Package: fastSOM (via r-universe)

September 7, 2024

Type Package

Version 1.0.1

Date 2019-11-19

Title Fast Calculation of Spillover Measures

Imports parallel

Description Functions for computing spillover measures, especially
spillover tables and spillover indices, as well as their
average, minimal, and maximal values.

License GPL (>= 2)

NeedsCompilation yes

Author Stefan Kloessner [aut, cre], Sven Wagner [aut]

Maintainer Stefan Kloessner <S.Kloessner@mx.uni-saarland.de>

Repository CRAN

Date/Publication 2019-11-19 12:40:06 UTC

Contents

---

fastSOM-package   *Fast Calculation of Spillover Measures*

Description

This package comprises various functions for computing spillover measures, especially spillover tables and spillover indices as proposed by Diebold and Yilmaz (2009) as well as their estimated and exact average, minimal, and maximal values.

Details

Package:	fastSOM
Type:	Package
Version:	1.0.0
Date:	2016-07-20
License:	GPL (>=2)

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers, with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers? Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1): 172-179

---

soi   *Calculation of the Spillover Index*

Description

This function calculates the spillover index as proposed by Diebold and Yilmaz (2009, see References).

Usage

```
soi(Sigma, A, ncores = 1, ...)
```

Arguments

`Sigma`	Either a covariance matrix or a list thereof.
`A`	Either a 3-dimensional array with A[,,h] being MA coefficient matrices of the same dimension as `Sigma` or a list thereof.
`ncores`	Number of cores, only relevant if Sigma is a list of matrices. Missing ncores or `ncores=1` means no parallelization (just one core is used). `ncores=0` means automatic detection of the number of available cores. Any other integer determines the maximal number of cores to be used.
`...`	Further arguments, especially `perm` which is used to reorder variables. If `perm` is missing, then the original ordering of the model variables will be used. If `perm` is a permutation of `1:N`, then the spillover index for the model with variables reordered according to perm will be calculated.

Details

The spillover index was introduced by Diebold and Yilmaz in 2009 (see References). It is based on a variance decompostion of the forecast error variances of an N-dimensional MA(∞) process. The underlying idea is to decompose the forecast error of each variable into own variance shares and cross variance shares. The latter are interpreted as contributions of shocks of one variable to the error variance in forecasting another variable (see also [sot](#)). The spillover index then is a number between 0 and 100, describing the relative amount of forecast error variances that can be explained by shocks coming from other variables in the model.

The typical application of the 'list' version of soi is a rolling windows approach when `Sigma` and `A` are lists representing the corresponding quantities at different points in time (rolling windows).

Value

Returns a single numeric value or a list thereof.

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers, with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers? Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1): 172-179

See Also

[fastSOM-package](#), [sot](#)

Examples

```
# generate randomly positive definite matrix Sigma of dimension N
N <- 10
Sigma <- crossprod(matrix(rnorm(N*N),nrow=N))
# generate randomly coefficient matrices
H <- 10
A <- array(rnorm(N*N*H),dim=c(N,N,H))
# calculate the spillover index
soi(Sigma, A)
```

---

soi_avg_est   *Estimation of Average, Minimal, and Maximal Spillover Index*

Description

Calculates an estimate of the average, the minimum, and the maximum spillover index based on different permutations.

Usage

```
soi_avg_est(Sigma, A, ncores = 1, ...)
```

Arguments

Sigma	Either a covariance matrix or a list thereof.
A	Either a 3-dimensional array with A[„h] being MA coefficient matrices of the same dimension as Sigma or a list thereof.
ncores	Number of cores. Missing ncores or ncores=1 means no parallelization (just one core is used). ncores=0 means automatic detection of the number of available cores. Any other integer determines the maximal number of cores to be used.
...	Further arguments, especially perms which is used to reorder variables. If perms is missing, then 10.000 randomly created permutations of 1:N will be used as reorderings of the model variables. If perms is defined, it has to be either a matrix with each column being a permutation of 1:N, or, alternatively, an integer value defining the number of randomly created permutations.

Details

The spillover index introduced by Diebold and Yilmaz (2009) (see References) depends on the ordering of the model variables. While soi_avg_exact provides a fast algorithm for exact calculation of average, minimum, and maximum of the spillover index over all permutations, there might be reasons to prefer to estimate these quantities using a limited number of permutations (mainly to save time when N is large). This is exactly what soi_avg_est does.

The typical application of the 'list' version of soi_avg_est is a rolling windows approach when Sigma and A are lists representing the corresponding quantities at different points in time (rolling windows).

Value

The 'single' version returns a list containing the estimated average, minimal, and maximal spillover index as well as permutations that generated the minimal and maximal value. The 'list' version returns a list consisting of three vectors (the average, minimal, and maximal spillover index values) and two matrices (the columns of which are the permutations generating the minima and maxima).

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers, with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers? Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1): 172-179

See Also

fastSOM-package, soi_avg_exact

Examples

```
# generate randomly positive definite matrix Sigma of dimension N
N <- 10
Sigma <- crossprod(matrix(rnorm(N*N),nrow=N))
# generate randomly coefficient matrices
H <- 10
A <- array(rnorm(N*N*H),dim=c(N,N,H))
# calculate estimates of the average, minimal,
# and maximal spillover index and determine the corresponding ordering
# of the model variables
soi_avg_est(Sigma, A)
```

---

soi_avg_exact   *Exact Calculation of Average, Minimal, and Maximal Spillover Index*

Description

Calculates the Average, Minimal, and Maximal Spillover Index exactly.

Usage

```
soi_avg_exact(Sigma, A, ncores = 1)
```

Arguments

Sigma   Either a covariance matrix or a list thereof.

A   Either a 3-dimensional array with A[,,h] being MA coefficient matrices of the
               same dimension as Sigma or a list thereof.

ncores   Number of cores. Missing ncores or ncores=1 means no parallelization (just
               one core is used). ncores=0 means automatic detection of the number of avail-
               able cores. Any other integer determines the maximal number of cores to be
               used.

Details

The spillover index introduced by Diebold and Yilmaz (2009) (see References) depends on the
ordering of the model variables. While soi_avg_est provides an algorithm to estimate aver-
age, minimum, and maximum of the spillover index over all permutations, soi_avg_est calcu-
lates these quantities exactly. Notice, however, that for large dimensions N, this might be quite
time- as well as memory-consuming. If only the exact average of the spillover index is wanted,
soi_from_sot(sot_avg_exact(Sigma,A,ncores)$Average) should be used.

The typical application of the 'list' version of soi_avg_exact is a rolling windows approach when
Sigma and A are lists representing the corresponding quantities at different points in time (rolling
windows).

Value

The 'single' version returns a list containing the exact average, minimal, and maximal spillover
index as well as permutations that generated the minimal and maximal value. The 'list' version
returns a list consisting of three vectors (the average, minimal, and maximal spillover index values)
and two matrices (the columns of which are the permutations generating the minima and maxima).

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers,
with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers?
Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1):
172-179

See Also

fastSOM-package, soi_avg_est

Examples

```
# generate randomly positive definite matrix Sigma of dimension N
N <- 10
Sigma <- crossprod(matrix(rnorm(N*N),nrow=N))
# generate randomly coefficient matrices
H <- 10
A <- array(rnorm(N*N*H),dim=c(N,N,H))
# calculate the exact average, minimal,
# and maximal spillover index and determine the corresponding ordering
# of the model variables
soi_avg_exact(Sigma, A)
```

---

soi_from_sot	*Calculation of the Spillover Index for a given Spillover Table*

Description

Given a spillover table, this function calculates the corresponding spillover index.

Usage

```
soi_from_sot(input_table)
```

Arguments

input_table   Either a spillover table or a list thereof

Details

The spillover index was introduced by Diebold and Yilmaz in 2009 (see References). It is based on a variance decompostion of the forecast error variances of an N-dimensional MA(∞) process. The underlying idea is to decompose the forecast error of each variable into own variance shares and cross variance shares. The latter are interpreted as contributions of shocks of one variable to the error variance in forecasting another variable (see also [sot](#)). The spillover index then is a number between 0 and 100, describing the relative amount of forecast error variances that can be explained by shocks coming from other variables in the model.

The typical application of the 'list' version of soi_from_sot is a rolling windows approach when input_table is a list representing the corresponding spillover tables at different points in time (rolling windows).

Value

Numeric value or a list thereof.

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers, with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers? Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1): 172-179

See Also

fastSOM-package, soi, sot

Examples

```
# generate randomly positive definite matrix Sigma of dimension N
N <- 10
Sigma <- crossprod(matrix(rnorm(N*N),nrow=N))
# generate randomly coefficient matrices
H <- 10
A <- array(rnorm(N*N*H),dim=c(N,N,H))
# calculate spillover table
SOT <- sot(Sigma,A)
# calculate spillover index from spillover table
soi_from_sot(SOT)
```

---

sot   *Calculation of Spillover Tables*

Description

This function calculates an NxN-dimensional spillover table.

Usage

```
sot(Sigma, A, ncores = 1, ...)
```

Arguments

Sigma	Either a covariance matrix or a list thereof.
A	Either a 3-dimensional array with A[,,h] being MA coefficient matrices of the same dimension as Sigma or a list thereof.
ncores	Number of cores, only relevant if Sigma is a list of matrices. Missing ncores or ncores=1 means no parallelization (just one core is used). ncores=0 means automatic detection of the number of available cores. Any other integer determines the maximal number of cores to be used.
...	Further arguments, especially perm which is used to reorder variables. If perm is missing, then the original ordering of the model variables will be used. If perm is a permutation of 1:N, then the spillover index for the model with variables reordered according to perm will be calculated.

Details

The (i, j)-entry of a spillover table represents the relative contribution of shocks in variable j (the column variable) to the forecasting error variance of variable i (the row variable). Hence, off-diagonal values are interpreted as spillovers, while the own variance shares appear on the diagonal. An overall spillover measure is given by soi.

The typical application of the 'list' version of sot is a rolling windows approach when Sigma and A are lists representing the corresponding quantities at different points in time (rolling windows).

Value

Matrix, or a list thereof, of dimensions NxN with non-negative entries summing up to 100 for each row.

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers, with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers? Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1): 172-179

See Also

fastSOM-package, soi

Examples

```
# generate randomly positive definite matrix Sigma of dimension N
N <- 10
Sigma <- crossprod(matrix(rnorm(N*N),nrow=N))
# generate randomly coefficient matrices
H <- 10
A <- array(rnorm(N*N*H),dim=c(N,N,H))
# calculate spillover table
sot(Sigma,A)
```

---

sot_avg_est	*Estimation of the Average, Minimal, and Maximal Entries of a Spillover Table*

Description

Calculates estimates of the average, minimal, and maximal entries of a spillover.

Usage

```
sot_avg_est(Sigma, A, ncores = 1, ...)
```

Arguments

Sigma	Either a covariance matrix or a list thereof.
A	Either a 3-dimensional array with A[,,h] being MA coefficient matrices of the same dimension as Sigma or a list thereof.
ncores	Number of cores. Missing ncores or ncores=1 means no parallelization (just one core is used). ncores=0 means automatic detection of the number of available cores. Any other integer determines the maximal number of cores to be used.
...	Further arguments, especially perms which is used to reorder variables. If perms is missing, then 10.000 randomly created permutations of 1:N will be used as reorderings of the model variables. If perms is defined, it has to be either a matrix with each column being a permutation of 1:N, or, alternatively, an integer value defining the number of randomly created permutations.

Details

The spillover tables introduced by Diebold and Yilmaz (2009) (see References) depend on the ordering of the model variables. While [sot_avg_exact](#) provides a fast algorithm for exact calculation of average, minimum, and maximum of the spillover table over all permutations, there might be reasons to prefer to estimate these quantities using a limited number of permutations (mainly to save time when N is large). This is exactly what sot_avg_est does.

The typical application of the 'list' version of sot_avg_est is a rolling windows approach when Sigma and A are lists representing the corresponding quantities at different points in time (rolling windows).

Value

The 'single' version returns a list containing the exact average, minimal, and maximal values for the spillover table. The 'list' version returns a list with three elements (Average, Minimum, Maximum) which themselves are lists of the corresponding tables.

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers, with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers? Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1): 172-179

See Also

fastSOM-package, sot_avg_exact

Examples

```
# generate randomly positive definite matrix Sigma of dimension N
N <- 10
Sigma <- crossprod(matrix(rnorm(N*N),nrow=N))
# generate randomly coefficient matrices
H <- 10
A <- array(rnorm(N*N*H),dim=c(N,N,H))
# calculate estimates of the average, minimal,
# and maximal entries within a spillover table
sot_avg_est(Sigma, A)
```

---

sot_avg_exact	*Calculation of the Exact Values for Average, Minimal, and Maximal Entries of a Spillover Table*

Description

Calculates the exact values of the average, the minimum, and the maximum entries of a spillover tables based on different permutations.

Usage

```
sot_avg_exact(Sigma, A, ncores = 1)
```

Arguments

Sigma	Either a covariance matrix or a list thereof.
A	Either a 3-dimensional array with A[„h] being MA coefficient matrices of the same dimension as Sigma or a list thereof.

ncores   Number of cores, only relevant for 'list' version. In this case, missing ncores or
                ncores=1 means no parallelization (just one core is used), ncores=0 means au-
                tomatic detection of the number of available cores, any other integer determines
                the maximal number of cores to be used.

Details

The spillover tables introduced by Diebold and Yilmaz (2009) (see References) depend on the
ordering of the model variables. While [sot_avg_est](#) provides an algorithm to estimate average,
minimal, and maximal values of the spillover table over all permutations, sot_avg_est calculates
these quantities exactly. Notice, however, that for large dimensions N, this might be quite time- as
well as memory-consuming.

The typical application of the 'list' version of sot_avg_exact is a rolling windows approach when
Sigma and A are lists representing the corresponding quantities at different points in time (rolling
windows).

Value

The 'single' version returns a list containing the exact average, minimal, and maximal values for the
spillover table. The 'list' version returns a list with three elements (Average, Minimum, Maximum)
which themselves are lists of the corresponding tables.

Author(s)

Stefan Kloessner (<S.Kloessner@mx.uni-saarland.de>),
with contributions by Sven Wagner (<sven.wagner@mx.uni-saarland.de>)

References

[1] Diebold, F. X. and Yilmaz, K. (2009): Measuring financial asset return and volatitliy spillovers,
with application to global equity markets, Economic Journal 199(534): 158-171.

[2] Kloessner, S. and Wagner, S. (2012): Exploring All VAR Orderings for Calculating Spillovers?
Yes, We Can! - A Note on Diebold and Yilmaz (2009), Journal of Applied Econometrics 29(1):
172-179

See Also

[fastSOM-package](#), [sot_avg_est](#)

Examples

```
# generate randomly positive definite matrix Sigma of dimension N
N <- 10
Sigma <- crossprod(matrix(rnorm(N*N),nrow=N))
# generate randomly coefficient matrices
H <- 10
A <- array(rnorm(N*N*H),dim=c(N,N,H))
# calculate the exact average, minimal,
# and maximal entries within a spillover table
sot_avg_exact(Sigma, A)
```

Index