

Package ‘fitOU’

November 7, 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

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
DataOU	2
FitOU	3
ModelOU	5

Index	8
--------------	----------

bivariate_ou

Bivariate Ornstein–Uhlenbeck Model Data

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.

DataOU

Preliminary Data Preparation

Description

Preliminary Data Preparation

Usage

```
DataOU(
  data,
  observed,
  id,
  time,
  insert_na = FALSE,
  center = FALSE,
  scale = FALSE,
  initial_na = TRUE
)
```

Arguments

<code>data</code>	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.
<code>observed</code>	Character vector. A vector of character strings of the names of the observed variables in the data.
<code>id</code>	Character string. A character string of the name of the ID variable in the data.
<code>time</code>	Character string. A character string of the name of the TIME variable in the data.
<code>insert_na</code>	Logical. Insert NA to observed variables for existing time points.
<code>center</code>	Logical. If <code>center = TRUE</code> , mean center by id.
<code>scale</code>	Logical. If <code>scale = TRUE</code> , standardize by id.
<code>initial_na</code>	Logical. Iteratively remove rows where any observed variable for the first time point has NA.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

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

Examples

```
data <- DataOU(
  data = bivariate_ou,
  observed = c("y1", "y2"),
  id = "id",
  time = "time"
)
summary(data)
```

FitOU

Fit the Ornstein–Uhlenbeck Model

Description

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

Usage

```
FitOU(model, retry = NULL, ...)
```

Arguments

model	Output of <code>ModelOU()</code> .
retry	Positive integer. Maximum number of reruns.
...	Additional arguments to pass to <code>dynr::dynr.cook()</code> .

Details

The measurement model is given by

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

where $\mathbf{y}_{i,t}$, $\boldsymbol{\eta}_{i,t}$, and $\boldsymbol{\varepsilon}_{i,t}$ are random variables and $\boldsymbol{\nu}$, $\boldsymbol{\Lambda}$, 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, $\boldsymbol{\Lambda}$ 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: `DataOU()`, `ModelOU()`

Examples

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

## End(Not run)
```

ModelOU*Specify the Ornstein–Uhlenbeck Model*

Description

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

Usage

```
ModelOU(
  data,
  observed,
  id,
  time,
  mu0 = NULL,
  sigma0 = NULL,
  mu_start = NULL,
  phi_start = NULL,
  sigma_start = NULL,
  theta_start = NULL,
  sigma_diag = FALSE,
  outfile = paste0(tempfile(), ".c")
)
```

Arguments

<code>data</code>	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.
<code>observed</code>	Character vector. A vector of character strings of the names of the observed variables in the data.
<code>id</code>	Character string. A character string of the name of the ID variable in the data.
<code>time</code>	Character string. A character string of the name of the TIME variable in the data.
<code>mu0</code>	Numeric vector. Mean of initial latent variable values ($\mu_{\eta 0}$). If <code>mu0</code> = NULL, a vector of zeros is used.
<code>sigma0</code>	Numeric matrix. Covariance matrix of initial latent variable values ($\Sigma_{\eta 0}$). If <code>sigma0</code> = NULL, an identity matrix is used.
<code>mu_start</code>	Numeric vector. Starting values of the mu vector, that is, the long-term mean or equilibrium level. If <code>mu_start</code> = NULL, a vector means of the observed variables is used.
<code>phi_start</code>	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 <code>phi_start</code> = NULL, a matrix of zeros is used.
<code>sigma_start</code>	Numeric matrix. Starting values of the sigma matrix, that is, the matrix of volatility or randomness in the process. If <code>sigma_start</code> = NULL, an identity matrix is used.
<code>theta_start</code>	Numeric matrix. Starting values of the theta matrix, that is, the measurement error covariance matrix (Θ). If <code>theta_start</code> = NULL, an identity matrix is used.
<code>sigma_diag</code>	Logical. If <code>sigma_diag</code> = TRUE, estimate only the diagonals of Σ .
<code>outfile</code>	A character string of the name of the output C script of model functions to be compiled for parameter estimation.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

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

Examples

```
## Not run:
data <- DataOU(
  data = bivariate_ou,
  observed = c("y1", "y2"),
  id = "id",
  time = "time"
)
```

```
ModelOU(  
  data = data,  
  observed = c("y1", "y2"),  
  id = "id",  
  time = "time"  
)  
  
## End(Not run)
```

Index

* **Fit Ornstein–Uhlenbeck Model Functions**

DataOU, [2](#)

FitOU, [3](#)

ModelOU, [5](#)

* **data**

bivariate_ou, [2](#)

* **fitOU**

bivariate_ou, [2](#)

DataOU, [2](#)

FitOU, [3](#)

ModelOU, [5](#)

* **fit**

DataOU, [2](#)

FitOU, [3](#)

ModelOU, [5](#)

bivariate_ou, [2](#)

DataOU, [2](#), [4](#), [6](#)

dynr::dynr.cook(), [4](#)

FitOU, [3](#), [3](#), [6](#)

ModelOU, [3](#), [4](#), [5](#)

ModelOU(), [4](#)