

Package ‘dynUtils’

May 1, 2024

Title Dynamic Modeling Utilities

Version 0.9.1

Description Utility functions for data preparation when fitting dynamic models.

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

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

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Depends R (>= 3.5.0)

Suggests knitr, rmarkdown, testthat, simStateSpace

RoxygenNote 7.3.1

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

DeleteInitialNA	2
InitialNA	3
InsertNA	5
print.dynutillist	7
ScaleByID	8
SubsetByID	10
Index	12

DeleteInitialNA

*Delete for NAs in Initial Row By ID***Description**

The function removes the initial row by ID if it contains missing values. This process is repeated recursively until the first row per ID no longer has missing observations.

Usage

```
DeleteInitialNA(data, id, time, observed, covariates = 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.
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.
observed	Character vector. A vector of character strings of the names of the observed variables in the data.
covariates	Character vector. A vector of character strings of the names of the covariates in the data.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of individuals is large.

Value

Returns a data frame.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Dynamic Modeling Utility Functions: [InitialNA\(\)](#), [InsertNA\(\)](#), [ScaleByID\(\)](#), [SubsetByID\(\)](#)

Examples

```

# prepare parameters
set.seed(42)
## number of individuals
n <- 5
## time points
time <- 5
## dynamic structure
p <- 3
mu0 <- rep(x = 0, times = p)
sigma0 <- 0.001 * diag(p)
sigma0_l <- t(chol(sigma0))
alpha <- rep(x = 0, times = p)
beta <- 0.50 * diag(p)
psi <- 0.001 * diag(p)
psi_l <- t(chol(psi))

library(simStateSpace)
ssm <- SimSSMVARFixed(
  n = n,
  time = time,
  mu0 = mu0,
  sigma0_l = sigma0_l,
  alpha = alpha,
  beta = beta,
  psi_l = psi_l,
  type = 0
)
data <- as.data.frame(ssm)
# Replace first row with NA
data[1, paste0("y", 1:p)] <- NA
DeleteInitialNA(
  data = data,
  id = "id",
  time = "time",
  observed = paste0("y", 1:p),
)

```

InitialNA

*Check for NAs in Initial Row By ID***Description**

The function checks if there are missing values for the initial row by ID.

Usage

```
InitialNA(data, id, time, observed, covariates = NULL, ncores = NULL)
```

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>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>observed</code>	Character vector. A vector of character strings of the names of the observed variables in the data.
<code>covariates</code>	Character vector. A vector of character strings of the names of the covariates in the data.
<code>ncores</code>	Positive integer. Number of cores to use. If <code>ncores = NULL</code> , use a single core. Consider using multiple cores when number of individuals is large.

Value

Returns a vector of ID numbers where the initial row has any missing value.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Dynamic Modeling Utility Functions: [DeleteInitialNA\(\)](#), [InsertNA\(\)](#), [ScaleByID\(\)](#), [SubsetByID\(\)](#)

Examples

```
# prepare parameters
set.seed(42)
## number of individuals
n <- 5
## time points
time <- 5
## dynamic structure
p <- 3
mu0 <- rep(x = 0, times = p)
sigma0 <- 0.001 * diag(p)
sigma0_l <- t(chol(sigma0))
alpha <- rep(x = 0, times = p)
beta <- 0.50 * diag(p)
psi <- 0.001 * diag(p)
psi_l <- t(chol(psi))

library(simStateSpace)
ssm <- SimSSMVARFixed(
  n = n,
```

```

    time = time,
    mu0 = mu0,
    sigma0_l = sigma0_l,
    alpha = alpha,
    beta = beta,
    psi_l = psi_l,
    type = 0
  )
data <- as.data.frame(ssm)
# Replace first row with NA
data[1, paste0("y", 1:p)] <- NA
InitialNA(
  data = data,
  id = "id",
  time = "time",
  observed = paste0("y", 1:p),
)

```

InsertNA

*Insert NAs for Missing Observations***Description**

The function creates a sequence of time values. It starts with the smallest time value as the starting point and the largest time value as the endpoint. The sequence is incremented by `delta_t`. This new sequence is combined with the existing empirical time values. For any specific time value where there are no observations, NAs are inserted.

Usage

```
InsertNA(data, id, time, observed, covariates = NULL, delta_t, ncores = NULL)
```

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>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>observed</code>	Character vector. A vector of character strings of the names of the observed variables in the data.
<code>covariates</code>	Character vector. A vector of character strings of the names of the covariates in the data.
<code>delta_t</code>	Positive number. Time interval.
<code>ncores</code>	Positive integer. Number of cores to use. If <code>ncores = NULL</code> , use a single core. Consider using multiple cores when number of individuals is large.

Value

Returns a data frame.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Dynamic Modeling Utility Functions: [DeleteInitialNA\(\)](#), [InitialNA\(\)](#), [ScaleByID\(\)](#), [SubsetByID\(\)](#)

Examples

```
# prepare parameters
set.seed(42)
## number of individuals
n <- 5
## time points
time <- 5
## dynamic structure
p <- 3
mu0 <- rep(x = 0, times = p)
sigma0 <- 0.001 * diag(p)
sigma0_l <- t(chol(sigma0))
alpha <- rep(x = 0, times = p)
beta <- 0.50 * diag(p)
psi <- 0.001 * diag(p)
psi_l <- t(chol(psi))

library(simStateSpace)
ssm <- SimSSMVARFixed(
  n = n,
  time = time,
  mu0 = mu0,
  sigma0_l = sigma0_l,
  alpha = alpha,
  beta = beta,
  psi_l = psi_l,
  type = 0
)
data <- as.data.frame(ssm)
InsertNA(
  data = data,
  id = "id",
  time = "time",
  observed = paste0("y", 1:p),
  delta_t = 0.10
)
```

print.dynutillist	<i>Print Method for Object of Class dynutillist</i>
-------------------	---

Description

Print Method for Object of Class dynutillist

Usage

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

Arguments

x	an object of class dynutillist.
...	further arguments.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# prepare parameters  
set.seed(42)  
## number of individuals  
n <- 5  
## time points  
time <- 5  
## dynamic structure  
p <- 3  
mu0 <- rep(x = 0, times = p)  
sigma0 <- 0.001 * diag(p)  
sigma0_l <- t(chol(sigma0))  
alpha <- rep(x = 0, times = p)  
beta <- 0.50 * diag(p)  
psi <- 0.001 * diag(p)  
psi_l <- t(chol(psi))  
  
library(simStateSpace)  
ssm <- SimSSMVARFixed(  
  n = n,  
  time = time,  
  mu0 = mu0,  
  sigma0_l = sigma0_l,  
  alpha = alpha,  
  beta = beta,  
  psi_l = psi_l,  
  type = 0
```

```

)
data <- as.data.frame(ssm)
out <- SubsetByID(
  data = data,
  id = "id",
  time = "time",
  observed = paste0("y", 1:p)
)
print(out)

```

ScaleByID

Scale by ID

Description

The function scales the data by ID.

Usage

```

ScaleByID(
  data,
  id,
  time,
  observed,
  covariates = NULL,
  scale = TRUE,
  obs_skip = NULL,
  cov_skip = 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.
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.
observed	Character vector. A vector of character strings of the names of the observed variables in the data.
covariates	Character vector. A vector of character strings of the names of the covariates in the data.
scale	Logical. If scale = TRUE, standardize by id. If scale = FALSE, mean center by id.

obs_skip	Character vector. A vector of character strings of the names of the observed variables to skip centering/scaling.
cov_skip	Character vector. A vector of character strings of the names of the covariates to skip centering/scaling.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of individuals is large.

Value

Returns a data frame.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Dynamic Modeling Utility Functions: [DeleteInitialNA\(\)](#), [InitialNA\(\)](#), [InsertNA\(\)](#), [SubsetByID\(\)](#)

Examples

```
# prepare parameters
set.seed(42)
## number of individuals
n <- 5
## time points
time <- 5
## dynamic structure
p <- 3
mu0 <- rep(x = 0, times = p)
sigma0 <- 0.001 * diag(p)
sigma0_l <- t(chol(sigma0))
alpha <- rep(x = 0, times = p)
beta <- 0.50 * diag(p)
psi <- 0.001 * diag(p)
psi_l <- t(chol(psi))

library(simStateSpace)
ssm <- SimSSMVARFixed(
  n = n,
  time = time,
  mu0 = mu0,
  sigma0_l = sigma0_l,
  alpha = alpha,
  beta = beta,
  psi_l = psi_l,
  type = 0
)
data <- as.data.frame(ssm)
ScaleByID(
```

```

data = data,
id = "id",
time = "time",
observed = paste0("y", 1:p)
)

```

SubsetByID

*Subset Data Set by ID***Description**

The function creates a list of data frames for each ID.

Usage

```
SubsetByID(data, id, time, observed, covariates = 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.
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.
observed	Character vector. A vector of character strings of the names of the observed variables in the data.
covariates	Character vector. A vector of character strings of the names of the covariates in the data.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of individuals is large.

Value

Returns a list by ID numbers.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Dynamic Modeling Utility Functions: [DeleteInitialNA\(\)](#), [InitialNA\(\)](#), [InsertNA\(\)](#), [ScaleByID\(\)](#)

Examples

```
# prepare parameters
set.seed(42)
## number of individuals
n <- 5
## time points
time <- 5
## dynamic structure
p <- 3
mu0 <- rep(x = 0, times = p)
sigma0 <- 0.001 * diag(p)
sigma0_l <- t(chol(sigma0))
alpha <- rep(x = 0, times = p)
beta <- 0.50 * diag(p)
psi <- 0.001 * diag(p)
psi_l <- t(chol(psi))

library(simStateSpace)
ssm <- SimSSMVARFixed(
  n = n,
  time = time,
  mu0 = mu0,
  sigma0_l = sigma0_l,
  alpha = alpha,
  beta = beta,
  psi_l = psi_l,
  type = 0
)
data <- as.data.frame(ssm)
SubsetByID(
  data = data,
  id = "id",
  time = "time",
  observed = paste0("y", 1:p)
)
```

Index

* **Dynamic Modeling Utility Functions**

- DeleteInitialNA, [2](#)
- InitialNA, [3](#)
- InsertNA, [5](#)
- ScaleByID, [8](#)
- SubsetByID, [10](#)

* **data**

- DeleteInitialNA, [2](#)
- InitialNA, [3](#)
- InsertNA, [5](#)
- ScaleByID, [8](#)
- SubsetByID, [10](#)

* **dynUtils**

- DeleteInitialNA, [2](#)
- InitialNA, [3](#)
- InsertNA, [5](#)
- ScaleByID, [8](#)
- SubsetByID, [10](#)

* **methods**

- print.dynutillist, [7](#)

DeleteInitialNA, [2](#), [4](#), [6](#), [9](#), [10](#)

InitialNA, [2](#), [3](#), [6](#), [9](#), [10](#)

InsertNA, [2](#), [4](#), [5](#), [9](#), [10](#)

print.dynutillist, [7](#)

ScaleByID, [2](#), [4](#), [6](#), [8](#), [10](#)

SubsetByID, [2](#), [4](#), [6](#), [9](#), [10](#)