Package ‘GCCfactor’

October 30, 2023

Type Package

Title GCC Estimation of the Multilevel Factor Model

Version 1.0.1

Maintainer Rui Lin <ruilin1081@gmail.com>

Description
Provides methods for model selection, estimation, bootstrap inference, and simulation for the multilevel factor model, based on the principal component estimation and generalised canonical correlation approach. Details can be found in ``Generalised Canonical Correlation Estimation of the Multilevel Factor Model.'' Lin and Shin (2023) <doi:10.2139/ssrn.4295429>.

Imports stats, stringr, sandwich

Suggests parallel, plm

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Depends R (>= 2.10)

NeedsCompilation no

Author Rui Lin [aut, cre],
Yongcheol Shin [aut]

Repository CRAN

Date/Publication 2023-10-30 05:10:02 UTC

R topics documented:

`AsymCI_local_loading` *Get an asymptotic confidence interval for the local component*

Description

This function computes the asymptotic confidence intervals for the local loadings for the j-th individual in block i. See Lin and Shin (2023) for details.

Usage

```
AsymCI_local_loading(object, i, j, alpha = 0.05)
```

Arguments

`object`	An S3 object of class 'multi_result' created by multilevel().
`i`	An integer indicating the i-th block.
`j`	An integer indicating the j-th individual in the i-th block.
`alpha`	The significance level, a single numeric between 0 and 1. 0.05 by default.

Value

A matrix containing the upper and lower band.

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                        depvar_header = "dlPrice", i_header = "Region",
                        j_header = "LPA_Type", t_header = "Date")
bs_local_loading_11 <- AsymCI_local_loading(est_multi, i = 1, j = 1)
```

`Bartlett` *Bartlett kernel function*

Description

Evaluate the Bartlett kernel function: $Bartlett(x) = 1 - |x|$ if $|x| \leq 1$ and $Bartlett(x) = 1 - |x|$ otherwise.

Usage

```
Bartlett(x)
```

Arguments

`x`	A single numeric.

Value

A single numeric between 0 and 1.

Examples

```
Bartlett(0.5)
```

`BS_global_comp` *Get a bootstrap confidence interval for the global component*

Description

This function employs a bootstrap procedure to obtain confidence intervals for the global component for the j-th individual in block i at time t. See Lin and Shin (2023) for details.

Usage

```
BS_global_comp(object, i, j, t, BB = 599, alpha = 0.05)
```

Arguments

`object`	An S3 object of class 'multi_result' created by multilevel().
`i`	An integer indicating the i-th block.
`j`	An integer indicating the j-th individual in the i-th block.
`t`	An integer specifying the time point at which the CI is constructed.
`BB`	An integer indicating the number of bootstrap repetition. 599 by default.
`alpha`	The significance level, a single numeric between 0 and 1. 0.05 by default.

Value

A matrix containing the upper and lower band.

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                         depvar_header = "dlPrice", i_header = "Region",
                         j_header = "LPA_Type", t_header = "Date")
bs_gcomp_111 <- BS_global_comp(est_multi, i = 1, j = 1, t = 1)
```

BS_global_factor *Get bootstrap confidence intervals for the global factors*

Description

This function employs a bootstrap procedure to obtain confidence intervals for the global factors at time t.

Usage

```
BS_global_factor(object, t, BB = 599, alpha = 0.05)
```

Arguments

`object`	An S3 object of class 'multi_result' created by multilevel().
`t`	An integer specifying the time point at which the CI is constructed.
`BB`	An integer indicating the number of bootstrap repetition. 599 by default.
`alpha`	The significance level, a single numeric between 0 and 1. 0.05 by default.

Value

A matrix containing the upper and lower band.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                        depvar_header = "dlPrice", i_header = "Region",
                        j_header = "LPA_Type", t_header = "Date")
bs_global_mid <- BS_global_factor(est_multi, t = est_multi$T / 2)
```

BS_global_loading — *Get a bootstrap confidence interval for the global factor loadings*

Description

This function employs a bootstrap procedure to obtain confidence intervals for the global factor loadings for the j-th individual in block i. See Lin and Shin (2023) for details.

Usage

```
BS_global_loading(object, i, j, BB = 599, alpha = 0.05)
```

Arguments

object	An S3 object of class 'multi_result' created by [multilevel()].
i	An integer indicating the i-th block.
j	An integer indicating the j-th individual in the i-th block.
BB	An integer indicating the number of bootstrap repetition. 599 by default.
alpha	The significance level, a single numeric between 0 and 1. 0.05 by default.

Value

A matrix containing the upper and lower band.

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                        depvar_header = "dlPrice", i_header = "Region",
                        j_header = "LPA_Type", t_header = "Date")
bs_gamma_11 <- BS_global_loading(est_multi, i = 1, j = 1)
```

BS_local_comp *Get a bootstrap confidence interval for the global component*

Description

This function employs a bootstrap procedure to obtain confidence intervals for the local component for the j-th individual in block i at time t. See Lin and Shin (2023) for details.

Usage

```
BS_local_comp(object, i, j, t, BB = 599, alpha = 0.05)
```

Arguments

`object`	An S3 object of class 'multi_result' created by multilevel().
`i`	An integer indicating the i-th block.
`j`	An integer indicating the j-th individual in the i-th block.
`t`	An integer specifying the time point at which the CI is constructed.
`BB`	An integer indicating the number of bootstrap repetition. 599 by default.
`alpha`	The significance level, a single numeric between 0 and 1. 0.05 by default.

Value

A matrix containing the upper and lower band.

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                        depvar_header = "dlPrice", i_header = "Region",
                        j_header = "LPA_Type", t_header = "Date")
bs_fcomp_111 <- BS_local_comp(est_multi, i = 1, j = 1, t = 1)
```

`BS_local_factor` *Get a bootstrap confidence interval for the local factors*

Description

This function employs a bootstrap procedure to obtain confidence intervals for the local factors in block i at time t. See Lin and Shin (2023) for details.

Usage

```
BS_local_factor(object, i, t, BB = 599, alpha = 0.05)
```

Arguments

`object`	An S3 object of class 'multi_result' created by multilevel().
`i`	An integer indicating the i-th block.
`t`	An integer specifying the time point at which the CI is constructed.
`BB`	An integer indicating the number of bootstrap repetition. 599 by default.
`alpha`	The significance level, a single numeric between 0 and 1. 0.05 by default.

Value

A matrix containing the upper and lower band.

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                        depvar_header = "dlPrice", i_header = "Region",
                        j_header = "LPA_Type", t_header = "Date")
bs_local_factor_11 <- BS_local_factor(est_multi, i = 1, t = 1)
```

check_data *Check validity of the data and headers*

Description

This is an internal function which checks the validity of the data and provide a list of matrices of length R for estimation.

Usage

```
check_data(
  data,
  depvar_header = NULL,
  i_header = NULL,
  j_header = NULL,
  t_header = NULL
)
```

Arguments

data	Either a data.frame or a list of data matrices of length R. See **Details**.
depvar_header	A character string specifying the header of the dependent variable. See **Details**.
i_header	A character string specifying the header of the block identifier. See **Details**.
j_header	A character string specifying the header of the individual identifier. See **Details**.
t_header	A character string specifying the header of the time identifier. See **Details**.

Details

See **Details** of GCC().

Value

A list of data matrices of length R.

Examples

```
panel <- UKhouse # load the data
Y_list <- check_data(panel,
  depvar_header = "dlPrice", i_header = "Region",
  j_header = "LPA_Type", t_header = "Date"
)
```

`dwBS` *Dependent wild bootstrap for resampling time series*

Description

Select an optimal bandwidth parameter and apply the dependent wild bootstrap with Bartlett kernel to obtain the resampled time series.

Usage

```
dwBS(y)
```

Arguments

y	A $T \times 1$ vector of time series to be resampled.

Value

A $T \times 1$ matrix of resampled time series.

References

Shao, X., 2010. The dependent wild bootstrap. Journal of the American Statistical Association, 105(489), pp.218-235.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                            depvar_header = "dlPrice", i_header = "Region",
                            j_header = "LPA_Type", t_header = "Date")
G_star <- dwBS(est_multi$G)
```

`GCC` *Generalised canonical correlation estimation for the global factors*

Description

This function is one of the main functions the package, employing the generalized canonical correlation estimation for both the global factors $\boldsymbol{G}$ and, when not explicitly provided, for the number of global factors r_0. Typically, this function is intended for internal purposes. However, users one can opt for GCC() instead of multilevel(), if the users only need to estimate the number of global factors.

Usage

```
GCC(
  data,
  standarise = TRUE,
  r_max = 10,
  r0 = NULL,
  ri = NULL,
  depvar_header = NULL,
  i_header = NULL,
  j_header = NULL,
  t_header = NULL
)
```

Arguments

`data`	Either a data.frame or a list of data matrices of length R. See **Details**.
`standarise`	A logical indicating whether the data is standardised before estimation or not. See **Details**.
`r_max`	An integer indicating the maximum number of factors allowed. See **Details**.
`r0`	An integer of the number of global factors. See **Details**.
`ri`	An array of length R containing the number of local factors in each block. See **Details**.
`depvar_header`	A character string specifying the header of the dependent variable. See **Details**.
`i_header`	A character string specifying the header of the block identifier. See **Details**.
`j_header`	A character string specifying the header of the individual identifier. See **Details**.
`t_header`	A character string specifying the header of the time identifier. See **Details**.

Details

The user-supplied data.frame should contain at least four columns, namely the dependent variable (y_{ijt}), block identifier (i), individual identifier (j), and time (t). The user needs to supply their corresponding headers in the data.frame to the function using the parameters "depvar_header", "i_header", "j_header", and "t_header", respectively. If the data is supplied as a list, these arguments will not be used.

If either r0 = NULL or ri = NULL, both of them will be estimated. In such case, "r_max" must be supplied. If "r0" and "ri" are supplied then "r_max" is not needed and will be ignored.

If standarise = TRUE, each time series will be standardised so it has zero mean and unit variance. It is recommended to standardise the data before estimation.

See Lin and Shin (2023) for more details.

Value

A list containing the estimated number of global factors $\hat{r}_0$, the global factors $\widehat{\boldsymbol{G}}$, and the other elements that are used in multilevel().

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
panel <- UKhouse # load the data
Y_list <- panel2list(panel, depvar_header = "dlPrice", i_header = "Region",
                                       j_header = "LPA_Type", t_header = "Date")
est_GCC <- GCC(Y_list, r_max = 10)
r0_hat <- est_GCC$r0 # number of global factors
G_hat <- est_GCC$G # global factors
```

---

get_bw *Get an optimal bandwidth using Bartlett kernel*

Description

Automatic bandwidth selection of Andrews (1991) using Bartlett kernel.

Usage

```
get_bw(y)
```

Arguments

y	A $T \times 1$ vector of time series

Value

A numeric.

References

Andrews, D.W., 1991. Heteroskedasticity and autocorrelation consistent covariance matrix estimation. Econometrica: Journal of the Econometric Society, pp.817-858.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                           depvar_header = "dlPrice", i_header = "Region",
                           j_header = "LPA_Type", t_header = "Date")
lT_G <- get_bw(est_multi$G)
```

`infocrit` *Selection criteria for the approximate factor model*

Description

This function performs model selection for the (2D) approximate factor model and returns the estimated number of factors.

Usage

```
infocrit(Y, method, r_max = 10)
```

Arguments

`Y`	A $T \times N$ data matrix. T = number of time series observations, N = cross-sectional dimension.
`method`	A character string indicating which criteria to use.
`r_max`	An integer indicating the maximum number of factors allowed. 10 by default.

Details

"method" can be one of the following: "ICp2" and "BIC3" by Bai and Ng (2002), "ER" by Ahn and Horenstein (2013), "ED" by Onatski (2010).

Value

The estimated number of factors.

References

Bai, J. and Ng, S., 2002. Determining the number of factors in approximate factor models. Econometrica, 70(1), pp.191-221.

Ahn, S.C. and Horenstein, A.R., 2013. Eigenvalue ratio test for the number of factors. Econometrica, 81(3), pp.1203-1227.

Onatski, A., 2010. Determining the number of factors from empirical distribution of eigenvalues. The Review of Economics and Statistics, 92(4), pp.1004-1016.

Examples

```
# simulate data

T <- 100
N <- 50
r <- 2
F <- matrix(stats::rnorm(T * r, 0, 1), nrow = T)
Lambda <- matrix(stats::rnorm(N * r, 0, 1), nrow = N)
err <- matrix(stats::rnorm(T * N, 0, 1), nrow = T)
```

```
Y <- F %*% t(Lambda) + err

# estimation

r_hat <- infocrit(Y, "BIC3", r_max = 10)
```

`multilevel` *Full estimation of the multilevel factor model*

Description

This is one of the main functions of this package which performs full estimation of the multilevel factor model.

Usage

```
multilevel(
  data,
  ic = "BIC3",
  standarise = TRUE,
  r_max = 10,
  r0 = NULL,
  ri = NULL,
  depvar_header = NULL,
  i_header = NULL,
  j_header = NULL,
  t_header = NULL
)
```

Arguments

data	Either a data.frame or a list of data matrices of length R. See **Details**.
ic	A character string of selection criteria to use for estimation of the numbers of local factors. See **Details**.
standarise	A logical indicating whether the data is standardised before estimation or not. See **Details**.
r_max	An integer indicating the maximum number of factors allowed. See **Details**.
r0	An integer of the number of global factors. See **Details**.
ri	An array of length R containing the number of local factors in each block. See **Details**.
depvar_header	A character string specifying the header of the dependent variable. See **Details**.
i_header	A character string specifying the header of the block identifier. See **Details**.
j_header	A character string specifying the header of the individual identifier. See **Details**.
t_header	A character string specifying the header of the time identifier. See **Details**.

Details

The user-supplied data.frame should contain at least four columns, namely the dependent variable (y_{ijt}), block identifier (i), individual identifier (j), and time (t). The user needs to supply their corresponding headers in the data.frame to the function using the parameters "depvar_header", "i_header", "j_header", and "t_header", respectively. If the data is supplied as a list, these arguments will not be used.

If either r0 = NULL or ri = NULL, then both of them will be estimated. In such case, "r_max" must be supplied. If "r0" and "ri" are supplied then "r_max" is not needed and will be ignored.

If standarise = TRUE, each time series will be standardised so it has zero mean and unit variance. It is recommended to standardise the data before estimation.

See Lin and Shin (2023) for more details.

Value

The return value is an S3 object of class "multi_result". It contains a list of the following items:

- G = A matrix of the estimated global factors.
- Gamma = A list of length R containing matrices of the estimated global loading matrices for each block.
- F = A list of length R containing matrices of the estimated local factors for each block.
- Lambda = A list of length R containing matrices of the estimated global loading matrices for each block.
- N = The total number of cross-sections in the panel.
- Ni = An array of length R containing the number of cross-sections in each block.
- r0 = The number of global factors. Unchanged if pre-specified.
- ri = An array of length R containing the number of local factors for each block. Unchanged if pre-specified.
- d = An array of length R containing the maximum total number of factors allowed for each block. The elements are identically equal to r_max if either r0 or ri is supplied as NULL.
- Resid = A list of length R containing the residual matrices for each block.
- delta2 = An array of the mock and the $r_{\max} + 1$ largest squared singular values.
- ic = Selection criteria used for estimating the numbers of local factors.
- block_names = A array of block names.

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
panel <- UKhouse # load the data

# use data.frame
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                           depvar_header = "dlPrice", i_header = "Region",
                           j_header = "LPA_Type", t_header = "Date")
# or one can use a list of data matrices
Y_list <- panel2list(panel, depvar_header = "dlPrice", i_header = "Region",
                                       j_header = "LPA_Type", t_header = "Date")
est_multi <- multilevel(Y_list, ic = "BIC3", standarise = TRUE, r_max = 5)
```

---

panel2list *data.frame to list of data matrices*

Description

This function converts the data.frame to a list of data matrices and finds the dimensions of the multilevel panel.

Usage

```
panel2list(
  panel,
  depvar_header = NULL,
  i_header = NULL,
  j_header = NULL,
  t_header = NULL
)
```

Arguments

panel	The user-supplied data frame for the multilevel panel data. See **Details**.
depvar_header	A character string specifying the header of the dependent variable. See **Details**.
i_header	A character string specifying the header of the block identifier. See **Details**.
j_header	A character string specifying the header of the individual identifier. See **Details**.
t_header	A character string specifying the header of the time identifier. See **Details**.

Details

See the details of GCC().

Value

A list containing the data matrices of the R blocks. Each of them has dimension $T \times N_i$.

Examples

```
panel <- UKhouse # load the data

# panel$Region identifies different blocks i=1,...,R.
# panel$LPA_Type identifies different individuals j=1,...,N_i.

Y_list<- panel2list(panel, depvar_header = "dlPrice", i_header = "Region",
                                       j_header = "LPA_Type", t_header = "Date")
```

PC *Principal component (PC) estimation of the approximate factor model*

Description

Perform PC estimation of the (2D) approximate factor model:

$$y_{it} = \boldsymbol{\lambda}_i' \boldsymbol{F}_t + e_{it},$$

or in matrix notation:

$$\boldsymbol{Y} = \boldsymbol{F}\boldsymbol{\Lambda}' + \boldsymbol{e}.$$

The factors $\boldsymbol{F}$ is estimated as $\sqrt{T}$ times the r eigenvectors of the matrix $\boldsymbol{Y}\boldsymbol{Y}'$ corresponding to the r largest eigenvalues in descending order, and the loading matrix is estimated by $\boldsymbol{\Lambda} = T^{-1}\boldsymbol{Y}'\boldsymbol{F}$. See e.g. Bai and Ng (2002).

Usage

```
PC(Y, r)
```

Arguments

Y	A $T \times N$ data matrix. T = number of time series observations, N = cross-sectional dimension.
r	= the number of factors.

Value

A list containing the factors and factor loadings:

- factor = a $T \times r$ matrix of the estimated factors.
- loading = a $N \times r$ matrix of the estimated factor loadings.

References

Bai, J. and Ng, S., 2002. Determining the number of factors in approximate factor models. Econometrica, 70(1), pp.191-221.

Examples

```
# simulate data

T <- 100
N <- 50
r <- 2
F <- matrix(stats::rnorm(T * r, 0, 1), nrow = T)
Lambda <- matrix(stats::rnorm(N * r, 0, 1), nrow = N)
err <- matrix(stats::rnorm(T * N, 0, 1), nrow = T)
Y <- F %*% t(Lambda) + err

# estimation

est_PC <- PC(Y, r)
```

`summary.multi_result` *Print the relative importance ratios*

Description

Print the relative importance ratios

Usage

```
## S3 method for class 'multi_result'
summary(object, ...)
```

Arguments

`object`	An S3 object of class 'multi_result' created by multilevel().
`...`	Additional arguments.

Value

A matrix containing the summary of the model.

Examples

```
panel <- UKhouse # load the data
est_multi <- multilevel(panel, ic = "BIC3", standarise = TRUE, r_max = 5,
                           depvar_header = "dlPrice", i_header = "Region",
                           j_header = "LPA_Type", t_header = "Date")
summary(est_multi)
```

UKhouse *England and Wales House Price Growth Data Categorised by Regions*

Description

A data.frame containing the quarterly (mean) house prices of four different types of properties, (detached, semi-detached, terraced and flats/maisonettes) for 331 local planning authorities (LPA) over the period 1996Q1 to 2021Q2. See also Lin and Shin (2023).

Usage

```
UKhouse
```

Format

'UKhouse'

Details

Each LPA belongs to one of the ten regions: North East (NE), North West (NW), Yorkshire and the Humber (YH), East Midlands (EM), West Midlands(WM), East of England (EE), London (LD), South East (SE), South West (SW) and Wales (WA). The real house price growth of the j-th LPA-type pair in region i by deflating the nominal house price by CPI and log-differencing it as

$$\pi_{ijt} = 100 \times \log\left(\frac{PRICE_{ijt}}{CPI_t}\right) - 100 \times \log\left(\frac{PRICE_{ij,t-1}}{CPI_{t-1}}\right).$$

By removing the series with missing observations, it ends up with a balanced panel with $R = 10$, $N = \sum_{i=1}^{R} N_i = 1300$ and $T = 102$.

Columns in the dataset:

- "Date" Time variable.
- "Region" Name of region which the LPA belongs to.
- "LPA" Name of the LPA.
- "Type" Name of the house type.
- "LPA_Type" Name of the LPA-type pair.

Source

Office for National Statistics (ONS), ONS website, statistical bulletin, House price statistics for small areas in England and Wales: year ending June 2021

References

Lin, R. and Shin, Y., 2022. Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Index