Package 'fitDTVARMx'

July 22, 2024

Title Fit the Discrete-Time Vector Autoregressive Model

Version 0.0.0.9000

Description Fit the discrete-time vector autoregressive model using the 'OpenMx' package.

URL <https://github.com/jeksterslab/fitDTVARMx>,
<https://jeksterslab.github.io/fitDTVARMx/>

BugReports <https://github.com/jeksterslab/fitDTVARMx/issues>

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Depends R (>= 3.0.0), OpenMx

Imports stats

Suggests knitr, rmarkdown, testthat, simStateSpace

RoxygenNote 7.3.2

NeedsCompilation no

Author Ivan Jacob Agaloos Pesigan [aut, cre, cph]
(<https://orcid.org/0000-0003-4818-8420>)

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

Contents

---

coef.fitdtvaridmx   *Parameter Estimates*

Description

Parameter Estimates

Usage

```
## S3 method for class 'fitdtvaridmx'
coef(object, alpha = FALSE, psi = FALSE, theta = FALSE, ...)
```

Arguments

object	Object of class `fitdtvaridmx`.
alpha	Logical. If `alpha = TRUE`, include estimates of the `alpha` vector, if available. If `alpha = FALSE`, exclude estimates of the `alpha` vector.
psi	Logical. If `psi = TRUE`, include estimates of the `psi` matrix, if available. If `psi = FALSE`, exclude estimates of the `psi` matrix.
theta	Logical. If `theta = TRUE`, include estimates of the `theta` matrix, if available. If `theta = FALSE`, exclude estimates of the `theta` matrix.
...	additional arguments.

Value

Returns a list of vectors of parameter estimates.

Author(s)

Ivan Jacob Agaloos Pesigan

---

coef.fitdtvarmx   *Parameter Estimates*

Description

Parameter Estimates

Usage

```
## S3 method for class 'fitdtvarmx'
coef(object, alpha = FALSE, psi = FALSE, theta = FALSE, ...)
```

Arguments

object	Object of class `fitdtvarmx`.
alpha	Logical. If `alpha = TRUE`, include estimates of the `alpha` vector, if available. If `alpha = FALSE`, exclude estimates of the `alpha` vector.
psi	Logical. If `psi = TRUE`, include estimates of the `psi` matrix, if available. If `psi = FALSE`, exclude estimates of the `psi` matrix.
theta	Logical. If `theta = TRUE`, include estimates of the `theta` matrix, if available. If `theta = FALSE`, exclude estimates of the `theta` matrix.
...	additional arguments.

Value

Returns a vector of parameter estimates.

Author(s)

Ivan Jacob Agaloos Pesigan

FitDTVARIDMx	*Fit the First-Order Discrete-Time Vector Autoregressive Model by ID*

Description

Fit the First-Order Discrete-Time Vector Autoregressive Model by ID

Usage

```
FitDTVARIDMx(
  data,
  observed,
  id,
  alpha_fixed = TRUE,
  alpha_start = NULL,
  alpha_lbound = NULL,
  alpha_ubound = NULL,
  beta_start = NULL,
  beta_lbound = NULL,
  beta_ubound = NULL,
  psi_diag = TRUE,
  psi_start = NULL,
  psi_lbound = NULL,
  psi_ubound = NULL,
  theta_fixed = TRUE,
  theta_start = NULL,
  theta_lbound = NULL,
  theta_ubound = NULL,
```

```
    mu0_fixed = TRUE,
    mu0_start = NULL,
    mu0_lbound = NULL,
    mu0_ubound = NULL,
    sigma0_fixed = TRUE,
    sigma0_diag = TRUE,
    sigma0_start = NULL,
    sigma0_lbound = NULL,
    sigma0_ubound = NULL,
    try = 1000,
    ncores = NULL
)
```

Arguments

data	Data frame. A data frame object of data for potentially multiple subjects that contain a column of subject ID numbers (i.e., an ID variable), and at least one column of observed values.
observed	Character vector. A vector of character strings of the names of the observed variables in the data.
id	Character string. A character string of the name of the ID variable in the data.
alpha_fixed	Logical. If alpha_fixed = TRUE, the dynamic model intercept vector alpha is fixed at zero. If alpha_fixed = FALSE, the dynamic model intercept vector alpha is estimated.
alpha_start	Optional starting values for alpha. If alpha_fixed = TRUE, alpha_start will be used as fixed values. If alpha_fixed = FALSE, alpha_start will be used as starting values.
alpha_lbound	Optional lower bound for alpha. Ignored if alpha_fixed = TRUE.
alpha_ubound	Optional upper bound for alpha. Ignored if alpha_fixed = TRUE.
beta_start	Numeric matrix. Optional starting values for beta.
beta_lbound	Numeric matrix. Optional lower bound for beta.
beta_ubound	Numeric matrix. Optional upper bound for beta.
psi_diag	Logical. If psi_diag = TRUE, psi is a diagonal matrix.
psi_start	Numeric matrix. Optional starting values for psi.
psi_lbound	Numeric matrix. Optional lower bound for psi.
psi_ubound	Optional upper bound for psi.
theta_fixed	Logical. If theta_fixed = TRUE, the measurement error matrix theta is fixed to zero. If theta_fixed = FALSE, estimate the diagonal measurement error matrix theta.
theta_start	Optional starting values for theta. Ignored if theta_fixed = TRUE.
theta_lbound	Optional lower bound for theta. Ignored if theta_fixed = TRUE.
theta_ubound	Optional upper bound for theta. Ignored if theta_fixed = TRUE.
mu0_fixed	Logical. If mu0_fixed = TRUE, initial mean vector mu0 is fixed. If mu0_fixed = FALSE, initial mean vector mu0 is estimated.

mu0_start	Optional starting values for mu0. If mu0_fixed = TRUE, mu0_start will be used as fixed values. If mu0_fixed = FALSE, mu0_start will be used as starting values.
mu0_lbound	Optional lower bound for mu0. Ignored if mu0_fixed = TRUE.
mu0_ubound	Optional upper bound for mu0. Ignored if mu0_fixed = TRUE.
sigma0_fixed	Logical. If sigma0_fixed = TRUE, initial mean vector sigma0 is fixed. If sigma0_fixed = FALSE, initial mean vector sigma0 is estimated.
sigma0_diag	Logical. If sigma0_diag = TRUE, sigma0 is a diagonal matrix.
sigma0_start	Optional starting values for sigma0. If sigma0_fixed = TRUE, sigma0_start will be used as fixed values. If sigma0_fixed = FALSE, sigma0_start will be used as starting values.
sigma0_lbound	Optional lower bound for sigma0. Ignored if sigma0_fixed = TRUE.
sigma0_ubound	Optional upper bound for sigma0. Ignored if sigma0_fixed = TRUE.
try	Positive integer. Number of extra optimization tries.
ncores	Positive integer. Number of cores to use.

Value

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

call Function call.

args List of function arguments.

fun Function used ("FitDTVARIDMx").

output A list of fitted OpenMx models.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Hunter, M. D. (2017). State space modeling in an open source, modular, structural equation modeling environment. *Structural Equation Modeling: A Multidisciplinary Journal*, *25*(2), 307–324. doi:10.1080/10705511.2017.1369354

Neale, M. C., Hunter, M. D., Pritikin, J. N., Zahery, M., Brick, T. R., Kirkpatrick, R. M., Estabrook, R., Bates, T. C., Maes, H. H., & Boker, S. M. (2015). OpenMx 2.0: Extended structural equation and statistical modeling. *Psychometrika*, *81*(2), 535–549. doi:10.1007/s1133601494358

See Also

Other DTVAR Functions: FitDTVARMx()

Examples

```
## Not run:
# Generate data using the simStateSpace package-----------------------------
set.seed(42)
beta_mu <- matrix(
  data = c(
    0.7, 0.5, -0.1,
    0.0, 0.6, 0.4,
    0, 0, 0.5
  ),
  nrow = 3
)
beta_sigma <- diag(3 * 3)
beta <- simStateSpace::SimBetaN(
  n = 5,
  beta = beta_mu,
  vcov_beta_vec_l = t(chol(beta_sigma))
)
sim <- simStateSpace::SimSSMVARIVary(
  n = 5,
  time = 100,
  mu0 = list(rep(x = 0, times = 3)),
  sigma0_l = list(t(chol(diag(3)))),
  alpha = list(rep(x = 0, times = 3)),
  beta = beta,
  psi_l = list(t(chol(diag(3))))
)
data <- as.data.frame(sim)

# Fit the model-------------------------------------------------------------
library(fitDTVARMx)
fit <- FitDTVARIDMx(
  data = data,
  observed = c("y1", "y2", "y3"),
  id = "id"
)
print(fit)
summary(fit)
coef(fit)
vcov(fit)

## End(Not run)
```

FitDTVARMx   *Fit the First-Order Discrete-Time Vector Autoregressive Model*

Description

Fit the First-Order Discrete-Time Vector Autoregressive Model

Usage

```
FitDTVARMx(
  data,
  observed,
  id,
  alpha_fixed = TRUE,
  alpha_start = NULL,
  alpha_lbound = NULL,
  alpha_ubound = NULL,
  beta_start = NULL,
  beta_lbound = NULL,
  beta_ubound = NULL,
  psi_diag = TRUE,
  psi_start = NULL,
  psi_lbound = NULL,
  psi_ubound = NULL,
  theta_fixed = TRUE,
  theta_start = NULL,
  theta_lbound = NULL,
  theta_ubound = NULL,
  mu0_fixed = TRUE,
  mu0_start = NULL,
  mu0_lbound = NULL,
  mu0_ubound = NULL,
  sigma0_fixed = TRUE,
  sigma0_diag = TRUE,
  sigma0_start = NULL,
  sigma0_lbound = NULL,
  sigma0_ubound = NULL,
  try = 1000,
  ncores = NULL
)
```

Arguments

data	Data frame. A data frame object of data for potentially multiple subjects that contain a column of subject ID numbers (i.e., an ID variable), and at least one column of observed values.
observed	Character vector. A vector of character strings of the names of the observed variables in the data.
id	Character string. A character string of the name of the ID variable in the data.
alpha_fixed	Logical. If alpha_fixed = TRUE, the dynamic model intercept vector alpha is fixed at zero. If alpha_fixed = FALSE, the dynamic model intercept vector alpha is estimated.
alpha_start	Optional starting values for alpha. If alpha_fixed = TRUE, alpha_start will be used as fixed values. If alpha_fixed = FALSE, alpha_start will be used as starting values.

alpha_lbound	Optional lower bound for alpha. Ignored if alpha_fixed = TRUE.
alpha_ubound	Optional upper bound for alpha. Ignored if alpha_fixed = TRUE.
beta_start	Numeric matrix. Optional starting values for beta.
beta_lbound	Numeric matrix. Optional lower bound for beta.
beta_ubound	Numeric matrix. Optional upper bound for beta.
psi_diag	Logical. If psi_diag = TRUE, psi is a diagonal matrix.
psi_start	Numeric matrix. Optional starting values for psi.
psi_lbound	Numeric matrix. Optional lower bound for psi.
psi_ubound	Optional upper bound for psi.
theta_fixed	Logical. If theta_fixed = TRUE, the measurement error matrix theta is fixed to zero. If theta_fixed = FALSE, estimate the diagonal measurement error matrix theta.
theta_start	Optional starting values for theta. Ignored if theta_fixed = TRUE.
theta_lbound	Optional lower bound for theta. Ignored if theta_fixed = TRUE.
theta_ubound	Optional upper bound for theta. Ignored if theta_fixed = TRUE.
mu0_fixed	Logical. If mu0_fixed = TRUE, initial mean vector mu0 is fixed. If mu0_fixed = FALSE, initial mean vector mu0 is estimated.
mu0_start	Optional starting values for mu0. If mu0_fixed = TRUE, mu0_start will be used as fixed values. If mu0_fixed = FALSE, mu0_start will be used as starting values.
mu0_lbound	Optional lower bound for mu0. Ignored if mu0_fixed = TRUE.
mu0_ubound	Optional upper bound for mu0. Ignored if mu0_fixed = TRUE.
sigma0_fixed	Logical. If sigma0_fixed = TRUE, initial mean vector sigma0 is fixed. If sigma0_fixed = FALSE, initial mean vector sigma0 is estimated.
sigma0_diag	Logical. If sigma0_diag = TRUE, sigma0 is a diagonal matrix.
sigma0_start	Optional starting values for sigma0. If sigma0_fixed = TRUE, sigma0_start will be used as fixed values. If sigma0_fixed = FALSE, sigma0_start will be used as starting values.
sigma0_lbound	Optional lower bound for sigma0. Ignored if sigma0_fixed = TRUE.
sigma0_ubound	Optional upper bound for sigma0. Ignored if sigma0_fixed = TRUE.
try	Positive integer. Number of extra optimization tries.
ncores	Positive integer. Number of cores to use.

Value

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

call Function call.

args List of function arguments.

fun Function used ("FitDTVARMx").

output A fitted OpenMx model.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Hunter, M. D. (2017). State space modeling in an open source, modular, structural equation modeling environment. *Structural Equation Modeling: A Multidisciplinary Journal*, *25*(2), 307–324. doi:10.1080/10705511.2017.1369354

Neale, M. C., Hunter, M. D., Pritikin, J. N., Zahery, M., Brick, T. R., Kirkpatrick, R. M., Estabrook, R., Bates, T. C., Maes, H. H., & Boker, S. M. (2015). OpenMx 2.0: Extended structural equation and statistical modeling. *Psychometrika*, *81*(2), 535–549. doi:10.1007/s1133601494358

See Also

Other DTVAR Functions: `FitDTVARIDMx()`

Examples

```
## Not run:
# Generate data using the simStateSpace package-----------------------------
set.seed(42)
sim <- simStateSpace::SimSSMVARFixed(
  n = 5,
  time = 100,
  mu0 = rep(x = 0, times = 3),
  sigma0_l = t(chol(diag(3))),
  alpha = rep(x = 0, times = 3),
  beta = matrix(
    data = c(
      0.7, 0.5, -0.1,
      0.0, 0.6, 0.4,
      0, 0, 0.5
    ),
    nrow = 3
  ),
  psi_l = t(chol(diag(3)))
)
data <- as.data.frame(sim)

# Fit the model-------------------------------------------------------------
library(fitDTVARMx)
fit <- FitDTVARMx(
  data = data,
  observed = c("y1", "y2", "y3"),
  id = "id"
)
print(fit)
summary(fit)
coef(fit)
vcov(fit)
```

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

---

print.fitdtvaridmx *Print Method for Object of Class* fitdtvaridmx

---

Description

Print Method for Object of Class `fitdtvaridmx`

Usage

```
## S3 method for class 'fitdtvaridmx'
print(x, means = TRUE, ...)
```

Arguments

x	an object of class `fitdtvaridmx`.
means	Logical. If `means = TRUE`, return means. Otherwise, the function returns raw estimates.
...	further arguments.

Author(s)

Ivan Jacob Agaloos Pesigan

---

print.fitdtvarmx *Print Method for Object of Class* fitdtvarmx

---

Description

Print Method for Object of Class `fitdtvarmx`

Usage

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

Arguments

x	an object of class `fitdtvarmx`.
...	further arguments.

Author(s)

Ivan Jacob Agaloos Pesigan

summary.fitdtvaridmx   *Summary Method for Object of Class* fitdtvaridmx

Description

Summary Method for Object of Class fitdtvaridmx

Usage

```
## S3 method for class 'fitdtvaridmx'
summary(object, means = TRUE, ...)
```

Arguments

object	an object of class fitdtvaridmx.
means	Logical. If means = TRUE, return means. Otherwise, the function returns raw estimates.
...	further arguments.

Author(s)

Ivan Jacob Agaloos Pesigan

summary.fitdtvarmx   *Summary Method for Object of Class* fitdtvarmx

Description

Summary Method for Object of Class fitdtvarmx

Usage

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

Arguments

object	an object of class fitdtvarmx.
...	further arguments.

Author(s)

Ivan Jacob Agaloos Pesigan

vcov.fitdtvaridmx	*Sampling Covariance Matrix of the Parameter Estimates*

Description

Sampling Covariance Matrix of the Parameter Estimates

Usage

```
## S3 method for class 'fitdtvaridmx'
vcov(object, alpha = FALSE, psi = FALSE, theta = FALSE, ...)
```

Arguments

object	Object of class `fitdtvaridmx`.
alpha	Logical. If `alpha = TRUE`, include estimates of the `alpha` vector, if available. If `alpha = FALSE`, exclude estimates of the `alpha` vector.
psi	Logical. If `psi = TRUE`, include estimates of the `psi` matrix, if available. If `psi = FALSE`, exclude estimates of the `psi` matrix.
theta	Logical. If `theta = TRUE`, include estimates of the `theta` matrix, if available. If `theta = FALSE`, exclude estimates of the `theta` matrix.
...	additional arguments.

Value

Returns a list of sampling variance-covariance matrices.

Author(s)

Ivan Jacob Agaloos Pesigan

vcov.fitdtvarmx	*Sampling Covariance Matrix of the Parameter Estimates*

Description

Sampling Covariance Matrix of the Parameter Estimates

Usage

```
## S3 method for class 'fitdtvarmx'
vcov(object, alpha = FALSE, psi = FALSE, theta = FALSE, ...)
```

Arguments

object	Object of class fitdtvarmx.
alpha	Logical. If alpha = TRUE, include estimates of the alpha vector, if available. If alpha = FALSE, exclude estimates of the alpha vector.
psi	Logical. If psi = TRUE, include estimates of the psi matrix, if available. If psi = FALSE, exclude estimates of the psi matrix.
theta	Logical. If theta = TRUE, include estimates of the theta matrix, if available. If theta = FALSE, exclude estimates of the theta matrix.
...	additional arguments.

Value

Returns a list of sampling variance-covariance matrices.

Author(s)

Ivan Jacob Agaloos Pesigan

Index