Package: GrFA (via r-universe)

August 25, 2024

Type Package

Title Group Factor Analysis

Date 2024-07-25

Version 0.2

Author Jiaqi Hu [cre, aut], Ting Li [aut], Xueqin Wang [aut]

Maintainer Jiaqi Hu <hujiaqi@mail.ustc.edu.cn>

Description Several group factor analysis algorithms are implemented, including Canonical Correlation-based Estimation by Choi et al. (2021) <doi:10.1016/j.jeconom.2021.09.008> , Generalised Canonical Correlation Estimation by Lin and Shin (2023) <doi:10.2139/ssrn.4295429>, Circularly Projected Estimation by Chen (2022) <doi:10.1080/07350015.2022.2051520>, and Aggregated projection method.

Imports mvtnorm

Depends R (>= 3.5.0)

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.3

LazyData true

NeedsCompilation no

Date/Publication 2024-08-24 10:00:02 UTC

Repository https://jiaqihu2021.r-universe.dev

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

RemoteRef HEAD

RemoteSha c9928fc10b22be0eb43432f252acd196df984115

Contents

`APM` *Aggregated Projection Method*

Description

Aggregated Projection Method

Usage

```
APM(y, rmax = 8, r0 = NULL, r = NULL, localfactor = FALSE, weight = TRUE,
      method = "ic", type = "IC3")
```

Arguments

`y`	a list of the observation data, each element is a data matrix of each group with dimension $T * N_m$.
`rmax`	the maximum factor numbers of all groups.
`r0`	the number of global factors, default is `NULL`, the algorithm will automatically estimate the number of global factors. If you have the prior information about the true number of global factors, you can set it by your own.
`r`	the number of local factors in each group, default is `NULL`, the algorithm will automatically estimate the number of local factors. If you have the prior information about the true number of local factors, you can set it by your own, notice it should be an integer vector of length M (the number of groups).
`localfactor`	if `localfactor` = `FALSE`, then we would not estimate the local factors; if `localfactor` = `TRUE`, then we will further estimate the local factors.
`weight`	the weight of each projection matrix, default is `TRUE`, means $w_m = N_m/N$, if `weight` = `FALSE`, then simply calculate the mean of all projection matrices.
`method`	the method used in the algorithm, default is `ic`, it can also be `gap`.
`type`	the method used in estimating the factor numbers in each group initially, default is `IC3`

Value

- `r0hat` — the estimated number of the global factors.
- `rho` — the estimated number of the local factors.
- `Ghat` — the estimated global factors.
- `loading_G` — a list consisting of the estimated global factor loadings.
- `Fhat` — the estimated local factors.
- `loading_F` — a list consisting of the estimated local factor loadings.
- `e` — a list consisting of the residuals.
- `threshold` — the threshold used in determining the number of global factors, only for `method = ic`.

Examples

```
dat = gendata()
dat
APM(dat$y, rmax = 8, localfactor = TRUE, method = "ic")
APM(dat$y, rmax = 8, localfactor = TRUE, method = "gap")
```

CCA — *Canonical Correlation Estimation*

Description

Canonical Correlation Estimation

Usage

```
CCA(y, rmax = 8, r0 = NULL, r = NULL, localfactor = FALSE, method = "CCD", type = "IC3")
```

Arguments

- `y` — a list of the observation data, each element is a data matrix of each group with dimension $T * N_m$.
- `rmax` — the maximum factor numbers of all groups.
- `r0` — the number of global factors, default is `NULL`, the algorithm will automatically estimate the number of global factors. If you have the prior information about the true number of global factors, you can set it by your own.
- `r` — the number of local factors in each group, default is `NULL`, the algorithm will automatically estimate the number of local factors. If you have the prior information about the true number of local factors, you can set it by your own, notice it should be an integer vector of length M (the number of groups).
- `localfactor` — if `localfactor = FALSE`, then we would not estimate the local factors; if `localfactor = TRUE`, then we will further estimate the local factors.
- `method` — the method used in the algorithm, default is `CCD`, it can also be `MCC`.
- `type` — the method used in estimating the factor numbers in each group initially, default is `IC3`.

Value

r0hat	the estimated number of the global factors.
rho	the estimated number of the local factors.
Ghat	the estimated global factors.
Fhat	the estimated local factors.
loading_G	a list consisting of the estimated global factor loadings.
loading_F	a list consisting of the estimated local factor loadings.
e	a list consisting of the residuals.
threshold	the threshold used in determining the number of global factors, only for method = "MCC".

References

Choi, I., Lin, R., & Shin, Y. (2021). Canonical correlation-based model selection for the multilevel factors. Journal of Econometrics.

Examples

```
dat = gendata()
dat
CCA(dat$y, rmax = 8, localfactor = TRUE, method = "CCD")
CCA(dat$y, rmax = 8, localfactor = TRUE, method = "MCC")
```

CP *Circularly Projected Estimation*

Description

Circularly Projected Estimation

Usage

```
CP(y, rmax = 8, r0 = NULL, r = NULL, localfactor = FALSE, type = "IC3")
```

Arguments

y	a list of the observation data, each element is a data matrix of each group with dimension $T * N_m$.
rmax	the maximum factor numbers of all groups.
r0	the number of global factors, default is NULL, the algorithm will automatically estimate the number of global factors. If you have the prior information about the true number of global factors, you can set it by your own.

`r`	the number of local factors in each group, default is `NULL`, the algorithm will automatically estimate the number of local factors. If you have the prior information about the true number of local factors, you can set it by your own, notice it should be an integer vector of length M (the number of groups).
`localfactor`	if `localfactor = FALSE`, then we would not estimate the local factors; if `localfactor = TRUE`, then we will further estimate the local factors.
`type`	the method used in estimating the local factor numbers in each group after projecting out the global factors, default is `IC3`.

Value

`r0hat`	the estimated number of the global factors.
`rho`	the estimated number of the local factors.
`Ghat`	the estimated global factors.
`Fhat`	the estimated local factors.
`loading_G`	a list consisting of the estimated global factor loadings.
`loading_F`	a list consisting of the estimated local factor loadings.
`e`	a list consisting of the residuals.

References

Chen, M. (2023). Circularly Projected Common Factors for Grouped Data. Journal of Business & Economic Statistics, 41(2), 636-649.

Examples

```
dat = gendata()
dat
CP(dat$y, rmax = 8, localfactor = TRUE)
```

`est_num` *Estimate factor numbers*

Description

Estimate factor numbers.

Usage

```
est_num(X, kmax = 8, type = "BIC3")
```

Arguments

`X`	the observation data matrix of dimension $T \times N$.
`kmax`	the maximum number of factors.
`type`	the criterion used in determining the number of factors, default is `type = "BIC3"`, it can also be `"PC1"`, `"PC2"`, `"PC3"`, `"IC1"`, `"IC2"`,`"IC3"`, `"AIC3"`, `"BIC3"`, `"ER"`, `"GR"`.

Value

`rhat`	the estimated number of factors.

References

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

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

`FA` *Factor analysis*

Description

Factor analysis.

Usage

```
FA(X, r)
```

Arguments

`X`	the observation data matrix of dimension $T \times N$.
`r`	the factor numbers need to estimated.

Value

`F`	the estimated factors.
`L`	the estimated factor loadings.

Author(s)

Jiaqi Hu

References

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

GCC *Generalised Canonical Correlation*

Description

Generalised Canonical Correlation

Usage

```
GCC(y, rmax = 8, r0 = NULL, r = NULL, localfactor = FALSE, type = "IC3")
```

Arguments

y	a list of the observation data, each element is a data matrix of each group with dimension $T * N_m$.
rmax	the maximum factor numbers of all groups.
r0	the number of global factors, default is NULL, the algorithm will automatically estimate the number of global factors. If you have the prior information about the true number of global factors, you can set it by your own.
r	the number of local factors in each group, default is NULL, the algorithm will automatically estimate the number of local factors. If you have the prior information about the true number of local factors, you can set it by your own, notice it should be an integer vector of length M (the number of groups).
localfactor	if localfactor = FALSE, then we would not estimate the local factors; if localfactor = TRUE, then we will further estimate the local factors.
type	the method used in estimating the factor numbers in each group initially, default is IC3.

Value

r0hat	the estimated number of the global factors.
rho	the estimated number of the local factors.
Ghat	the estimated global factors.
Fhat	the estimated local factors.
loading_G	a list consisting of the estimated global factor loadings.
loading_F	a list consisting of the estimated local factor loadings.
e	a list consisting of the residuals.

References

Lin, R., & Shin, Y. (2023). Generalised Canonical Correlation Estimation of the Multilevel Factor Model. Available at SSRN 4295429.

Examples

```
dat = gendata()
dat
GCC(dat$y, rmax = 8, localfactor = TRUE)
```

`gendata` *Generate the grouped data.*

Description

Generate the grouped data.

Usage

```
gendata(seed = 1, T = 50, N = rep(30, 5), r0 = 2, r = rep(2, 5),
        Phi_G = 0.5, Phi_F = 0.5, Phi_e = 0.5, W_F = 0.5, beta = 0.2,
        kappa = 1, case = 1)
```

Arguments

`seed`	the seed used in `set.seed`.
`T`	the number of time points.
`N`	a vector representing the number of variables in each group.
`r0`	the number of global factors.
`r`	a vector representing the number of the local factors. Notice, the length of r is the same as N.
`Phi_G`	hyperparameter of the global factors, default is 0.5, the value should between 0 and 1.
`Phi_F`	hyperparameter of the local factors, default is 0.5, the value should between 0 and 1.
`Phi_e`	hyperparameter of the errors, default is 0.5, the value should between 0 and 1.
`W_F`	hyperparameter of the correlation of local factors, only applicable in `case = 3`, the value should between 0 and 1.
`beta`	hyperparameter of the errors, default is 0.2.
`kappa`	hyperparameter of signal to noise ratio, default is 1.
`case`	the case of the data-generating process, default is 1, it can also be 2 and 3.

Value

y	a list of the data.
G	the global factors.
F	a list of the local factors.
loading_G	the global factor loadings.
loading_F	the local factor loadings.
T	the number of time points.
N	a vector representing the number of variables in each group.
M	the number of groups.
r0	the number of global factors.
r	a vector representing the number of the local factors.
case	the case of the data-generating process.

Examples

```
dat = gendata()
dat
```

---

print.GFA *Print*

Description

Print the summarized results of the estimated group factor model, such as the estimated global and local factors.

Usage

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

Arguments

x	the GFA object returned from the algorithm.
...	additional print arguments.

Value

No return value, called for side effects

TraceRatio *Trace ratio*

Description

Evaluation of the estimated factors by trace ratios, the values is between 0 and 1, higher values means better estimation.

Usage

```
TraceRatio(G, Ghat)
```

Arguments

G	the true factors.
Ghat	the estimated factors.

Value

trace ratio	defined as $\mathrm{TR} = \mathrm{tr}(\mathbf{G}'\widehat{\mathbf{G}}(\widehat{\mathbf{G}}'\widehat{\mathbf{G}})^{-1}\widehat{\mathbf{G}}'\mathbf{G})/\mathrm{tr}(\mathbf{G}'\mathbf{G})$.

UShouseprice *Housing price data for 16 states in the U.S.*

Description

Housing price data for 16 states in the U.S over the period Jan 2000 to April 2023.

Usage

```
data("UShouseprice")
```

Format

A list with a length of 16. Each element is a matrix of dimension $T * N_m$.

Source

The original data is downloaded from the website of Zillow.

Examples

```
data(UShouseprice)
log_diff = function(x){
  T = nrow(x)
  res = log(x[2:T,]/x[1:(T-1),])*100
  scale(res, center = TRUE, scale = TRUE)
}
UShouseprice1 = lapply(UShouseprice, log_diff)
```

Index