Package 'EMLI'

May 17, 2022

Type Package

Title Efficient Maximum Likelihood Identification

Version 0.1.0

Description Provides implementations of computationally efficient maximum likelihood estimation algorithms for system identification tasks. Currently, one such algorithm is implemented which identifies the one-dimensional cumulative structural equation model with normality assumptions. The corresponding scientific paper is yet to be published, therefore the relevant reference is not available yet.

License GPL-2

Copyright Vilnius University Institute of Data Science and Digital Technologies

Imports stats

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Author Vytautas Dulskis [cre, aut], Leonidas Sakalauskas [aut]

Maintainer Vytautas Dulskis <vytautas.dulskis@gmail.com>

Repository CRAN

Date/Publication 2022-05-17 16:30:02 UTC

R topics documented:

evaluate_identification *evaluate_identification*

Description

Calculates a discrepancy function based metric of estimation accuracy for the one-dimensional cumulative structural equation model with normality assumptions. Suitable when there are no contradictions in the factuals/estimates.

Usage

```
evaluate_identification(f, e, n)
```

Arguments

f	A list consisting of 3 elements: 1) the factual Sigma ((m + 1) x (m + 1) matrix of finite numeric elements); 2) the factual sigma_y^2 (vector of length 1, finite numeric element); 3) the factual mu ((m + 1) x 1 matrix of finite numeric elements).
e	Analogous to parameter f but with estimates instead of factuals.
n	The number of time moments used for obtaining parameter e (vector of length 1, finite positive integer).

Value

Calculated metric value (vector of length 1, numeric element). The lower the value, the better the accuracy, with 0 indicating perfect accuracy.

Examples

```
set.seed(1)

m <- 4
k <- 2

L <- matrix(runif((m + 1) * k, min = -10, max = 10), nrow = m + 1)
sigma <- matrix(runif(m + 2, min = 0, max = 10), nrow = m + 2)
mu <- matrix(runif(m + 1, min = -10, max = 10), nrow = m + 1)

n <- 100
data <- generate_data(n, L, sigma, mu)

Sigma <- L %*% t(L) + diag(sigma[1:(m + 1),] ^ 2)
sigma_y_squared <- sigma[m + 2,] ^ 2
Sigma[m + 1, m + 1] <- Sigma[m + 1, m + 1] + 2 * sigma_y_squared

factual_parameters <- list(Sigma, sigma_y_squared, mu)
```

```
estimated_parameters <- identify_model(data[[1]], data[[2]], 0.00001)

evaluate_identification(factual_parameters, estimated_parameters, n)
```

---

generate_data *generate_data*

Description

Generates data according to the one-dimensional cumulative structural equation model with normality assumptions with given model parameter values.

Usage

```
generate_data(n, L, sigma, mu)
```

Arguments

n	The number of time moments to generate the data for (vector of length 1, finite positive integer).
L	Factor loadings ((m + 1) x k matrix of finite numeric elements: the first m rows correspond to the input measurement equation; the last row corresponds to the transition equation).
sigma	Standard deviations of the error/noise terms ((m + 2) x 1 matrix of finite non-negative numeric elements: the first m rows correspond to the input measurement equation; the row before the last one corresponds to the transition equation; the last row corresponds to the output measurement equation).
mu	Intercept terms ((m + 1) x 1 matrix of finite numeric elements; the first m rows correspond to the input measurement equation; the last row corresponds to the transition equation).

Value

A list consisting of 2 elements: 1) observed input data (n x m matrix of numeric elements); 2) observed output differences data (n x 1 matrix of numeric elements).

Examples

```
set.seed(1)

m <- 4
k <- 2

L <- matrix(runif((m + 1) * k, min = -10, max = 10), nrow = m + 1)
sigma <- matrix(runif(m + 2, min = 0, max = 10), nrow = m + 2)
mu <- matrix(runif(m + 1, min = -10, max = 10), nrow = m + 1)
generate_data(10, L, sigma, mu)
```

identify_model *identify_model*

Description

Calculates maximum likelihood estimates of the statistical parameters of the one-dimensional cumulative structural equation model with normality assumptions.

Usage

```
identify_model(x, dy, tol)
```

Arguments

x	Observed input data (n x m matrix of finite numeric elements).
dy	Observed output differences data (n x 1 matrix of finite numeric elements).
tol	A tolerance parameter of the golden section search algorithm used for minimizing the one-dimensional likelihood function (vector of length 1, finite positive numeric element).

Value

A list consisting of 3 elements: 1) estimate of the covariance of cbind(x, dy) at lag 0 (Sigma; (m + 1) x (m + 1) matrix of numeric elements); 2) estimate of the only non-zero element of the negative covariance of cbind(x, dy) at lag 1 (sigma_y^2; vector of length 1, numeric element); 3) estimate of the mean of cbind(x, dy) (mu; (m + 1) x 1 matrix of numeric elements).

Examples

```
set.seed(1)

m <- 4
k <- 2

L <- matrix(runif((m + 1) * k, min = -10, max = 10), nrow = m + 1)
sigma <- matrix(runif(m + 2, min = 0, max = 10), nrow = m + 2)
mu <- matrix(runif(m + 1, min = -10, max = 10), nrow = m + 1)

data <- generate_data(100, L, sigma, mu)

identify_model(data[[1]], data[[2]], 0.00001)
```

Index