Package ‘RRRR’

July 21, 2025

Type Package

Title Online Robust Reduced-Rank Regression Estimation

Version 1.1.1

Description Methods for estimating online robust reduced-rank regression.
The Gaussian maximum likelihood estimation method is described in Johansen, S. (1991) <doi:10.2307/2938278>.
The majorisation-minimisation estimation method is partly described in Zhao, Z., & Palomar, D. P. (2017) <doi:10.1109/GlobalSIP.2017.8309093>.
The description of the generic stochastic successive upper-bound minimisation method and the sample average approximation can be found in Razaviyayn, M., Sanjabi, M., & Luo, Z. Q. (2016) <doi:10.1007/s10107-016-1021-7>.

License GPL-3

Encoding UTF-8

URL `https://pkg.yangzhuoranyang.com/RRRR/`,
`https://github.com/FinYang/RRRR`

BugReports `https://github.com/FinYang/RRRR/issues/`

Imports matrixcalc, expm, ggplot2, magrittr, mvtnorm, stats

Suggests lazybar, knitr, rmarkdown

RoxygenNote 7.2.3

Language en-AU

VignetteBuilder knitr

NeedsCompilation no

Author Yangzhuoran Fin Yang [aut, cre] (ORCID:
<https://orcid.org/0000-0002-1232-8017>),
Ziping Zhao [aut] (ORCID: <https://orcid.org/0000-0002-8668-6263>)

Maintainer Yangzhuoran Fin Yang <yangyangzhuoran@gmail.com>

Repository CRAN

Date/Publication 2023-02-24 09:02:29 UTC

Contents

RRRR-package *Online Robust Reduced-Rank Regression Estimation*

Description

Methods for estimating online Robust Reduced-Rank Regression.

Author(s)

Yangzhuoran Yang. <yangyangzhuoran@gmail.com>

Ziping Zhao. <zhaoziping@shanghaitech.edu.cn>

ORRRR *Online Robust Reduced-Rank Regression*

Description

Online robust reduced-rank regression with two major estimation methods:

SMM Stochastic Majorisation-Minimisation

SAA Sample Average Approximation

Usage

```
ORRRR(
  y,
  x,
  z = NULL,
  mu = TRUE,
  r = 1,
  initial_size = 100,
  addon = 10,
  method = c("SMM", "SAA"),
  SAAmethod = c("optim", "MM"),
```

```
  ...,
  initial_A = matrix(rnorm(P * r), ncol = r),
  initial_B = matrix(rnorm(Q * r), ncol = r),
  initial_D = matrix(rnorm(P * R), ncol = R),
  initial_mu = matrix(rnorm(P)),
  initial_Sigma = diag(P),
  ProgressBar = requireNamespace("lazybar"),
  return_data = TRUE
)
```

Arguments

y	Matrix of dimension N*P. The matrix for the response variables. See Detail.
x	Matrix of dimension N*Q. The matrix for the explanatory variables to be projected. See Detail.
z	Matrix of dimension N*R. The matrix for the explanatory variables not to be projected. See Detail.
mu	Logical. Indicating if a constant term is included.
r	Integer. The rank for the reduced-rank matrix AB'. See Detail.
initial_size	Integer. The number of data points to be used in the first iteration.
addon	Integer. The number of data points to be added in the algorithm in each iteration after the first.
method	Character. The estimation method. Either "SMM" or "SAA". See Description and Detail.
SAAmethod	Character. The sub solver used in each iteration when the method is chosen to be "SAA". See Detail.
...	Additional arguments to function optim when the method is "SAA" and the SAAmethod is "optim" RRRR when the method is "SAA" and the SAAmethod is "MM"
initial_A	Matrix of dimension P*r. The initial value for matrix A. See Detail.
initial_B	Matrix of dimension Q*r. The initial value for matrix B. See Detail.
initial_D	Matrix of dimension P*R. The initial value for matrix D. See Detail.
initial_mu	Matrix of dimension P*1. The initial value for the constant mu. See Detail.
initial_Sigma	Matrix of dimension P*P. The initial value for matrix Sigma. See Detail.
ProgressBar	Logical. Indicating if a progress bar is shown during the estimation process. The progress bar requires package lazybar to work.
return_data	Logical. Indicating if the data used is return in the output. If set to TRUE, update.RRRR can update the model by simply provide new data. Set to FALSE to save output size.

Details

The formulation of the reduced-rank regression is as follow:

$$y = \mu + AB'x + Dz + innov,$$

where for each realization y is a vector of dimension P for the P response variables, x is a vector of dimension Q for the Q explanatory variables that will be projected to reduce the rank, z is a vector of dimension R for the R explanatory variables that will not be projected, μ is the constant vector of dimension P, $innov$ is the innovation vector of dimension P, D is a coefficient matrix for z with dimension $P * R$, A is the so called exposure matrix with dimension $P * r$, and B is the so called factor matrix with dimension $Q * r$. The matrix resulted from AB' will be a reduced rank coefficient matrix with rank of r. The function estimates parameters μ, A, B, D, and $Sigma$, the covariance matrix of the innovation's distribution.

The algorithm is online in the sense that the data is continuously incorporated and the algorithm can update the parameters accordingly. See `?update.RRRR` for more details.

At each iteration of SAA, a new realisation of the parameters is achieved by solving the minimisation problem of the sample average of the desired objective function using the data currently incorporated. This can be computationally expensive when the objective function is highly nonconvex. The SMM method overcomes this difficulty by replacing the objective function by a well-chosen majorising surrogate function which can be much easier to optimise.

SMM method is robust in the sense that it assumes a heavy-tailed Cauchy distribution for the innovations.

Value

A list of the estimated parameters of class `ORRRR`.

method The estimation method being used

SAAmethod If SAA is the major estimation method, what is the sub solver in each iteration.

spec The input specifications. N is the sample size.

history The path of all the parameters during optimization and the path of the objective value.

mu The estimated constant vector. Can be `NULL`.

A The estimated exposure matrix.

B The estimated factor matrix.

D The estimated coefficient matrix of `z`.

Sigma The estimated covariance matrix of the innovation distribution.

obj The final objective value.

data The data used in estimation if `return_data` is set to `TRUE`. `NULL` otherwise.

Author(s)

Yangzhuoran Yang

See Also

`update.RRRR`, `RRRR`, `RRR`

Examples

```
set.seed(2222)
data <- RRR_sim()
res <- ORRRR(y=data$y, x=data$x, z = data$z)
res
```

---

plot.RRRR *Plot Objective value of a Robust Reduced-Rank Regression*

Description

Plot Objective value of a Robust Reduced-Rank Regression

Usage

```
## S3 method for class 'RRRR'
plot(x, aes_x = c("iteration", "runtime"), xlog10 = TRUE, ...)
```

Arguments

x	An RRRR object.
aes_x	Either "iteration" or "runtime". The x axis in the plot.
xlog10	Logical, indicates whether the scale of x axis is log 10 transformed.
...	Additional argument to ggplot2.

Value

An ggplot2 object

Author(s)

Yangzhuoran Fin Yang

Examples

```
set.seed(2222)
data <- RRR_sim()
res <- RRRR(y=data$y, x=data$x, z = data$z)
plot(res)
```

RRR *Reduced-Rank Regression using Gaussian MLE*

Description

Gaussian Maximum Likelihood Estimation method for Reduced-Rank Regression. This method is not robust in the sense that it assumes a Gaussian distribution for the innovations which does not take into account the heavy-tailedness of the true distribution and outliers.

Usage

```
RRR(y, x, z = NULL, mu = TRUE, r = 1)
```

Arguments

y	Matrix of dimension N*P. The matrix for the response variables. See `Detail`.
x	Matrix of dimension N*Q. The matrix for the explanatory variables to be projected. See `Detail`.
z	Matrix of dimension N*R. The matrix for the explanatory variables not to be projected. See `Detail`.
mu	Logical. Indicating if a constant term is included.
r	Integer. The rank for the reduced-rank matrix AB'. See `Detail`.

Details

The formulation of the reduced-rank regression is as follow:

$$y = \mu + AB'x + Dz + innov,$$

where for each realization y is a vector of dimension P for the P response variables, x is a vector of dimension Q for the Q explanatory variables that will be projected to reduce the rank, z is a vector of dimension R for the R explanatory variables that will not be projected, μ is the constant vector of dimension P, $innov$ is the innovation vector of dimension P, D is a coefficient matrix for z with dimension $P * R$, A is the so called exposure matrix with dimension $P * r$, and B is the so called factor matrix with dimension $Q*r$. The matrix resulted from AB' will be a reduced rank coefficient matrix with rank of r. The function estimates parameters μ, A, B, D, and $Sigma$, the covariance matrix of the innovation's distribution, assuming the innovation has a Gaussian distribution.

Value

A list of the estimated parameters of class `RRR`.

spec The input specifications. N is the sample size.

mu The estimated constant vector. Can be `NULL`.

A The estimated exposure matrix.

B The estimated factor matrix.

D The estimated coefficient matrix of `z`. Can be `NULL`.

Sigma The estimated covariance matrix of the innovation distribution.

Author(s)

Yangzhuoran Yang

References

S. Johansen, "Estimation and Hypothesis Testing of Cointegration Vectors in Gaussian Vector Autoregressive Models,"Econometrica, vol. 59,p. 1551, Nov. 1991.

See Also

For robust reduced-rank regression estimation see function RRRR.

Examples

```
set.seed(2222)
data <- RRR_sim()
res <- RRR(y=data$y, x=data$x, z = data$z)
res
```

RRRR *Robust Reduced-Rank Regression using Majorisation-Minimisation*

Description

Majorisation-Minimisation based Estimation for Reduced-Rank Regression with a Cauchy Distribution Assumption. This method is robust in the sense that it assumes a heavy-tailed Cauchy distribution for the innovations. This method is an iterative optimization algorithm. See `References` for a similar setting.

Usage

```
RRRR(
  y,
  x,
  z = NULL,
  mu = TRUE,
  r = 1,
  itr = 100,
  earlystop = 1e-04,
  initial_A = matrix(rnorm(P * r), ncol = r),
  initial_B = matrix(rnorm(Q * r), ncol = r),
  initial_D = matrix(rnorm(P * R), ncol = R),
  initial_mu = matrix(rnorm(P)),
  initial_Sigma = diag(P),
  return_data = TRUE
)
```

Arguments

y	Matrix of dimension N*P. The matrix for the response variables. See `Detail`.
x	Matrix of dimension N*Q. The matrix for the explanatory variables to be projected. See `Detail`.
z	Matrix of dimension N*R. The matrix for the explanatory variables not to be projected. See `Detail`.
mu	Logical. Indicating if a constant term is included.
r	Integer. The rank for the reduced-rank matrix AB'. See `Detail`.
itr	Integer. The maximum number of iteration.
earlystop	Scalar. The criteria to stop the algorithm early. The algorithm will stop if the improvement on objective function is small than $earlystop * objective_from_last_iteration$.
initial_A	Matrix of dimension P*r. The initial value for matrix A. See `Detail`.
initial_B	Matrix of dimension Q*r. The initial value for matrix B. See `Detail`.
initial_D	Matrix of dimension P*R. The initial value for matrix D. See `Detail`.
initial_mu	Matrix of dimension P*1. The initial value for the constant mu. See `Detail`.
initial_Sigma	Matrix of dimension P*P. The initial value for matrix Sigma. See `Detail`.
return_data	Logical. Indicating if the data used is return in the output. If set to `TRUE`, `update.RRRR` can update the model by simply provide new data. Set to `FALSE` to save output size.

Details

The formulation of the reduced-rank regression is as follow:

$$y = \mu + AB'x + Dz + innov,$$

where for each realization y is a vector of dimension P for the P response variables, x is a vector of dimension Q for the Q explanatory variables that will be projected to reduce the rank, z is a vector of dimension R for the R explanatory variables that will not be projected, μ is the constant vector of dimension P, $innov$ is the innovation vector of dimension P, D is a coefficient matrix for z with dimension $P * R$, A is the so called exposure matrix with dimension $P * r$, and B is the so called factor matrix with dimension $Q * r$. The matrix resulted from AB' will be a reduced rank coefficient matrix with rank of r. The function estimates parameters μ, A, B, D, and $Sigma$, the covariance matrix of the innovation's distribution, assuming the innovation has a Cauchy distribution.

Value

A list of the estimated parameters of class `RRRR`.

spec The input specifications. N is the sample size.

history The path of all the parameters during optimization and the path of the objective value.

mu The estimated constant vector. Can be `NULL`.

A The estimated exposure matrix.

B The estimated factor matrix.

D The estimated coefficient matrix of z.

Sigma The estimated covariance matrix of the innovation distribution.

obj The final objective value.

data The data used in estimation if `return_data` is set to `TRUE`. `NULL` otherwise.

Author(s)

Yangzhuoran Yang

References

Z. Zhao and D. P. Palomar, "Robust maximum likelihood estimation of sparse vector error correction model," in2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP), pp. 913–917,IEEE, 2017.

Examples

```
set.seed(2222)
data <- RRR_sim()
res <- RRRR(y=data$y, x=data$x, z = data$z)
res
```

RRR_sim *Simulating data for Reduced-Rank Regression*

Description

Simulate data for Reduced-rank regression. See `Detail` for the formulation of the simulated data.

Usage

```
RRR_sim(
  N = 1000,
  P = 3,
  Q = 3,
  R = 1,
  r = 1,
  mu = rep(0.1, P),
  A = matrix(rnorm(P * r), ncol = r),
  B = matrix(rnorm(Q * r), ncol = r),
  D = matrix(rnorm(P * R), ncol = R),
  varcov = diag(P),
  innov = mvtnorm::rmvt(N, sigma = varcov, df = 3),
  mean_x = 0,
  mean_z = 0,
  x = NULL,
  z = NULL
)
```

Arguments

N	Integer. The total number of simulated realizations.
P	Integer. The dimension of the response variable matrix. See `Detail`.
Q	Integer. The dimension of the explanatory variable matrix to be projected. See `Detail`.
R	Integer. The dimension of the explanatory variable matrix not to be projected. See `Detail`.
r	Integer. The rank of the reduced rank coefficient matrix. See `Detail`.
mu	Vector with length P. The constants. Can be `NULL` to drop the term. See `Detail`.
A	Matrix with dimension P*r. The exposure matrix. See `Detail`.
B	Matrix with dimension Q*r. The factor matrix. See `Detail`.
D	Matrix with dimension P*R. The coefficient matrix for `z`. Can be `NULL` to drop the term. See `Detail`.
varcov	Matrix with dimension P*P. The covariance matrix of the innovation. See `Detail`.
innov	Matrix with dimension N*P. The innovations. Default to be simulated from a Student t distribution, See `Detail`.
mean_x	Integer. The mean of the normal distribution x is simulated from.
mean_z	Integer. The mean of the normal distribution z is simulated from.
x	Matrix with dimension N*Q. Can be used to specify x instead of simulating form a normal distribution.
z	Matrix with dimension N*R. Can be used to specify z instead of simulating form a normal distribution.

Details

The data simulated can be used for the standard reduced-rank regression testing with the following formulation

$$y = \mu + AB'x + Dz + innov,$$

where for each realization y is a vector of dimension P for the P response variables, x is a vector of dimension Q for the Q explanatory variables that will be projected to reduce the rank, z is a vector of dimension R for the R explanatory variables that will not be projected, μ is the constant vector of dimension P, $innov$ is the innovation vector of dimension P, D is a coefficient matrix for z with dimension $P * R$, A is the so called exposure matrix with dimension $P * r$, and B is the so called factor matrix with dimension $Q * r$. The matrix resulted from AB' will be a reduced rank coefficient matrix with rank of r. The function simulates x, z from multivariate normal distribution and y by specifying parameters μ, A, B, D, and $varcov$, the covariance matrix of the innovation's distribution. The constant μ and the term Dz can be dropped by setting `NULL` for arguments `mu` and `D`. The `innov` in the argument is the collection of innovations of all the realizations.

Value

A list of the input specifications and the data y, x, and z, of class `RRR_data`.

y Matrix of dimension N*P

x Matrix of dimension N*Q

z Matrix of dimension N*R

Author(s)

Yangzhuoran Yang

Examples

```
set.seed(2222)
data <- RRR_sim()
```

`update.RRRR` *Update the RRRR/ORRRR type model with addition data*

Description

`update.RRRR` will update online robust reduced-rank regression model with class `RRRR(ORRRR)` using newly added data to achieve online estimation. Estimation methods:

SMM Stochastic Majorisation-Minimisation

SAA Sample Average Approximation

Usage

```
## S3 method for class 'RRRR'
update(
  object,
  newy,
  newx,
  newz = NULL,
  addon = object$spec$addon,
  method = object$method,
  SAAmethod = object$SAAmethod,
  ...,
  ProgressBar = requireNamespace("lazybar")
)
```

Arguments

`object`	A model with class `RRRR(ORRRR)`
`newy`	Matrix of dimension N*P, the new data y. The matrix for the response variables. See `Detail`.
`newx`	Matrix of dimension N*Q, the new data x. The matrix for the explanatory variables to be projected. See `Detail`.
`newz`	Matrix of dimension N*R, the new data z. The matrix for the explanatory variables not to be projected. See `Detail`.
`addon`	Integer. The number of data points to be added in the algorithm in each iteration after the first.

`method`	Character. The estimation method. Either "SMM" or "SAA". See `Description`.
`SAAmethod`	Character. The sub solver used in each iteration when the `methid` is chosen to be "SAA". See `Detail`.
`...`	Additional arguments to function `optim` when the `method` is "SAA" and the `SAAmethod` is "optim" `RRRR` when the `method` is "SAA" and the `SAAmethod` is "MM"
`ProgressBar`	Logical. Indicating if a progress bar is shown during the estimation process. The progress bar requires package `lazybar` to work.

Details

The formulation of the reduced-rank regression is as follow:

$$y = \mu + AB'x + Dz + innov,$$

where for each realization y is a vector of dimension P for the P response variables, x is a vector of dimension Q for the Q explanatory variables that will be projected to reduce the rank, z is a vector of dimension R for the R explanatory variables that will not be projected, μ is the constant vector of dimension P, $innov$ is the innovation vector of dimension P, D is a coefficient matrix for z with dimension $P * R$, A is the so called exposure matrix with dimension $P * r$, and B is the so called factor matrix with dimension $Q * r$. The matrix resulted from AB' will be a reduced rank coefficient matrix with rank of r. The function estimates parameters μ, A, B, D, and $Sigma$, the covariance matrix of the innovation's distribution.

See `?ORRRR` for details about the estimation methods.

Value

A list of the estimated parameters of class `ORRRR`.

method The estimation method being used

SAAmethod If SAA is the major estimation method, what is the sub solver in each iteration.

spec The input specifications. N is the sample size.

history The path of all the parameters during optimization and the path of the objective value.

mu The estimated constant vector. Can be `NULL`.

A The estimated exposure matrix.

B The estimated factor matrix.

D The estimated coefficient matrix of `z`.

Sigma The estimated covariance matrix of the innovation distribution.

obj The final objective value.

data The data used in estimation.

Author(s)

Yangzhuoran Yang

See Also

ORRRR, RRRR, RRR

Examples

```
set.seed(2222)
data <- RRR_sim()
newdata <- RRR_sim(A = data$spec$A,
                   B = data$spec$B,
                   D = data$spec$D)
res <- ORRRR(y=data$y, x=data$x, z = data$z)
res <- update(res, newy=newdata$y, newx=newdata$x, newz=newdata$z)
res
```

Index