

Package ‘fitOU’

October 26, 2023

Title The Ornstein–Uhlenbeck Model

Version 0.0.0.9000

Description Fit the Ornstein–Uhlenbeck model using the 'dynr' package.

URL <https://github.com/ijapesigan/fitOU>,
<https://ijapesigan.github.io/fitOU/>

BugReports <https://github.com/ijapesigan/fitOU/issues>

License GPL (>= 3)

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Depends R (>= 3.5.0)

Imports stats, dynr, parallel, pbapply

Suggests knitr, rmarkdown, testthat

RoxygenNote 7.2.3

NeedsCompilation no

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

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

R topics documented:

bivariate_ou	2
FitOU	2
FitOUID	4

Index	7
-------	---

bivariate_ou	<i>Bivariate Ornstein–Uhlenbeck Model Data</i>
--------------	--

Description

Bivariate Ornstein–Uhlenbeck Model Data

Usage

```
bivariate_ou
```

Format

A dataframe with 10000 rows and 4 columns (y1, y2, id, and time) generated from the bivariate Ornstein–Uhlenbeck model from Chow et al. (2023).

References

Chow, S.-M., Losardo, D., Park, J., & Molenaar, P. C. M. (2023). Continuous-time dynamic models: Connections to structural equation models and other discrete-time models. In R. H. Hoyle (Ed.), Handbook of structural equation modeling (2nd ed.). The Guilford Press.

FitOU	<i>Fit the Ornstein–Uhlenbeck Model</i>
-------	---

Description

This is a wrapper function that makes fitting the Ornstein–Uhlenbeck model convenient using the dynr package.

Usage

```
FitOU(
  data,
  observed,
  id,
  time,
  mu0 = NULL,
  sigma0 = NULL,
  mu_start = NULL,
  phi_start = NULL,
  sigma_start = NULL,
  theta_start = NULL,
  center = FALSE,
  ...
)
```

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), a column indicating subject-specific measurement occasions (i.e., a TIME variable), 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.
time	Character string. A character string of the name of the TIME variable in the data.
mu0	Numeric vector. Mean of initial latent variable values ($\mu_{\eta 0}$). If mu0 = NULL, a vector of zeros is used.
sigma0	Numeric matrix. Covariance matrix of initial latent variable values ($\Sigma_{\eta 0}$). If sigma0 = NULL, an identity matrix is used.
mu_start	Numeric vector. Starting values of the mu vector, that is, the long-term mean or equilibrium level. If mu_start = NULL, a vector means of the observed variables is used.
phi_start	Numeric matrix. Starting values of the phi matrix, that is, the rate of mean reversion, determining how quickly the variable returns to its mean. If phi_start = NULL, a matrix of zeros is used.
sigma_start	Numeric matrix. Starting values of the sigma matrix, that is, the matrix of volatility or randomness in the process. If sigma_start = NULL, an identity matrix is used.
theta_start	Numeric matrix. Starting values of the theta matrix, that is, the measurement error covariance matrix (Θ). If theta_start = NULL, an identity matrix is used.
center	Logical. If center = TRUE, mean center by id.
...	Additional arguments to pass to dynr::dynr.cook() .

Details

The measurement model is given by

$$\mathbf{y}_{i,t} = \boldsymbol{\nu} + \mathbf{\Lambda}\boldsymbol{\eta}_{i,t} + \boldsymbol{\varepsilon}_{i,t} \quad \text{with} \quad \boldsymbol{\varepsilon}_{i,t} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Theta})$$

where $\mathbf{y}_{i,t}$, $\boldsymbol{\eta}_{i,t}$, and $\boldsymbol{\varepsilon}_{i,t}$ are random variables and $\boldsymbol{\nu}$, $\mathbf{\Lambda}$, and $\boldsymbol{\Theta}$ are model parameters. $\mathbf{y}_{i,t}$ is a vector of observed random variables at time t and individual i , $\boldsymbol{\eta}_{i,t}$ is a vector of latent random variables at time t and individual i , and $\boldsymbol{\varepsilon}_{i,t}$ is a vector of random measurement errors at time t and individual i , while $\boldsymbol{\nu}$ is a vector of intercept, $\mathbf{\Lambda}$ is a matrix of factor loadings, and $\boldsymbol{\Theta}$ is the covariance matrix of $\boldsymbol{\varepsilon}$.

The dynamic structure is given by

$$d\boldsymbol{\eta}_{i,t} = \boldsymbol{\Phi}(\boldsymbol{\mu} - \boldsymbol{\eta}_{i,t})dt + \boldsymbol{\Sigma}^{\frac{1}{2}}d\mathbf{W}_{i,t}$$

where $\boldsymbol{\mu}$ is the long-term mean or equilibrium level, $\boldsymbol{\Phi}$ is the rate of mean reversion, determining how quickly the variable returns to its mean, $\boldsymbol{\Sigma}$ is the matrix of volatility or randomness in the process, and $d\mathbf{W}$ is a Wiener process or Brownian motion, which represents random fluctuations.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Chow, S.-M., Losardo, D., Park, J., & Molenaar, P. C. M. (2023). Continuous-time dynamic models: Connections to structural equation models and other discrete-time models. In R. H. Hoyle (Ed.), *Handbook of structural equation modeling* (2nd ed.). The Guilford Press.

Ou, L., Hunter, M. D., & Chow, S.-M. (2019). What's for dynr: A package for linear and nonlinear dynamic modeling in R. *The R Journal*, 11(1), 91. doi:10.32614/rj2019012

Uhlenbeck, G. E., & Ornstein, L. S. (1930). On the theory of the brownian motion. *Physical Review*, 36(5), 823–841. doi:doi.org/10.1103/physrev.36.823

See Also

Other Fit Ornstein–Uhlenbeck Model Functions: [FitOUID\(\)](#)

Examples

```
## Not run:
FitOU(
  data = bivariate_ou,
  observed = c("y1", "y2"),
  id = "id",
  time = "time",
  verbose = FALSE
)

## End(Not run)
```

FitOUID

Fit the Ornstein–Uhlenbeck Model for Each Individual

Description

This is a wrapper function that makes fitting the Ornstein–Uhlenbeck model for each individual convenient using the dynr package.

Usage

```
FitOUID(
  data,
  observed,
  id,
  time,
  mu0 = NULL,
  sigma0 = NULL,
```

```

    mu_start = NULL,
    phi_start = NULL,
    sigma_start = NULL,
    theta_start = NULL,
    ...,
    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), a column indicating subject-specific measurement occasions (i.e., a TIME variable), 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.
time	Character string. A character string of the name of the TIME variable in the data.
mu0	Numeric vector. Mean of initial latent variable values ($\mu_{\eta 0}$). If mu0 = NULL, a vector of zeros is used.
sigma0	Numeric matrix. Covariance matrix of initial latent variable values ($\Sigma_{\eta 0}$). If sigma0 = NULL, an identity matrix is used.
mu_start	Numeric vector. Starting values of the mu vector, that is, the long-term mean or equilibrium level. If mu_start = NULL, a vector means of the observed variables is used.
phi_start	Numeric matrix. Starting values of the phi matrix, that is, the rate of mean reversion, determining how quickly the variable returns to its mean. If phi_start = NULL, a matrix of zeros is used.
sigma_start	Numeric matrix. Starting values of the sigma matrix, that is, the matrix of volatility or randomness in the process. If sigma_start = NULL, an identity matrix is used.
theta_start	Numeric matrix. Starting values of the theta matrix, that is, the measurement error covariance matrix (Θ). If theta_start = NULL, an identity matrix is used.
...	Additional arguments to pass to dynr::dynr.cook() .
ncores	Positive integer. Number of cores to use.

Details

The measurement model is given by

$$\mathbf{y}_{i,t} = \boldsymbol{\nu} + \mathbf{\Lambda}\boldsymbol{\eta}_{i,t} + \boldsymbol{\varepsilon}_{i,t} \quad \text{with} \quad \boldsymbol{\varepsilon}_{i,t} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Theta})$$

where $\mathbf{y}_{i,t}$, $\boldsymbol{\eta}_{i,t}$, and $\boldsymbol{\varepsilon}_{i,t}$ are random variables and $\boldsymbol{\nu}$, $\mathbf{\Lambda}$, and $\boldsymbol{\Theta}$ are model parameters. $\mathbf{y}_{i,t}$ is a vector of observed random variables at time t and individual i , $\boldsymbol{\eta}_{i,t}$ is a vector of latent random variables at time t and individual i , and $\boldsymbol{\varepsilon}_{i,t}$ is a vector of random measurement errors at time t

and individual i , while ν is a vector of intercept, Λ is a matrix of factor loadings, and Θ is the covariance matrix of ϵ .

The dynamic structure is given by

$$d\eta_{i,t} = \Phi (\mu - \eta_{i,t}) dt + \Sigma^{\frac{1}{2}} dW_{i,t}$$

where μ is the long-term mean or equilibrium level, Φ is the rate of mean reversion, determining how quickly the variable returns to its mean, Σ is the matrix of volatility or randomness in the process, and dW is a Wiener process or Brownian motion, which represents random fluctuations.

Author(s)

Ivan Jacob Agaloos Pesigan

References

- Chow, S.-M., Losardo, D., Park, J., & Molenaar, P. C. M. (2023). Continuous-time dynamic models: Connections to structural equation models and other discrete-time models. In R. H. Hoyle (Ed.), *Handbook of structural equation modeling* (2nd ed.). The Guilford Press.
- Ou, L., Hunter, M. D., & Chow, S.-M. (2019). What's for dynr: A package for linear and nonlinear dynamic modeling in R. *The R Journal*, 11(1), 91. doi:10.32614/rj2019012
- Uhlenbeck, G. E., & Ornstein, L. S. (1930). On the theory of the brownian motion. *Physical Review*, 36(5), 823–841. doi:doi.org/10.1103/physrev.36.823

See Also

Other Fit Ornstein–Uhlenbeck Model Functions: [FitOU\(\)](#)

Examples

```
## Not run:
FitOUID(
  data = bivariate_ou,
  observed = c("y1", "y2"),
  id = "id",
  time = "time",
  verbose = FALSE
)

## End(Not run)
```

Index

* **Fit Ornstein–Uhlenbeck Model Functions**

FitOU, [2](#)

FitOUID, [4](#)

* **data**

bivariate_ou, [2](#)

* **fitOU**

bivariate_ou, [2](#)

FitOU, [2](#)

FitOUID, [4](#)

* **fit**

FitOU, [2](#)

FitOUID, [4](#)

bivariate_ou, [2](#)

dynr::dynr.cook(), [3](#), [5](#)

FitOU, [2](#), [6](#)

FitOUID, [4](#), [4](#)