

Package ‘HDRFA’

July 21, 2025

Type Package

Title High-Dimensional Robust Factor Analysis

Version 0.1.5

Author Yong He [aut],
Lingxiao Li [aut],
Dong Liu [aut, cre],
Wenxin Zhou [aut]

Maintainer Dong Liu <liudong_stat@163.com>

Description

Factor models have been widely applied in areas such as economics and finance, and the well-known heavy-tailedness of macroeconomic/financial data should be taken into account when conducting factor analysis. We propose two algorithms to do robust factor analysis by considering the Huber loss. One is based on minimizing the Huber loss of the idiosyncratic error's L2 norm, which turns out to do Principal Component Analysis (PCA) on the weighted sample covariance matrix and thereby named as Huber PCA. The other one is based on minimizing the element-wise Huber loss, which can be solved by an iterative Huber regression algorithm. In this package we also provide the code for traditional PCA, the Robust Two Step (RTS) method by He et al. (2022) and the Quantile Factor Analysis (QFA) method by Chen et al. (2021) and He et al. (2023).

License GPL-2 | GPL-3

Encoding UTF-8

Imports quantreg, pracma

Depends R (>= 3.5.0)

NeedsCompilation no

Repository CRAN

Date/Publication 2024-07-22 04:50:02 UTC

Contents

HPCA	2
HPCA_FN	3
IQR	5

IQR_FN	6
PCA	7
PCA_FN	8
RTS	9
RTS_FN	10

Index	12
--------------	-----------

HPCA	<i>Huber Principal Component Analysis for Large-Dimensional Factor Models</i>
------	---

Description

This function is to fit the factor models via the Huber Principal Component Analysis (HPCA) method. One is based on minimizing the Huber loss of the idiosyncratic error's ℓ_2 norm, which turns out to do Principal Component Analysis (PCA) on the weighted sample covariance matrix and thereby named as Huber PCA. The other one is based on minimizing the elementwise Huber loss, which can be solved by an iterative Huber regression algorithm.

Usage

```
HPCA(X, r, Method = "E", tau = NULL, scale_est="MAD", L_init = NULL,
      F_init = NULL, maxiter_HPCA = 100, maxiter_HLM = 100, eps = 0.001)
```

Arguments

X	Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .
r	A positive integer indicating the factor numbers.
Method	Method="P" indicates minimizing the Huber loss of the idiosyncratic error's ℓ_2 norm while Method="E" indicates minimizing the elementwise Huber loss. The default is the elementwise Huber loss.
tau	Optional user-supplied parameter for Huber loss; default is NULL, and τ is provided by default.
scale_est	A parameter for the elementwise Huber loss. scale_est="MAD" indicates robust variance estimation in each iteration, while scale_est="const" indicates fixing user-supplied τ . The default is scale_est="MAD".
L_init	User-supplied initial value of loadings; default is the PCA estimator.
F_init	User-supplied initial value of factors; default is the PCA estimator.
maxiter_HPCA	The maximum number of iterations in the HPCA. The default is 100.
maxiter_HLM	The maximum number of iterations in the iterative Huber regression algorithm. The default is 100.
eps	The stopping criterion parameter in the HPCA. The default is 1e-3.

Details

See He et al. (2023) for details.

Value

The return value is a list. In this list, it contains the following:

Fhat	The estimated factor matrix of dimension $T \times r$.
Lhat	The estimated loading matrix of dimension $N \times r$.
m	The number of iterations.

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

He Y, Li L, Liu D, Zhou W., 2023 Huber Principal Component Analysis for Large-dimensional Factor Models.

Examples

```
set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

fit=HPCA(X,r,Method = "E")
fit$Fhat;fit$Lhat

fit=HPCA(X,r,Method = "P")
fit$Fhat;fit$Lhat
```

HPCA_FN

Estimating Factor Numbers via Rank Minimization Corresponding to Huber PCA

Description

This function is to estimate factor numbers via rank minimization corresponding to Huber Principal Component Analysis (HPCA).

Usage

```
HPCA_FN(X, rmax, Method = "E", tau = NULL, scale_est="MAD", threshold = NULL,
        L_init = NULL, F_init = NULL, maxiter_HPCA = 100, maxiter_HLM = 100,
        eps = 0.001)
```

Arguments

X	Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .
rmax	The user-supplied maximum factor numbers.
Method	Method="P" indicates minimizing the Huber loss of the idiosyncratic error's ℓ_2 norm while Method="E" indicates minimizing the elementwise Huber loss. The default is the elementwise Huber loss.
tau	Optional user-supplied parameter for Huber loss; default is NULL, and τ is provided by default.
scale_est	A parameter for the elementwise Huber loss. scale_est="MAD" indicates robust variance estimation in each iteration, while scale_est="const" indicates fixing user-supplied τ . The default is scale_est="MAD".
threshold	The threshold of rank minimization; default is NULL.
L_init	User-supplied initial value of loadings in the HPCA; default is the PCA estimator.
F_init	User-supplied initial value of factors in the HPCA; default is the PCA estimator.
maxiter_HPCA	The maximum number of iterations in the HPCA. The default is 100.
maxiter_HLM	The maximum number of iterations in the iterative Huber regression algorithm. The default is 100.
eps	The stopping criterion parameter in the HPCA. The default is 1e-3.

Details

See He et al. (2023) for details.

Value

rhat The estimated factor number.

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

He Y, Li L, Liu D, Zhou W., 2023 Huber Principal Component Analysis for Large-dimensional Factor Models.

Examples

```
set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

HPCA_FN(X,8,Method="E")
```

```
HPCA_FN(X,8,Method="P")
```

IQR

Iterative Quantile Regression Methods for Quantile Factor Models

Description

This function is to fit the quantile factor model via the Iterative Quantile Regression (IQR) algorithm.

Usage

```
IQR(X, r, tau, L_init = NULL, F_init = NULL, max_iter = 100, eps = 0.001)
```

Arguments

<code>X</code>	Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .
<code>r</code>	A positive integer indicating the factor numbers.
<code>tau</code>	The user-supplied quantile level.
<code>L_init</code>	User-supplied initial value of loadings; default is the PCA estimator.
<code>F_init</code>	User-supplied initial value of factors; default is the PCA estimator.
<code>max_iter</code>	The maximum number of iterations. The default is 100.
<code>eps</code>	The stopping criterion parameter. The default is $1e-06$.

Details

See Chen et al. (2021) and He et al. (2023) for details.

Value

The return value is a list. In this list, it contains the following:

<code>Fhat</code>	The estimated factor matrix of dimension $T \times r$.
<code>Lhat</code>	The estimated loading matrix of dimension $N \times r$.
<code>t</code>	The number of iterations.

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

Chen, L., Dolado, J.J., Gonzalo, J., 2021. Quantile factor models. *Econometrica* 89, 875–910.
He Y, Kong X, Yu L, Zhao P., 2023 Quantile factor analysis for large-dimensional time series with statistical guarantee <arXiv:2006.08214>.

Examples

```

set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

tau=0.5
fit=IQR(X,r,tau)
fit$Fhat;fit$Lhat

```

IQR_FN	<i>Estimating Factor Numbers via Rank Minimization Corresponding to IQR</i>
--------	---

Description

This function is to estimate factor numbers via rank minimization corresponding to Iterative Quantile Regression (IQR).

Usage

```

IQR_FN(X, rmax, tau, threshold = NULL, L_init = NULL, F_init = NULL,
       max_iter = 100, eps = 10^(-6))

```

Arguments

<code>X</code>	Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .
<code>rmax</code>	The user-supplied maximum factor numbers.
<code>tau</code>	The user-supplied quantile level.
<code>threshold</code>	The threshold of rank minimization; default is NULL.
<code>L_init</code>	User-supplied initial value of loadings in the IQR; default is the PCA estimator.
<code>F_init</code>	User-supplied initial value of factors in the IQR; default is the PCA estimator.
<code>max_iter</code>	The maximum number of iterations. The default is 100.
<code>eps</code>	The stopping criterion parameter of the IQR method. The default is 1e-06.

Details

See Chen et al. (2021) for more details.

Value

<code>rhat</code>	The estimated factor number.
-------------------	------------------------------

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

Chen, L., Dolado, J.J., Gonzalo, J., 2021. Quantile factor models. *Econometrica* 89, 875–910.

Examples

```
set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

tau=0.5
IQR_FN(X,8,tau)
```

PCA

*Principal Component Analysis for Large-Dimensional Factor Models***Description**

This function is to fit the factor models via Principal Component Analysis (PCA) methods.

Usage

```
PCA(X, r, constraint = "L")
```

Arguments

<code>X</code>	Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .
<code>r</code>	A positive integer indicating the factor numbers.
<code>constraint</code>	The type of identification condition. If <code>constraint="L"</code> , the columns of the estimated loading matrix are orthogonal and <code>constraint="F"</code> indicates the columns of the estimated factor matrix are orthogonal.

Details

See Bai (2003) for details.

Value

The return value is a list. In this list, it contains the following:

<code>Fhat</code>	The estimated factor matrix of dimension $T \times r$.
<code>Lhat</code>	The estimated loading matrix of dimension $N \times r$.

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

Bai, J., 2003. Inferential theory for factor models of large dimensions. *Econometrica* 71, 135–171.

Examples

```
set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

fit=PCA(X,3,"L")
t(fit$Lhat)%*%fit$Lhat/N

fit=PCA(X,3,"F")
t(fit$Fhat)%*%fit$Fhat/T
```

PCA_FN	<i>Estimating Factor Numbers via Eigenvalue Ratios Corresponding to PCA</i>
--------	---

Description

This function is to estimate factor numbers via eigenvalue ratios corresponding to Principal Component Analysis (PCA).

Usage

```
PCA_FN(X, rmax)
```

Arguments

X	Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .
rmax	The user-supplied maximum factor numbers.

Details

See Ahn and Horenstein (2013) for details.

Value

rhat	The estimated factor numbers.
------	-------------------------------

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

Ahn, S.C., Horenstein, A.R., 2013. Eigenvalue ratio test for the number of factors. *Econometrica* 81, 1203–1227.

Examples

```
set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

PCA_FN(X,8)
```

RTS

Robust Two Step Algorithm for Large-Dimensional Elliptical Factor Models

Description

This function is to fit the large-dimensional elliptical factor models via the Robust Two Step (RTS) algorithm.

Usage

RTS(X, r)

Arguments

X Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .

r A positive integer indicating the factor numbers.

Details

See He et al. (2022) for details.

Value

The return value is a list. In this list, it contains the following:

Fhat The estimated factor matrix of dimension $T \times r$.

Lhat The estimated loading matrix of dimension $N \times r$.

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

He, Y., Kong, X., Yu, L., Zhang, X., 2022. Large-dimensional factor analysis without moment constraints. *Journal of Business & Economic Statistics* 40, 302–312.

Examples

```
set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

fit=RTS(X,3)
fit$Fhat;fit$Lhat
```

RTS_FN

Estimating Factor Numbers Robustly via Multivariate Kendall's Tau Eigenvalue Ratios

Description

This function is to estimate factor numbers robustly via multivariate Kendall's tau eigenvalue ratios.

Usage

```
RTS_FN(X, rmax)
```

Arguments

X	Input matrix, of dimension $T \times N$. Each row is an observation with N features at time point t .
rmax	The user-supplied maximum factor numbers.

Details

See Yu et al. (2019) for details.

Value

rhat	The estimated factor number.
------	------------------------------

Author(s)

Yong He, Lingxiao Li, Dong Liu, Wenxin Zhou.

References

Yu, L., He, Y., Zhang, X., 2019. Robust factor number specification for large-dimensional elliptical factor model. *Journal of Multivariate analysis* 174, 104543.

Examples

```
set.seed(1)
T=50;N=50;r=3
L=matrix(rnorm(N*r,0,1),N,r);F=matrix(rnorm(T*r,0,1),T,r)
E=matrix(rnorm(T*N,0,1),T,N)
X=F%*%t(L)+E

RTS_FN(X,8)
```

Index

HPCA, [2](#)
HPCA_FN, [3](#)

IQR, [5](#)
IQR_FN, [6](#)

PCA, [7](#)
PCA_FN, [8](#)

RTS, [9](#)
RTS_FN, [10](#)