Package 'VARshrink'

June 18, 2026

Title Shrinkage Estimation Methods for Vector Autoregressive Models

Version 0.5.0

Description Vector autoregressive (VAR) model is a fundamental and effective approach for multivariate time series analysis. Shrinkage estimation methods can be applied to high-dimensional VAR models with dimensionality greater than the number of observations, contrary to the standard ordinary least squares method. This package is an integrative package delivering nonparametric, parametric, and semiparametric methods in a unified and consistent manner, such as the multivariate ridge regression in Golub, Heath, and Wahba (1979) <doi:10.2307/1268518>, a James-Stein type nonparametric shrinkage method in Opgen-Rhein and Strimmer (2007) <doi:10.1186/1471-2105-8-S2-S3>, and Bayesian estimation methods using noninformative and informative priors in Lee, Choi, and S.-H. Kim (2016) <doi:10.1016/j.csda.2016.03.007> and Ni and Sun (2005) <doi:10.1198/073500104000000622>.

License GPL (>= 3)

Depends R (>= 3.6.0), vars (>= 1.6.1)

Imports ars (>= 0.6), corpcor (>= 1.6.9), strucchange, stats, MASS, mvtnorm, MCMCpack

Suggests knitr, rmarkdown, kableExtra, devtools, lintr, testthat (>= 3.0.0)

URL https://github.com/namgillee/VARshrink/

BugReports https://github.com/namgillee/VARshrink/issues/

Encoding UTF-8

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

Config/testthat/edition 3

NeedsCompilation no

Author Namgil Lee [aut, cre, cph] (ORCID: <https://orcid.org/0000-0003-0593-9028>), Heon Young Yang [ctb], Sung-Ho Kim [aut, cph]

Maintainer Namgil Lee <namgil.lee@kangwon.ac.kr>

Repository CRAN

Date/Publication 2026-06-18 07:50:02 UTC

Contents

`Acoef_sh` *Coefficient matrices of the lagged endogenous variables*

Description

Returns the estimated coefficient matrices of the lagged endogenous variables as a list of $(K \times K)$ matrices.

Usage

```
Acoef_sh(x)
```

Arguments

`x`	An object of class "varshrinkeset", generated by VARshrink().

Details

Consider VAR(p) model:

$$\mathbf{y}_t = \mathbf{A}_1\mathbf{y}_{t-1} + ... + \mathbf{A}_p\mathbf{y}_{t-p} + \mathbf{C}\mathbf{d}_t + \mathbf{e}_t.$$

The function returns the $(K \times K)$ matrices $\mathbf{A}_1, ..., \mathbf{A}_p$ as a list object.

This function modifies `vars::Acoef()` for the class "varshrinkest", preventing redundant copying of data matrix objects.

Value

A list object with K-by-K VAR coefficient matrices $\mathbf{A}_1, ..., \mathbf{A}_p$.

See Also

`Acoef`

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
estim <- VARshrink(y, p = 2, type = "const", method = "ridge")
Acoef_sh(estim)
```

Bcoef_sh *Coefficient matrix of an estimated VAR(p)*

Description

Returns the estimated coefficients of a VAR(p) model as a matrix.

Usage

```
Bcoef_sh(x)
```

Arguments

x	An object of class "varshrinkest" generated by `VARshrink()`.

Details

Consider VAR(p) model:

$$\mathbf{y}_t = \mathbf{A}_1\mathbf{y}_{t-1} + ... + \mathbf{A}_p\mathbf{y}_{t-p} + \mathbf{C}\mathbf{d}_t + \mathbf{e}_t.$$

The function returns the concatenated matrix $(\mathbf{A}_1, ..., \mathbf{A}_p, \mathbf{C})$ as a matrix object.

This function modifies `vars::Bcoef()` for the class "varshrinkest", preventing redundant copying of data matrix objects.

Value

A matrix holding the estimated coefficients of a VAR.

See Also

Bcoef

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
estim <- VARshrink(y, p = 2, type = "const", method = "ridge")
Bcoef_sh(estim)
```

---

convPsi2varresult *Convert format for VAR coefficients from Psi to varresult*

Description

Convert a matrix of VAR coefficients estimated by a shrinkage method into a list of "shrinklm" object, where the class "shrinklm" inherits the class "lm".

Usage

```
convPsi2varresult(
  Psi,
  Y,
  X,
  lambda0,
  type = c("const", "trend", "both", "none"),
  ybar = NULL,
  xbar = NULL,
  Q_values = NULL,
  noise_variances = NULL,
  callstr = ""
)
```

Arguments

Psi	An M-by-K matrix of VAR coefficients
Y	An N-by-K data matrix of dependent variables
X	An N-by-M data matrix of regressors
lambda0	A rescaled shrinkage intensity parameter, based on which the effective number of parameters is computed by $$Trace(X(X'X + \lambda_0 * I)^{-1}X')$$

type	Type of deterministic variables in the VAR estimation problem. Either of "const", "trend", "both", or "none".
ybar, xbar	NULL if Y and X are not centered. Mean vectors if Y and X had been centered. If Y and X had been centered (ybar and xbar are not NULL) and type is "const" or "both", then the coefficients for the constant term is computed and concatenated to the coefficients.
Q_values	Nonnegative weight vector of length N. Default is NULL. Take weights on rows (samples) of Y and X by sqrt(Q).
noise_variances	A length-K vector for variances of noises. If NULL, it is assumed as a vector of one's.
callstr	The call to VARshrink().

Details

Consider VAR(p) model:

$$y_t = A_1 y_{t-1} + ... + A_p y_{t-p} + C d_t + e_t.$$

It can be written in the matrix form:

$$Y = X\Psi + E,$$

where Ψ is a concatenated M-by-K matrix, $\Psi = (A_1, ..., A_p, C)^T$. It can be written in the multiple linear regression form of a VAR(p) model:

$$y_j = X\psi_j + e_j, \quad j = 1, ..., K,$$

where y_j, ψ_j, and e_j are the j-th column vectors of Y, Ψ, and E, respectively. This function converts Ψ into a list of "shrinklm" objects, where each "shrinklm" object contains the length-M vector ψ_j as coefficients.

Considering that each coefficient vector ψ_j is estimated by a shrinkage method, the effective number of parameters, k_{eff}, is computed as:

$$k_{eff} = Trace(X(X^T X + \lambda_0 * I)^{-1} X^T).$$

Then, the degree of freedom of residuals is computed as:

$$df.residual = N - k_{eff},$$

where N is the number of rows of data matrices Y and X.

Value

A list object with objects of class c("shrinklm", "lm"). Each "shrinklm" object has components: coefficients, residuals, fitted.values, rank, df.residual, lambda0, call, terms, svd

createVARCoefs_ltriangular *Create coefficients of a VAR model*

Description

Randomly create sparse lower-triangular matrices for VAR coefficients of lagged endogenous variables, and set a constant vector.

Usage

```
createVARCoefs_ltriangular(
  p = 1,
  K = 5,
  diag_val = 1/p,
  num_nonzero = 0,
  const_vector = NULL,
  range_min = 0.2,
  range_max = 1/p
)
```

Arguments

p	lag order
K	Number of time series variables.
diag_val	diagonal values of $A_1, \ldots, A_p$.
num_nonzero	Number of nonzero entries on the lower-triangular parts of A1, ..., Ap
const_vector	constant vector c of the VAR model
range_min, range_max	Each nonzero off-diagonal entry of coefficient matrices is drawn uniformly from the interval [-range_max, -range_min] U [range_min, range_max]

Details

Consider VAR(p) model:

$$y_t = A_1 y_{t-1} + \cdots + A_p y_{t-p} + c + e_t,$$

with the constant deterministic variable ($d_t = 1$). The function creates the coefficient matrices $A_1, \ldots, A_p$ and constant vector c.

Diagonal elements of each K-by-K matrix A_i are all equal to diag_val, and off-diagonal elements are all zero except for a few randomly selected nonzero elements. Nonzero off-diagonal elements are selected from lower-triangular parts of A_i and the values are drawn from a uniform distribution over [-range_max, -range_min] U [range_min, range_max].

Value

A list object with components $A and $c. $A is a list of K-by-K matrices $A_1, \ldots, A_p$, and $c is a constant vector of length K.

Examples

```
p <- 1; K <- 20;
const_vector <- c(rep(0.2, 5), rep(0.7, 15))
createVARCoefs_ltriangular(p = p, K = K, diag_val = 0.6,
num_nonzero = K, const_vector = const_vector, range_max = 1)
```

---

irf.varshrinkest *Impulse response function*

Description

Computes the impulse response coefficients of a VAR(p) (or transformed VECM to VAR(p)) for n.ahead steps.

Usage

```
## S3 method for class 'varshrinkest'
irf(
  x,
  impulse = NULL,
  response = NULL,
  n.ahead = 10,
  ortho = TRUE,
  cumulative = FALSE,
  boot = TRUE,
  ci = 0.95,
  runs = 100,
  seed = NULL,
  ...
)
```

Arguments

x	Object of class "varshrinkest", generated by VARshrink()
impulse	A character vector of the impulses, default is all variables.
response	A character vector of the responses, default is all variables.
n.ahead	Integer specifying the steps.
ortho	Logical, if TRUE (the default) the orthogonalised impulse response coefficients are computed (only for objects of class 'varshrinkest').
cumulative	Logical, if TRUE the cumulated impulse response coefficients are computed. The default value is false.

boot	Logical, if TRUE (the default) bootstrapped confidence bands or Bayesian credible bands for the impulse response coefficients are computed.
ci	Numeric, the confidence level for the bootstrapped confidence or Bayesian credible bands.
runs	An integer, specifying the runs for the bootstrap.
seed	An integer, specifying the seed for the rng of the bootstrap.
...	Currently not used.

Details

Confidence intervals are computed by bootstrap methods in `vars::irf()` or by Bayesian posterior distributions for an object of class "varshrinkest".

For objects without stored MCMC draws, this method delegates to `vars::irf()`. For objects with `mcmc.param`, point estimates are computed from the fitted VAR coefficients and credible bands are computed from posterior MCMC draws.

Value

An object of class "varirf" with the same component structure as `vars::irf()`.

See Also

`irf`

---

lm_full_Bayes *Full Bayesian Shrinkage Estimation Method for Multivariate Regression*

Description

Estimate regression coefficients and scale matrix for noise by using Gibbs MCMC algorithm. The function assumes 1) multivariate t-distribution for noise as a sampling distribution, and 2) noninformative priors for regression coefficients and scale matrix for noise.

Usage

```
lm_full_Bayes(
  Y,
  X,
  dof = Inf,
  prior_type = "NCJ",
  burnincycle = 1000,
  mcmccycle = 2000,
  store_mcmc = TRUE
)
```

Arguments

`Y`	An N x K matrix of dependent variables.
`X`	An N x M matrix of regressors.
`dof`	Degree of freedom for multivariate t-distribution. If `dof = Inf` (default), then multivariate normal distribution is applied and weight vector q is not estimated. If `dof = NULL` or `dof <= 0`, then `dof` and q are estimated automatically. If `dof` is a positive number, q is estimated.
`prior_type`	If "NCJ" (default), use shrinkage-reference prior. If "CJ", use shrinkage-inverse Wishart prior.
`burnincycle, mcmccycle`	Number of burnin cycles is the number of initially generated sample values to drop. Number of MCMC cycles is the number of generated sample values to compute estimates.
`store_mcmc`	TRUE to return all MCMC chain of estimated parameters across mcmccycle. Default is TRUE.

Details

Consider the multivariate regression:

$$Y = X\Psi + e, \quad e \sim MVT(0, \nu, \Sigma).$$

Ψ is a M-by-K matrix of regression coefficients and Σ is a K-by-K scale matrix for multivariate t-distribution for noise.

Sampling distribution for noise e is multivariate t-distribution with degree of freedom dof and scale matrix Σ : $e \sim MVT(0, \nu, \Sigma)$. The priors are 1) the shrinkage prior for regression coefficients Ψ, and 2) the reference prior or inverse-Wishart for scale matrix Σ.

The function implements Gibbs MCMC algorithm for estimating regression coefficients Psi and scale matrix Sigma.

Value

A list object with estimated parameters: Psi, Sigma, dof, delta (delta is the reciprocal of lambda), and lambda. Additional components are se.param (standard error of the parameters) and mcmc.param (parameters for whole mcmc chain, stored if store_mcmc is TRUE).

References

S. Ni and D. Sun (2005). Bayesian estimates for vector autoregressive models. Journal of Business & Economic Statistics 23(1), 105-117.

`lm_multiv_ridge` *Multivariate Ridge Regression*

Description

Estimate regression coefficients by using ridge regression.

Usage

```
lm_multiv_ridge(Y, X, lambda = 0, do_scale = FALSE)
```

Arguments

`Y`	An N x K matrix of dependent variables.
`X`	An N x M matrix of regressors.
`lambda`	Numeric vector of lambda values
`do_scale`	If true, X is centered and scaled, and Y is centered.

Details

Consider the multivariate regression:

$$\mathbf{Y} = \mathbf{X}\mathbf{\Psi} + \mathbf{e}.$$

$\mathbf{\Psi}$ is a M-by-K matrix of regression coefficients. The ridge regression estimate for the coefficients is

$$\mathbf{\Psi} = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}'\mathbf{Y}.$$

Value

A list object with the components: 1) Psi - A list of estimated $\mathbf{\Psi}$ matrices, 2) lambda - A vector of λ values, 3) GCV - A vector of GCV values

References

G. H. Golub, M. Heath, G. Wahba (1979). Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics 21(2), 215-223. doi: 10.2307/1268518

`lm_semi_Bayes_PCV` *Semiparametric Bayesian Shrinkage Estimation Method for Multivariate Regression*

Description

Estimate regression coefficients and scale matrix for noise by using a parameterized cross validation (PCV). The function assumes 1) multivariate t-distribution for noise as a sampling distribution, and 2) informative priors for regression coefficients and scale matrix for noise.

Usage

```
lm_semi_Bayes_PCV(
  Y,
  X,
  dof = Inf,
  lambda = NULL,
  lambda_var = NULL,
  prior_type = c("NCJ", "CJ"),
  num_folds = 5,
  m0 = ncol(Y)
)
```

Arguments

`Y`	An N x K matrix of dependent variables.
`X`	An N x M matrix of regressors.
`dof`	Degrees-of-freedom, ν, for multivariate t-distribution. If `dof = Inf` (default), then multivariate normal distribution is applied and weight vector q is not estimated. If `dof = NULL` or a numeric vector, then dof is selected by k-fold CV automatically and q is estimated.
`lambda`	If NULL or a vector of length >=2, it is selected by PCV.
`lambda_var`	If NULL, it is selected by a Stein-type shrinkage method.
`prior_type`	"NCJ" for non-conjugate prior and "CJ" for conjugate prior for scale matrix Sigma.
`num_folds`	Number of folds for PCV.
`m0`	A hyperparameter for inverse Wishart distribution for Sigma

Details

Consider the multivariate regression:

$$\mathbf{Y} = \mathbf{X}\mathbf{\Psi} + \mathbf{e}, \quad \mathbf{e} \sim MVT(0, \nu, \mathbf{\Sigma}).$$

$\mathbf{\Psi}$ is a $(M \times K)$ matrix of regression coefficients and $\mathbf{\Sigma}$ is a $(K \times K)$ scale matrix for multivariate t-distribution for noise.

Sampling distribution for noise $\mathbf{e}$ is the multivariate t-distribution with the degrees-of-freedom ν and scale matrix $\mathbf{\Sigma}$: $\mathbf{e} \sim MVT(0, \nu, \mathbf{\Sigma})$. The priors are informative priors: 1) a shrinkage prior for regression coefficients $\mathbf{Psi}$, and 2) inverse Wishart prior for scale matrix $\mathbf{\Sigma}$, which can be either non-conjugate ("NCJ") or conjugate ("CJ") to the shrinkage prior for coefficients $\mathbf{\Psi}$.

The function implements parameterized cross validation (PCV) for selecting a shrinkage parameter lambda for estimating regression coefficients (0 < lambda <= 1). In addition, the function uses a Stein-type shrinkage method for selecting a shrinkage parameter lambda_var for estimating variances of time series variables.

References

N. Lee, H. Choi, and S.-H. Kim (2016). Bayes shrinkage estimation for high-dimensional VAR models with scale mixture of normal distributions for noise. Computational Statistics & Data Analysis 101, 250-276. doi: 10.1016/j.csda.2016.03.007

`lm_ShVAR_KCV` *K-fold Cross Validation for Selection of Shrinkage Parameters of Semiparametric Bayesian Shrinkage Estimator for Multivariate Regression*

Description

Estimate regression coefficients and scale matrix for noise by using semiparametric Bayesian shrinkage estimator, whose shrinkage parameters are selected by k-fold cross validation (KCV).

Usage

```
lm_ShVAR_KCV(
  Y,
  X,
  dof = Inf,
  lambda = NULL,
  lambda_var = NULL,
  prior_type = c("NCJ", "CJ"),
  num_folds = 5,
  m0 = ncol(Y)
)
```

Arguments

`Y`	An N x K matrix of dependent variables.
`X`	An N x M matrix of regressors.
`dof`	Degree of freedom for multivariate t-distribution. If `dof = Inf` (default), then multivariate normal distribution is applied and weight vector q is not estimated. If `dof = NULL` or a numeric vector, then `dof` is selected by k-fold CV automatically and q is estimated.

lambda	If NULL or a vector of length >=2, it is selected by KCV.
lambda_var	If NULL or a vector of length >=2, it is selected by KCV.
prior_type	"NCJ" for non-conjugate prior and "CJ" for conjugate prior for scale matrix Sigma.
num_folds	Number of folds for KCV.
m0	A hyperparameter for inverse Wishart distribution for Sigma

Details

The shrinkage parameters, lambda and lambda_var, for the semiparametric Bayesian shrinkage estimator are selected by KCV. See help(lm_semi_Bayes_PCV) for details about semiparametric Bayesian estimator.

References

N. Lee, H. Choi, and S.-H. Kim (2016). Bayes shrinkage estimation for high-dimensional VAR models with scale mixture of normal distributions for noise. Computational Statistics & Data Analysis 101, 250-276. doi: 10.1016/j.csda.2016.03.007

---

logLik.varshrinkest *Log-likelihood method for class "varshrinkest"*

Description

Returns the log-likelihood of a VAR model estimated by VARshrink(). It extends vars::logLik.varest() to incorporate 1) multivariate t-distribution for residuals, 2) scale matrix Sigma provided by shrinkage methods, and 3) effective number of parameters provided by shrinkage methods.

Usage

```
## S3 method for class 'varshrinkest'
logLik(object, ...)
```

Arguments

object	An object of class "varshrinkest"
...	Currently not used.

Value

log-likelihood of the fitted VAR model, with the class attribute "logLik" and the attributes df and nobs.

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
estim <- VARshrink(y, p = 2, type = "const", method = "ridge")
logLik(estim)
```

Phi.varshrinkest *Coefficient matrices of the MA representation*

Description

Returns the estimated coefficient matrices of the moving average representation of a stable VAR(p).

Usage

```
## S3 method for class 'varshrinkest'
Phi(x, nstep = 10, ...)
```

Arguments

x	An object of class "varshrinkest", generated by VARshrink().
nstep	An integer specifying the number of moving error coefficient matrices to be calculated.
...	Currently not used.

Details

This is a modification of vars::Phi.varest() for the class "varshrinkest", preventing redundant copying of data matrix objects.

Value

An array with dimension $K \times K \times (nstep + 1)$ holding the estimated coefficients of the moving average representation. The first slice of the array is the starting value, i.e., $\mathbf{\Phi}_0$.

See Also

Phi

print.varshrinkest *Print method for class "varshrinkest"*

Description

print method for an object of class "varshrinkest"

Usage

```
## S3 method for class 'varshrinkest'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

x	An object of class "varshrinkest"
digits, ...	Other arguments for print() method

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
estim <- VARshrink(y, p = 2, type = "const", method = "ridge")
print(estim)
```

---

print.varshsum *Print method for class "varshsum"*

Description

print method for an object obtained by summary.varshrinkest().

Usage

```
## S3 method for class 'varshsum'
print(
  x,
  digits = max(3, getOption("digits") - 3),
  signif.stars = getOption("show.signif.stars"),
  ...
)
```

Arguments

x	An object of class "varshsum"
digits, signif.stars, ...	Other arguments for print(), printCoefmat(), format() method

Details

This function extends print.varsum() for VAR models estimated by shrinkage methods. The output includes the covariance matrix Sigma and the degrees-of-freedom dof for the multivariate t-distribution for the residuals.

Psi.varshrinkest *Coefficient matrices of the orthogonalized MA representation*

Description

Returns the estimated orthogonalized coefficient matrices of the moving average representation of a stable VAR(p).

Usage

```
## S3 method for class 'varshrinkest'
Psi(x, nstep = 10, ...)
```

Arguments

x	An object of class "varshrinkest", generated by VARshrink().
nstep	An integer specifying the number of orthogonalized moving error coefficient matrices to be calculated.
...	Currently not used.

Details

This is an extension of vars::Psi.varest() for the class "varshrinkest".

Value

An array with dimension $K \times K \times (nstep + 1)$ holding the estimated coefficients of the orthogonalized moving average representation.

See Also

Psi

serial.test_sh *Test for serially correlated errors*

Description

Computes the multivariate Portmanteau- and Breusch-Godfrey test for serially correlated errors.

Usage

```
serial.test_sh(
  x,
  lags.pt = 16,
  lags.bg = 5,
  type = c("PT.asymptotic", "PT.adjusted", "BG", "ES")
)
```

Arguments

x	An object of class "varshrinkest" obtained by VARshrink().
lags.pt, lags.bg, type	Other arguments for vars::serial.test().

Details

An extension of vars::serial.test() to the class "varshrinkest".

Value

An object of class "varcheck" computed by vars::serial.test().

See Also

serial.test

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
estim <- VARshrink(y, p = 2, type = "const", method = "ridge")
serial.test_sh(estim)
```

---

shrinkVARcoef *Semiparametric Bayesian Shrinkage Estimator for Multivariate Regression*

Description

Compute the semiparametric Bayesian shrinkage estimator of Psi and Sigma for a given shrinkage parameter lambda. The function is a private function for lm_semi_Bayes_PCV() and lm_ShVAR_KCV().

Usage

```
shrinkVARcoef(
  Y,
  X,
  lambda,
  dof = Inf,
  prior_type = "NCJ",
  TolDRes = 1e-04,
  m0 = ncol(Y)
)
```

Arguments

Y	An N x K matrix of dependent variables.
X	An N x M matrix of regressors.
lambda	A shrinkage intensity parameter value between 0~1.
dof	Degree of freedom for multivariate t-distribution. If NULL or Inf, then use multivariate normal distribution.
prior_type	"NCJ" for non-conjugate prior and "CJ" for conjugate prior for scale matrix Sigma.
TolDRes	Tolerance parameter for stopping criterion.
m0	A hyperparameter for inverse Wishart distribution for Sigma

References

N. Lee, H. Choi, and S.-H. Kim (2016). Bayes shrinkage estimation for high-dimensional VAR models with scale mixture of normal distributions for noise. Computational Statistics & Data Analysis 101, 250-276. doi: 10.1016/j.csda.2016.03.007

---

simVARmodel *Generate multivariate time series data using the given VAR model*

Description

Generate a multivariate time series data set using the given VAR model.

Usage

```
simVARmodel(numT, model, burnin = 0)
```

Arguments

numT	Number of observed time points, T.
model	A list object with Coef, Sigma, dof; Coef is a list with A and c; A is a list object of K-by-K coefficient matrices and c is a length-K vector. Sigma is a K-by-K scale matrix and dof is a degree of freedom for multivariate t-distribution for noise.
burnin	Number of initial points which are not included in the final values.

Details

First, it creates (p+burnin+numT x K) data, then it remove the first (p+burnin) vectors. Finally, it returns (numT x K) data.

Value

A numT-by-K matrix

Examples

```
myCoef <- list(A = list(matrix(c(0.5, 0, 0, 0.5), 2, 2)), c = c(0.2, 0.7))
myModel <- list(Coef = myCoef, Sigma = diag(0.1^2, 2), dof = Inf)
simVARmodel(numT = 100, model = myModel, burnin = 10)
```

`stability_sh` *Structural stability of a VAR(p)*

Description

Computes empirical fluctuation processes for VAR estimates. Utilizes `strucchange::efp()` for the VAR estimates of each time series variable.

Usage

```
stability_sh(
  x,
  type = c("OLS-CUSUM", "Rec-CUSUM", "Rec-MOSUM", "OLS-MOSUM", "RE", "ME", "Score-CUSUM",
    "Score-MOSUM", "fluctuation"),
  h = 0.15,
  dynamic = FALSE,
  rescale = TRUE,
  ...
)
```

Arguments

`x`	Object of class "varshrinkest"
`type`, `h`, `dynamic`, `rescale`, `...`	Other arguments to `strucchange::efp()`

Details

A variant of `vars::stability()` for an object of class "varshrinkest".

Value

A list with class attribute "varstabil" which contains the following elements: `stability`, `endog`, `K`. The `stability` is a list of `strucchange::efp()` outputs.

See Also

`stability`

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
estim <- VARshrink(y, p = 2, type = "const", method = "ridge")
stabil <- stability_sh(estim)
plot(stabil)
```

`summary.shrinklm` *Summarizing shrinkage estimates of an AR model*

Description

`summary` method for an object of class "shrinklm".

Usage

```
## S3 method for class 'shrinklm'
summary(object, correlation = FALSE, symbolic.cor = FALSE, ...)
```

Arguments

`object`	An object of class "shrinklm"
`correlation`	If TRUE, the correlation matrix of the the estimated coefficients is returned and printed.
`symbolic.cor`	If TRUE, print the correlations in a symbolic form rather than as numbers
`...`	Currently not used.

Details

The class "shrinklm" inherits from the class "lm", and extends it to incorporate shrinkage estimates with the effective number of parameters.

Value

Returns a list with class attribute "summary.lm", which contains summary statistics of the fitted linear model given in object.

summary.varshrinkest *Summary method for an object of class "varshrinkest", VAR parameters estimated by VARshrink()*

Description

Extend summary.varest() to the class "varshrinkest" to incorporate adapted methods for the new classes: summary.shrinklm(), logLik.varshrinkest().

Usage

```
## S3 method for class 'varshrinkest'
summary(object, equations = NULL, ...)
```

Arguments

object	An object of class "varshrinkest", usually a result of call to "VARshrink()".
equations	Subset of names of endogenous time series variables to summarize.
...	Currently not used.

Details

The code has been modified to eliminate direct calls to the data matrices ($y, $datamat) and to use the effective numbers of parameters obtained from the shrinkage estimates. The output additionally includes the covariance matrix Sigma and the degrees-of-freedom dof of the multivariate t-distribution for the residuals.

Value

Returns a list with class attribute "varshsum" and "varsum" which contains the following elements: names, logLik, obs, roots, type, call, varresult, covres, corres, Sigma, dof.

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
estim <- VARshrink(y, p = 2, type = "const", method = "ridge")
summary(estim)
```

VARshrink *Shrinkage estimation of VAR parameters*

Description

Shrinkage estimation methods for high-dimensional VAR models. Consider VAR(p) model:

$$y_t = A_1 y_{t-1} + \cdots + A_p y_{t-p} + C d_t + e_t,$$

where y_t is K-dimensional time series, d_t is deterministic regressors, e_t is a noise process, and $A_1, \ldots, A_p$, and C are coefficient matrices. Exogenous variables can be included additionally as regressors.

Usage

```
VARshrink(
  y,
  p = 1,
  type = c("const", "trend", "both", "none"),
  season = NULL,
  exogen = NULL,
  method = c("ridge", "ns", "fbayes", "sbayes", "kcv", "ols"),
  lambda = NULL,
  lambda_var = NULL,
  dof = Inf,
  ...
)
```

Arguments

y	A T-by-K matrix of endogenous variables
p	Integer for the lag order
type	Type of deterministic regressors to include. 1) const - the constant. 2) trend - the trend. 3) both - both the constant and the trend. 4) none - no deterministic regressors. ***Note: method="ns" does not accept type="const" and type="both" to avoid constant term.
season	An integer value of frequency for inclusion of centered seasonal dummy variables. abs(season) >= 3.
exogen	A T-by-L matrix of exogenous variables. Default is NULL.
method	1) "ridge" - multivariate ridge regression. 2) "ns" - a Stein-type nonparametric shrinkage method. 3) "fbayes" - a full Bayesian shrinkage method using noninformative priors. 4) "sbayes" - a semiparametric Bayesian shrinkage method using parameterized cross validation. 5) "kcv" - a semiparametric Bayesian shrinkage method using K-fold cross validation. 6) "ols" - ordinary least squares method.

`lambda`, `lambda_var`
: Shrinkage parameter value(s). Use of this parameter is slightly different for each method: the same value does not imply the same shrinkage estimates.

`dof`
: Degree of freedom of multivariate t-distribution for noise. Valid only for `method = "fbayes"` and `method = "sbayes"`. `dof = Inf` means multivariate normal distribution.

`...`
: Extra arguments to pass to a specific function of the estimation method. For example, burnincycle and mcmccycle are for `"fbayes"`.

Details

Shrinkage estimation methods can estimate the coefficients even when the dimensionality K is larger than the number of observations.

Value

An object of class "varshrinkest" with the components: varresult, datamat, y, type, p, K, obs, totobs, restrictions, method, lambda, call. The class "varshrinkest" inherits the class "varest" in the package vars.

Examples

```
data(Canada, package = "vars")
y <- diff(Canada)
VARshrink(y, p = 2, type = "const", method = "ridge")
```

Index