the column names corresponding to the mediator variable at each study wave. By default, this is NULL as this argument only requires when sub_Model is "MED".

x_type: A string indicating the type of predictor variable used in the model. Supported values are "baseline" and "longitudinal". By default, this is NULL as this argument only requires when sub_Model is "MED".

x_var: A string specifying the baseline predictor if x_type = "baseline", or the prefix of the column names corresponding to the predictor variable at each study wave if x_type = "longitudinal". By default, this is NULL as this argument only requires when sub_Model is "MED".

TVC: A string that specifies the prefix of the column names corresponding to the time-varying covariate at each time point. By default, this is NULL as this argument only requires when sub_Model is "TVC".

decompose: An integer specifying the decomposition option for temporal states. Supported values include 0 (no decomposition), 1 (decomposition with interval-specific slopes as temporal states), 2 (decomposition with interval-specific changes as temporal states), and 3 (decomposition with change-from-baseline as temporal
states). By default, this is NULL as this argument only requires when sub_Model is "TVC".

growth_TIC A string or character vector of column names of time-invariant covariate(s) accounting for the variability of growth factors if any. Default is NULL, indicating no growth TICs present in the model.

starts A list containing initial values for the parameters. Default is NULL, indicating no user-specified initial values.

res_scale A list where each element is a (vector of) numeric scaling factor(s) for residual variance to calculate the corresponding initial value for a latent class, between 0 and 1 exclusive. By default, this is NULL, as it is unnecessary when the user specifies the initial values using the starts argument.

res_cor A list where each element is a (vector of) numeric initial value(s) for residual correlation in each class. It needs to be specified if the sub_Model is "TVC" (when decompose != 0), "MGM", or "MED". By default, this is NULL, as it is unnecessary when the user specifies the initial values using the starts argument.

tries An integer specifying the number of additional optimization attempts. Default is NULL.

OKStatus An integer (vector) specifying acceptable status codes for convergence. Default is 0.

jitterD A string specifying the distribution for jitter. Supported values are: "runif" (uniform distribution), "rnorm" (normal distribution), and "rcauchy" (Cauchy distribution). Default is "runif".

loc A numeric value representing the location parameter of the jitter distribution. Default is 1.

scale A numeric value representing the scale parameter of the jitter distribution. Default is 0.25.

paramOut A logical flag indicating whether to output the parameter estimates and standard errors. Default is FALSE.

names A character vector specifying parameter names. Default is NULL.

Value

A list containing the fitted latent change score model and, if paramOut = TRUE, a data frame with parameter estimates and standard errors.

References


Examples

OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
data("RMS_dat")
RMS_dat0 <- RMS_dat
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)
RMS_dat0$gx1 <- scale(RMS_dat0$INCOME)
RMS_dat0$gx2 <- scale(RMS_dat0$EDU)

MIX_BLS_LGCM.TIC_r <- getMIX(
  dat = RMS_dat0, prop_starts = c(0.45, 0.55), sub_Model = "LGCM",
  cluster_TIC = NULL, y_var = "M", t_var = "T", records = 1:9,
  curveFun = "BLS", intrinsic = FALSE, res_scale = list(0.3, 0.3)
)
paraBLS.TIC_LGCM.r <- c(
  "alpha0", "alpha1", "alpha2", "knot",
  paste0("psi", c("00", "01", "02", "11", "12", "22")), "residuals",
  paste0("beta1", 0:2), paste0("beta2", 0:2),
  paste0("mux", 1:2), paste0("phi", c("11", "12", "22")),
  "mueta0", "mueta1", "mueta2"
)
set.seed(20191029)
MIX_BLS_LGCM.TIC_r <- getMIX(
  dat = RMS_dat0, prop_starts = c(0.33, 0.34, 0.33), sub_Model = "LGCM",
  cluster_TIC = c("gx1", "gx2"), y_var = "M", t_var = "T", records = 1:9,
  curveFun = "BLS", intrinsic = FALSE, res_scale = list(0.3, 0.3, 0.3),
  growth_TIC = c("ex1", "ex2"), tries = 10, paramOut = TRUE,
  names = paraBLS.TIC_LGCM.r
)
MIX_BLS_LGCM.TIC_r[[2]]
**getPosterior**

*Compute Posterior Probabilities, Cluster Assignments, and Model Entropy for a Longitudinal Mixture Model*

**Description**

This function computes posterior probabilities, cluster assignments, and model entropy for a given mixture model with a predefined number of classes. If the true labels are available, it can also compute the model accuracy.

**Usage**

```r
getPosterior(model, nClass, label = FALSE, cluster_TIC = NULL)
```

**Arguments**

- `model` A fitted `mxModel` object. This is the output from `getMIX()`.
- `nClass` An integer representing the predefined number of latent classes in the model.
- `label` A logical value indicating whether the data contains true labels. Default is `FALSE`.
- `cluster_TIC` A string or character vector representing the column name(s) for time-invariant covariate(s) indicating cluster formations. Default is `NULL`, indicating that no such time-invariant covariates are present in the model.

**Value**

A list containing:

- `prob`: A matrix of posterior probabilities.
- `membership`: A vector indicating class membership based on maximum posterior probability.
- `entropy`: The entropy of the model, a measure of uncertainty in class assignment.
- `accuracy`: The model’s accuracy, if true labels are provided (`label = TRUE`).

**References**

Examples

```r
OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
data("RMS_dat")
RMS_dat0 <- RMS_dat
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)
RMS_dat0$gx1 <- scale(RMS_dat0$INCOME)
RMS_dat0$gx2 <- scale(RMS_dat0$EDU)

set.seed(20191029)
MIX_BLS_LGCM_r <- getMIX(
  dat = RMS_dat0, prop_starts = c(0.33, 0.34, 0.33), sub_Model = "LGCM",
  cluster_TIC = NULL, y_var = "M", t_var = "T", records = 1:9, curveFun = "BLS",
  intrinsic = FALSE, res_scale = list(0.3, 0.3, 0.3), growth_TIC = NULL, tries = 10
)
label1 <- getPosterior(
  model = MIX_BLS_LGCM_r, nClass = 3, label = FALSE, cluster_TIC = NULL
)

set.seed(20191029)
MIX_BLS_LGCM_TIC_r <- getMIX(
  dat = RMS_dat0, prop_starts = c(0.33, 0.34, 0.33), sub_Model = "LGCM",
  cluster_TIC = c("gx1", "gx2"), y_var = "M", t_var = "T", records = 1:9,
  curveFun = "BLS", intrinsic = FALSE, res_scale = list(0.3, 0.3, 0.3),
  growth_TIC = c("ex1", "ex2"), tries = 10
)
label2 <- getPosterior(
  model = MIX_BLS_LGCM_TIC_r, nClass = 3, label = FALSE, cluster_TIC = c("gx1", "gx2")
)
```

---

**getSummary** *Summarize Model Fit Statistics for Fitted Models*

**Description**

This function summarizes the model fit statistics for a list of fitted models. The summary includes the number of parameters, estimated likelihood (-2ll), AIC, BIC, and other relevant statistics.
Usage

getSummary(model_list, HetModels = FALSE)

Arguments

model_list  A list of fitted mxModel objects.
HetModels  A logical flag indicating whether a mixture model or a multiple group model is included in the list. If set to TRUE, the function can also be used for the enumeration process, allowing the determination of the optimal number of latent classes based on model fit statistics such as BIC. The default value is FALSE.

Value

A data frame summarizing model fit statistics (number of parameters, estimated likelihood, AIC, BIC, etc.) for each model.

Examples

OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
# Load ECLS-K (2011) data
data("RMS_dat")
RMS_dat0 <- RMS_dat
# Re-baseline the data so that the estimated initial status is for the starting point of the study
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- RMS_dat0$T1 - baseT
RMS_dat0$T2 <- RMS_dat0$T2 - baseT
RMS_dat0$T3 <- RMS_dat0$T3 - baseT
RMS_dat0$T4 <- RMS_dat0$T4 - baseT
RMS_dat0$T5 <- RMS_dat0$T5 - baseT
RMS_dat0$T6 <- RMS_dat0$T6 - baseT
RMS_dat0$T7 <- RMS_dat0$T7 - baseT
RMS_dat0$T8 <- RMS_dat0$T8 - baseT
RMS_dat0$T9 <- RMS_dat0$T9 - baseT
# Fit bilinear spline growth model with fix knot (non-intrinsically nonlinear model)

BLS_LGCM1 <- getLGCM(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "BLS", intrinsic = FALSE,
  records = 1:9, res_scale = 0.1
)
getSummary(model_list = list(BLS_LGCM1), HetModels = FALSE)
set.seed(20191029)
BLS_LGCM2 <- getMIX(
  dat = RMS_dat0, prop_starts = c(0.45, 0.55), sub_Model = "LGCM", cluster_TIC = NULL,
  y_var = "M", t_var = "T", records = 1:9, curveFun = "BLS", intrinsic = FALSE,
  res_scale = list(0.3, 0.3), growth_TIC = NULL, tries = 10
)
set.seed(20191029)
BLS_LGCM3 <- getMIX(
  dat = RMS_dat0, prop_starts = c(0.33, 0.34, 0.33), sub_Model = "LGCM", cluster_TIC = NULL,
  y_var = "M", t_var = "T", records = 1:9, curveFun = "BLS", intrinsic = FALSE,
  res_scale = list(0.3, 0.3, 0.3), growth_TIC = NULL, tries = 10
)
getTVCmodel

Fit a Latent Growth Curve Model or Latent Change Score Model with Time-varying and Time-invariant Covariates

Description

This function fits a latent growth curve model or latent change score model with a time-varying covariate and potential time-invariant covariates to the provided data. It manages model setup, optimization, and if requested, outputs parameter estimates and standard errors.

Usage

getTVCmodel(
  dat,
  t_var,
  y_var,
  curveFun,
  intrinsic = TRUE,
  records,
  y_model,
  TVC,
  decompose,
  growth_TIC = NULL,
  starts = NULL,
  res_scale = NULL,
  res_cor = NULL,
  tries = NULL,
  OKStatus = 0,
  jitterD = "runif",
  loc = 1,
  scale = 0.25,
  paramOut = FALSE,
  names = NULL
)

Arguments

dat A wide-format data frame, with each row corresponding to a unique ID. It contains the observed variables with repeated measurements (for the longitudinal outcome and time-varying covariates), occasions, and time-invariant covariates (TICs) if any.
getTVCmodel

\textbf{t\_var} \hspace{1em} A string specifying the prefix of the column names corresponding to the time variable at each study wave.

\textbf{y\_var} \hspace{1em} A string specifying the prefix of the column names corresponding to the outcome variable at each study wave.

\textbf{curveFun} \hspace{1em} A string specifying the functional form of the growth curve. Supported options for \texttt{y\_model = "LGCM"} include: "linear" (or "LIN"), "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "bilinear spline" (or "BLS"). Supported options for \texttt{y\_model = "LCSM"} include: "quadratic" (or "QUAD"), "negative exponential" (or "EXP"), "Jenss-Bayley" (or "JB"), and "nonparametric" (or "NonP").

\textbf{intrinsic} \hspace{1em} A logical flag indicating whether to build an intrinsically nonlinear longitudinal model. Default is \texttt{TRUE}.

\textbf{records} \hspace{1em} A numeric vector specifying the indices of the observed study waves.

\textbf{y\_model} \hspace{1em} A string specifying how to fit the longitudinal outcome. Supported values are "LGCM" and "LCSM".

\textbf{TVC} \hspace{1em} A string specifying the prefix of the column names corresponding to the time-varying covariate at each study wave.

\textbf{decompose} \hspace{1em} An integer specifying the decomposition option for temporal states. Supported values include 0 (no decomposition), 1 (decomposition with interval-specific slopes as temporal states), 2 (decomposition with interval-specific changes as temporal states), and 3 (decomposition with change-from-baseline as temporal states).

\textbf{growth\_TIC} \hspace{1em} A string or character vector specifying the column name(s) of time-invariant covariate(s) that account for the variability of growth factors, if any. Default is NULL, indicating no growth TICs present in the model.

\textbf{starts} \hspace{1em} A list containing initial values for the parameters. Default is NULL, indicating no user-specified initial values.

\textbf{res\_scale} \hspace{1em} A numeric value or numeric vector. For a model with \texttt{decompose = 0}, it is a numeric value representing the scaling factor used to calculate the initial value for the residual variance of the longitudinal outcome. In cases where \texttt{decompose != 0}, it is a numeric vector of user-specified scaling factors used to calculate the initial values for the residual variance of both the longitudinal outcome and the time-varying covariate. By default, this is NULL, as it is unnecessary when the user specifies the initial values using the \texttt{starts} argument.

\textbf{res\_cor} \hspace{1em} A numeric value. When \texttt{decompose != 0}, this represents the user-specified residual correlation between the longitudinal outcome and the time-varying covariate, which is used to calculate the corresponding initial value. If \texttt{decompose = 0}, this should be NULL. By default, this is NULL, as it is unnecessary when the user specifies the initial values using the \texttt{starts} argument.

\textbf{tries} \hspace{1em} An integer specifying the number of additional optimization attempts. Default is NULL.

\textbf{OKStatus} \hspace{1em} An integer (vector) specifying acceptable status codes for convergence. Default is 0.
getTVCmodel

jitterD  A string specifying the distribution for jitter. Supported values are: "runif" (uniform distribution), "rnorm" (normal distribution), and "rcauchy" (Cauchy distribution). Default is "runif".

loc  A numeric value representing the location parameter of the jitter distribution. Default is 1.

scale  A numeric value representing the scale parameter of the jitter distribution. Default is 0.25.

paramOut  A logical flag indicating whether to output the parameter estimates and standard errors. Default is FALSE.

names  A character vector specifying parameter names. Default is NULL.

Value

A list containing the fitted model and, if paramOut = TRUE, a data frame with parameter estimates and standard errors.

References


Examples

OpenMx::mxOption(model = NULL, key = "Default optimizer", "CSOLNP", reset = FALSE)
data("RMS_dat")
RMS_dat0 <- RMS_dat
baseT <- RMS_dat0$T1
RMS_dat0$T1 <- (RMS_dat0$T1 - baseT)/12
RMS_dat0$T2 <- (RMS_dat0$T2 - baseT)/12
RMS_dat0$T3 <- (RMS_dat0$T3 - baseT)/12
RMS_dat0$T4 <- (RMS_dat0$T4 - baseT)/12
RMS_dat0$T5 <- (RMS_dat0$T5 - baseT)/12
RMS_dat0$T6 <- (RMS_dat0$T6 - baseT)/12
RMS_dat0$T7 <- (RMS_dat0$T7 - baseT)/12
RMS_dat0$T8 <- (RMS_dat0$T8 - baseT)/12
RMS_dat0$T9 <- (RMS_dat0$T9 - baseT)/12
RMS_dat0$ex1 <- scale(RMS_dat0$Approach_to_Learning)
RMS_dat0$ex2 <- scale(RMS_dat0$Attention_focus)
BL_mean <- mean(RMS_dat0[, "R1"])
BL_var <- var(RMS_dat0[, "R1"])
RMS_dat0$Rs1 <- (RMS_dat0$R1 - BL_mean)/sqrt(BL_var)
RMS_dat0$Rs2 <- (RMS_dat0$R2 - BL_mean)/sqrt(BL_var)
RMS_dat0$Rs3 <- (RMS_dat0$R3 - BL_mean)/sqrt(BL_var)
RMS_dat0$Rs4 <- (RMS_dat0$R4 - BL_mean)/sqrt(BL_var)
RMS_dat0$Rs5 <- (RMS_dat0$R5 - BL_mean)/sqrt(BL_var)
RMS_dat0$Rs6 <- (RMS_dat0$R6 - BL_mean)/sqrt(BL_var)
RMS_dat0$Rs7 <- (RMS_dat0$R7 - BL_mean)/sqrt(BL_var)
RMS_dat

RMS_dat$Rs8 <- (RMS_dat$R8 - BL_mean)/sqrt(BL_var)
RMS_dat$Rs9 <- (RMS_dat$R9 - BL_mean)/sqrt(BL_var)

# Fit bilinear spline latent growth curve model (fixed knot) with a time-varying
# reading ability for mathematics development
BLS_TVC_LGCM1 <- getTVCmodel(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "BLS", intrinsic = FALSE,
  records = 1:9, y_model = "LGCM", TVC = "Rs", decompose = 0, growth_TIC = NULL,
  res_scale = 0.1
)

# Fit negative exponential latent growth curve model (random ratio) with a
# decomposed time-varying reading ability and time-invariant covariates for
# mathematics development
paraEXP_LGCM3.f <- c(
  "Y_alpha0", "Y_alpha1", "Y_alphag",
  paste0("Y_psi", c("00", "01", "0g", "11", "lg", "gg")), "Y_residuals",
  "X_mueta0", "X_mueta1", paste0("X_psi", c("00", "01", "11")),
  paste0("X_abs_rate", 2:8), paste0("X_rel_rate", 1:8), "X_residuals",
  paste0("betaTIC", c(0:1, "g")), paste0("betaTIC", c(0:1, "g")),
  paste0("betaTVC", c(0:1, "g")),
  "muTIC1", "muTIC2", "phiTIC11", "phiTIC12", "phiTIC22",
  "Y_mueta0", "Y_mueta1", "Y_mu_slp_ratio",
  "covBL1", "covBL2", "kappa", "Cov_XYres"
)
set.seed(20191029)
EXP_TVCslp_LGCM3.f <- getTVCmodel(
  dat = RMS_dat0, t_var = "T", y_var = "M", curveFun = "EXP", intrinsic = TRUE,
  records = 1:9, y_model = "LGCM", TVC = "Rs", decompose = 1,
  growth_TIC = c("ex1", "ex2"), res_scale = c(0.1, 0.1),
  res_cor = 0.3, tries = 10, paramOut = TRUE, names = paraEXP_LGCM3.f
)
EXP_TVCslp_LGCM3.f[[2]]

---

RMS_dat  ECLS-K (2011) Sample Dataset for Demonstration

**Description**

A sample dataset extracted from the public-use Early Childhood Longitudinal Study, Kindergarten Class of 2010-11 (ECLS-K:2011) collected by the National Center for Education Statistics (NCES). This dataset is NOT a posting of the original data, and it has been processed and formatted for use in demonstration purposes within this package. For access to the original data, please visit the NCES data products page at [https://nces.ed.gov/ecls/dataproduts.asp](https://nces.ed.gov/ecls/dataproduts.asp).

**Usage**

RMS_dat
Format

A data frame with 500 rows and 49 variables:

ID Identification number.
R1, R2, R3, R4, R5, R6, R7, R8, R9 Reading scores from 9 study waves.
M1, M2, M3, M4, M5, M6, M7, M8, M9 Math scores from 9 study waves.
S2, S3, S4, S5, S6, S7, S8, S9 Science scores from 8 study waves (starting from the second study wave).
T1, T2, T3, T4, T5, T6, T7, T8, T9 Children’s age-in-month at 9 study waves.
SEX Sex of the child.
RACE Race of the child.
LOCALE Locale of the child’s school.
INCOME Family income.
SCHOOL_TYPE Type of the child’s school.
Approach_to_Learning Teacher’s rating on the child’s approach to learning.
Self_control Teacher’s rating on the child’s self-control.
Interpersonal Teacher’s rating on the child’s interpersonal skills.
External_prob_Behavior Teacher’s rating on the child’s external problem behaviors.
Internal_prob_Behavior Teacher’s rating on the child’s internal problem behaviors.
Attention_focus Teacher’s rating on the child’s attention focus.
Inhibitory_ctrl Teacher’s rating on the child’s inhibitory control.
EDU Highest education level between the child’s parents.

Details

The ECLS-K:2011 offers a comprehensive and detailed set of information about children’s early life experiences, focusing on children’s health, development, education, and experiences in the years leading up to kindergarten.

The sample dataset included in this package is used for demonstrating the functionality of the package’s functions and it does not include survey weights. In real analysis, the complex survey weights provided by NCES should be utilized appropriately, for instance, as done in R packages such as lavaan.survey or EdSurvey if not using SEM.

Please note that this data must not be used to attempt to identify respondents. For detailed documentation and proper usage of the ECLS-K:2011 data, please refer to the original source at the National Center for Education Statistics (NCES) website: https://nces.ed.gov/.

Source

https://nces.ed.gov/ecls/dataproducts.asp
Index

* datasets
  RMS_dat, 37

getEstimateStats, 2
getFigure, 4
getIndFS, 7
getLatentKappa, 8
getLCSM, 10
getLGCM, 13
getLRT, 15
getMediation, 17
getMGM, 20
getMGroup, 23
getMIX, 26
getPosterior, 31
getSummary, 32
getTVCmodel, 34

RMS_dat, 37