Package: EMLI (via r-universe)

September 17, 2024

Type Package

Title Computationally Efficient Maximum Likelihood Identification of Linear Dynamical Systems

Version 0.2.0

Description Provides implementations of computationally efficient maximum likelihood parameter estimation algorithms for models that represent linear dynamical systems. Currently, one such algorithm is implemented for the one-dimensional cumulative structural equation model with shock-error output measurement equation and assumptions of normality and independence. The corresponding scientific paper is yet to be published, therefore the relevant reference will be provided later.

License GPL-2

Copyright Vilnius University Institute of Data Science and Digital Technologies

Imports stats

Encoding UTF-8

RoxygenNote 7.2.2

NeedsCompilation no

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

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

Date/Publication 2022-11-20 19:40:02 UTC

Repository https://vy-du.r-universe.dev

RemoteUrl https://github.com/cran/EMLI

RemoteRef HEAD

RemoteSha 14f70fed5266505012ede1dc5000575194d0b63a

Contents

`calculate_likelihood` *calculate_likelihood*

Description

Calculates the likelihood function value for given data and statistical measure values of the output-differenced version of the one-dimensional cumulative structural equation model with shock-error output measurement equation and assumptions of normality and independence. Suitable when there are no contradictions in statistical measure values.

Usage

```
calculate_likelihood(dat, params)
```

Arguments

`dat`	An (n + 1) x (m + 1) data frame of finite numeric elements (possibly except for row 1 columns 1 to m) containing observed input (columns 1 to m) and output (column m + 1) data of the original model.
`params`	A list consisting of 3 elements: 1) Sigma ((m + 1) x (m + 1) matrix of finite numeric elements); 2) sigma_y^2 (vector of length 1, finite numeric element); 3) mu ((m + 1) x 1 matrix of finite numeric elements).

Value

Calculated likelihood function value (vector of length 1, numeric element).

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)
estimated_parameters <- estimate_parameters(data, 0.00001)

calculate_likelihood(data, estimated_parameters)
```

estimate_parameters *estimate_parameters*

Description

Calculates maximum likelihood estimates of the statistical measures of the output-differenced version of the one-dimensional cumulative structural equation model with shock-error output measurement equation and assumptions of normality and independence.

Usage

```
estimate_parameters(dat, tol)
```

Arguments

dat	An (n + 1) x (m + 1) data frame of finite numeric elements (possibly except for row 1 columns 1 to m) containing observed input (columns 1 to m) and output (column m + 1) data of the original model.
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 at lag 0 of the data that result from the output-differenced model (Sigma; (m + 1) x (m + 1) matrix of numeric elements); 2) estimate of the only non-zero element of the negative covariance at lag 1 of the data that result from the output-differenced model (sigma_y^2; vector of length 1, numeric element); 3) estimate of the mean of the data that result from the output-differenced model (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)

estimate_parameters(data, 0.00001)
```

evaluate_estimates *evaluate_estimates*

Description

Calculates a discrepancy-function-based metric of accuracy of the statistical measure estimates for the output-differenced version of the one-dimensional cumulative structural equation model with shock-error output measurement equation and assumptions of normality and independence. Suitable when there are no contradictions in the factuals/estimates.

Usage

```
evaluate_estimates(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 accuracy 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 <- estimate_parameters(data, 0.00001)

evaluate_estimates(factual_parameters, estimated_parameters, n)
```

---

generate_data *generate_data*

Description

Generates data according to the one-dimensional cumulative structural equation model with shock-error output measurement equation and assumptions of normality and independence 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

An (n + 1) x (m + 1) data frame of numeric elements (except for row 1 columns 1 to m that contain NA's) containing observed input (columns 1 to m) and output (column m + 1) data.

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)
```

Index