Package ‘generalCorr’

October 30, 2019

Type Package

Title Generalized Correlations and Plausible Causal Paths

Version 1.1.5

Date 2019-10-30

Author Prof. H. D. Vinod, Fordham University, NY.

Maintainer H. D. Vinod <vinod@fordham.edu>

Encoding UTF-8

Depends R (>= 3.0.0), np (>= 0.60), xtable (>= 1.8), meboot (>= 1.4), psych, lattice

Suggests R.rsp

VignetteBuilder R.rsp

Description Since causal paths from data are important for all sciences, the package provides many sophisticated functions. causeSummBlk() gives easy-to-interpret causal paths. Let Z denote control variables and compare two flipped kernel regressions: X=f(Y, Z)+e1 and Y=g(X,Z)+e2. Our criterion Cr1 says that if |e1*Y|>|e2*X| then variation in X is more ``exogenous or independent'' than in Y and causal path is X to Y. Criterion Cr2 requires |e2|<|e1|. These inequalities between many absolute value are quantified by four orders of stochastic dominance. Our third criterion Cr3 for the causal path X to Y requires new generalized partial correlations to satisfy |r*(x|y,z)|< |r*(y|x,z)|. The function parcorBMany() reports generalized partials between the first variable and all others. The package provides additional R tools for causal assessment, ``outlier detection,'' and for numerical integration by the trapezoidal rule, stochastic dominance, pillar 3D charts, etc. We also provide functions for bootstrap-based statistical inference for causal paths. causeSummary() and causeSummBlk() are easiest to use functions.

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-10-30 20:50:02 UTC

R topics documented:

`absBstdres` *Block version of abs-stdres Absolute values of residuals of kernel regressions of standardized x on standardized y, no control variables.*

Description

1) Standardize the data to force mean zero and variance unity, 2) kernel regress x on y, with the option 'residuals = TRUE' and finally 3) compute the absolute values of residuals.

Usage

```
absBstdres(x, y, blksiz = 10)
```

Arguments

`x`	vector of data on the dependent variable
`y`	data on the regressors which can be a matrix
`blksiz`	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done

Details

The first argument is assumed to be the dependent variable. If `abs_stdres(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression residuals are returned after standardizing the data on both sides so that the magnitudes of residuals are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
abs_stdres(x,y)

## End(Not run)
```

---

absBstdresC — *Block version of Absolute values of residuals of kernel regressions of standardized x on standardized y and control variables.*

Description

1) standardize the data to force mean zero and variance unity, 2) kernel regress x on y and a matrix of control variables, with the option 'residuals = TRUE' and finally 3) compute the absolute values of residuals.

Usage

```
absBstdresC(x, y, ctrl, blksiz = 10)
```

Arguments

x	vector of data on the dependent variable
y	data on the regressors which can be a matrix
ctrl	Data matrix on the control variable(s) beyond causal path issues
blksiz	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done

Details

The first argument is assumed to be the dependent variable. If `abs_stdres(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with two or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression residuals are returned after standardizing the data on both sides so that the magnitudes of residuals are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See `abs_stdres`.

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
z=sample(21:51)
absBstdresC(x,y,ctrl=z)

## End(Not run)
```

---

absBstdrhserC

*Block version abs_stdrhser Absolute residuals kernel regressions of standardized x on y and control variables, Cr1 has abs(Resid*RHS).*

Description

1) standardize the data to force mean zero and variance unity, 2) kernel regress x on y and a matrix of control variables, with the option 'residuals = TRUE' and finally 3) compute the absolute values of residuals.

Usage

```
absBstdrhserC(x, y, ctrl, ycolumn = 1, blksiz = 10)
```

Arguments

`x`	vector of data on the dependent variable
`y`	data on the regressors which can be a matrix
`ctrl`	Data matrix on the control variable(s) beyond causal path issues
`ycolumn`	if y has more than one column, the column number used when multiplying residuals times this column of y, default=1 or first column of y matrix is used
`blksiz`	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done

Details

The first argument is assumed to be the dependent variable. If `absBstdrhserC(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression residuals are returned after standardizing the data on both sides so that the magnitudes of residuals are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, `http://dx.doi.org/10.1080/03610918.2015.1122048`

See Also

See `abs_stdres`.

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
z=sample(21:51)
absBstdrhserC(x,y,ctrl=z)
```

```
## End(Not run)
```

abs_res *Absolute residuals of kernel regression of x on y.*

Description

This internal function calls the kern function to implement kernel regression with the option residuals=TRUE and returns absolute residuals.

Usage

```
abs_res(x, y)
```

Arguments

x	vector of data on the dependent variable
y	vector of data on the regressor

Details

The first argument is assumed to be the dependent variable. If abs_res(x,y) is used, you are regressing x on y (not the usual y on x)

Value

absolute values of kernel regression residuals are returned.

Note

This function is intended for internal use.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
abs_res(x,y)

## End(Not run)
```

abs_stdapd — *Absolute values of gradients (apd's) of kernel regressions of x on y when both x and y are standardized.*

Description

1) standardize the data to force mean zero and variance unity, 2) kernel regress x on y, with the option 'gradients = TRUE' and finally 3) compute the absolute values of gradients

Usage

```
abs_stdapd(x, y)
```

Arguments

x	vector of data on the dependent variable
y	data on the regressors which can be a matrix

Details

The first argument is assumed to be the dependent variable. If `abs_stdapd(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression gradients are returned after standardizing the data on both sides so that the magnitudes of amorphous partial derivatives (apd's) are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
abs_stdapd(x,y)

## End(Not run)
```

abs_stdapdC *Absolute values of gradients (apd's) of kernel regressions of x on y when both x and y are standardized and control variables are present.*

Description

1) standardize the data to force mean zero and variance unity, 2) kernel regress x on y and a matrix of control variables, with the option 'gradients = TRUE' and finally 3) compute the absolute values of gradients

Usage

```
abs_stdapdC(x, y, ctrl)
```

Arguments

x	vector of data on the dependent variable
y	data on the regressors which can be a matrix
ctrl	Data matrix on the control variable(s) beyond causal path issues

Details

The first argument is assumed to be the dependent variable. If `abs_stdapdC(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression gradients are returned after standardizing the data on both sides so that the magnitudes of amorphous partial derivatives (apd's) are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

See Also

See `abs_stdapd`.

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
z=sample(20:50)
abs_stdapdC(x,y,ctrl=z)
```

```
## End(Not run)
```

---

abs_stdres — *Absolute values of residuals of kernel regressions of x on y when both x and y are standardized.*

Description

1) Standardize the data to force mean zero and variance unity, 2) kernel regress x on y, with the option 'residuals = TRUE' and finally 3) compute the absolute values of residuals.

Usage

```
abs_stdres(x, y)
```

Arguments

x	vector of data on the dependent variable
y	data on the regressors which can be a matrix

Details

The first argument is assumed to be the dependent variable. If `abs_stdres(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression residuals are returned after standardizing the data on both sides so that the magnitudes of residuals are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
abs_stdres(x,y)

## End(Not run)
```

`abs_stdresC` *Absolute values of residuals of kernel regressions of x on y when both x and y are standardized and control variables are present.*

Description

1) standardize the data to force mean zero and variance unity, 2) kernel regress x on y and a matrix of control variables, with the option 'residuals = TRUE' and finally 3) compute the absolute values of residuals.

Usage

```
abs_stdresC(x, y, ctrl)
```

Arguments

`x`	vector of data on the dependent variable
`y`	data on the regressors which can be a matrix
`ctrl`	Data matrix on the control variable(s) beyond causal path issues

Details

The first argument is assumed to be the dependent variable. If `abs_stdres(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with two or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression residuals are returned after standardizing the data on both sides so that the magnitudes of residuals are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See `abs_stdres`.

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
z=sample(21:51)
abs_stdresC(x,y,ctrl=z)

## End(Not run)
```

---

abs_stdrhserC — *Absolute residuals kernel regressions of standardized x on y and control variables, Cr1 has abs(RHS*y) not gradients.*

Description

1) standardize the data to force mean zero and variance unity, 2) kernel regress x on y and a matrix of control variables, with the option 'residuals = TRUE' and finally 3) compute the absolute values of residuals.

Usage

```
abs_stdrhserC(x, y, ctrl, ycolumn = 1)
```

Arguments

`x`	vector of data on the dependent variable
`y`	data on the regressors which can be a matrix
`ctrl`	Data matrix on the control variable(s) beyond causal path issues
`ycolumn`	if y has more than one column, the column number used when multiplying residuals times this column of y, default=1 or first column of y matrix is used

Details

The first argument is assumed to be the dependent variable. If `abs_stdrhserC(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression residuals are returned after standardizing the data on both sides so that the magnitudes of residuals are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See abs_stdres.

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
z=sample(21:51)
abs_stdrhserC(x,y,ctrl=z)

## End(Not run)
```

---

abs_stdrhserr *Absolute values of Hausman-Wu null in kernel regressions of x on y when both x and y are standardized.*

Description

1) standardize the data to force mean zero and variance unity, 2) kernel regress x on y, with the option 'gradients = TRUE' and finally 3) compute the absolute values of Hausman-Wu null hypothesis for testing exogeneity, or E(RHS.regressor*error)=0 where error is approximated by kernel regression residuals

Usage

```
abs_stdrhserr(x, y)
```

Arguments

x	vector of data on the dependent variable
y	data on the regressors which can be a matrix

Details

The first argument is assumed to be the dependent variable. If `abs_stdrhserr(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

Absolute values of kernel regression RHS*residuals are returned after standardizing the data on both sides so that the magnitudes of Hausman-Wu null values are comparable between regression of x on y on the one hand and flipped regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
abs_stdrhserr(x,y)

## End(Not run)
```

---

allPairs

Report causal identification for all pairs of variables in a matrix (deprecated function). It is better to choose a target variable and pair it with all others, instead of considering all possible targets.

Description

This studies all possible (perhaps too many) causal directions in a matrix. It is deprecated because it uses older criterion 1 by caling `abs_stdapd` I recommend using `causeSummary` or its block version `cuseSummBlk`. This uses `abs_stdres`, `comp_portfo2`, etc. and returns a matrix with 7 columns having detailed output. Criterion 1 has been revised as described in Vinod (2019) and is known to work better.

Usage

```
allPairs(mtx, dig = 6, verbo = FALSE, typ = 1, rnam = FALSE)
```

Arguments

`mtx`	Input matrix with variable names
`dig`	Digits of accuracy in reporting (=6, default)
`verbo`	Logical variable, set to 'TRUE' if printing is desired
`typ`	Causal direction criterion number (typ=1 is default) Criterion 1 (Cr1) compares kernel regression absolute values of gradients. Criterion 2 (Cr2) compares kernel regression absolute values of residuals. Criterion 3 (Cr3) compares kernel regression based r*(x|y) with r*(y|x).
`rnam`	Logical variable, default `rnam=FALSE` means the user does not want the row names to be (somewhat too cleverly) assigned by the function.

Value

A 7-column matrix called 'outcause' with names of variables X and Y in the first two columns and the name of the 'causal' variable in 3rd col. Remaining four columns report numerical computations of SD1 to SD4, r*(x|y), r*(y|x). Pearson r and p-values for its traditional significance testing.

Note

The cause reported in the third column is identified from the sign of the first SD1 only, ignoring SD2, SD3 and SD4 under both Cr1 and Cr2. It is a good idea to loop a call to this function with typ=1:3. One can print the resulting 'outcause' matrix with the `xtable(outcause)` for the Latex output. A similar deprecated function included in this package, called `some0Pairs`, incorporates all SD1 to SD4 and all three criteria Cr1 rto Cr3 to report a 'sum' of indexes representing the signed number whose sign can more comprehensively help determine the causal direction(s). Since the Cr1 here is revised in later work, this is deprecated.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'New exogeneity tests and causal paths,' Chapter 2 in 'Handbook of Statistics: Conceptual Econometrics Using R', Vol.32, co-editors: H. D. Vinod and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2019, pp. 33-64.

See Also

See Also `somePairs`, `some0Pairs` `causeSummary`

Examples

```
data(mtcars)
options(np.messages=FALSE)
for(j in 1:3){
a1=allPairs(mtcars[,1:3], typ=j)
print(a1)}
```

---

badCol *internal badCol*

Description

intended for internal use

Usage

```
data(badCol)
```

Format

The format is: int 4

---

bigfp *Compute the numerical integration by the trapezoidal rule.*

Description

See page 220 of Vinod (2008) "Hands-on Intermediate Econometrics Using R," for the trapezoidal integration formula needed for stochastic dominance. The book explains pre-multiplication by two large sparse matrices denoted by I_F, I_f. Here we accomplish the same computation without actually creating the large sparse matrices. For example, the I_f is replaced by `cumsum` in this code (unlike the R code in my textbook).

Usage

```
bigfp(d, p)
```

Arguments

d	A vector of consecutive interval lengths, upon combining both data vectors
p	Vector of probabilities of the type 1/2T, 2/2T, 3/2T, etc. to 1.

Value

Returns a result after pre-multiplication by I_F, I_f matrices, without actually creating the large sparse matrices. This is an internal function.

Note

This is an internal function, called by the function `stochdom2`, for comparison of two portfolios in terms of stochastic dominance (SD) of orders 1 to 4. Typical usage is: `sd1b=bigfp(d=dj,p=rhs)` `sd2b=bigfp(d=dj,p=sd1b)` `sd3b=bigfp(d=dj,p=sd2b)` `sd4b=bigfp(d=dj,p=sd3b)`. This produces numerical evaluation vectors for the four orders, SD1 to SD4.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.', 'Hands-On Intermediate Econometrics Using R' (2008) World Scientific Publishers: Hackensack, NJ. https://www.worldscientific.com/worldscibooks/10.1142/6895

`bootPairs` *Compute matrix of n999 rows and p-1 columns of bootstrap 'sum' (strength from Cr1 to Cr3).*

Description

Maximum entropy bootstrap (meboot) package is used for statistical inference using the sum of three signs sg1 to sg3 from the three criteria Cr1 to Cr3 to assess preponderance of evidence in favor of a sign. (+1, 0, -1). The bootstrap output can be analyzed to assess approximate preponderance of a particular sign which determines the causal direction.

Usage

```
bootPairs(mtx, ctrl = 0, n999 = 9)
```

Arguments

`mtx`	data matrix with two or more columns
`ctrl`	data matrix having control variable(s) if any
`n999`	Number of bootstrap replications (default=9)

Value

out When `mtx` has p columns, `out` of `bootPairs(mtx)` is a matrix of n999 rows and p-1 columns each containing resampled 'sum' values summarizing the weighted sums associated with all three criteria from the function `silentPairs(mtx)` applied to each bootstrap sample separately.

Note

This computation is computer intensive and generally very slow. It may be better to use it at a later stage in the investigation when a preliminary causal determination is already made. A positive sign for j-th weighted sum reported in the column 'sum' means that the first variable listed in the argument matrix `mtx` is the 'kernel cause' of the variable in the (j+1)-th column of `mtx`.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Zheng, S., Shi, N.-Z., and Zhang, Z. (2012). Generalized measures of correlation for asymmetry, nonlinearity, and beyond. Journal of the American Statistical Association, vol. 107, pp. 1239-1252.

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See Also `silentPairs`.

Examples

```
## Not run:
options(np.messages = FALSE)
set.seed(34);x=sample(1:10);y=sample(2:11)
bb=bootPairs(cbind(x,y),n999=29)
apply(bb,2,summary) #gives summary stats for n999 bootstrap sum computations

bb=bootPairs(airquality,n999=999);options(np.messages=FALSE)
apply(bb,2,summary) #gives summary stats for n999 bootstrap sum computations

data('EuroCrime')
attach(EuroCrime)
bootPairs(cbind(crim,off),n999=29)#First col. crim causes officer deployment,
#hence positives signs are most sensible for such call to bootPairs
#note that n999=29 is too small for real problems, chosen for quickness here.

## End(Not run)
```

`bootPairs0` *Compute matrix of n999 rows and p-1 columns of bootstrap 'sum' index (strength from older criterion Cr1, with newer Cr2 and Cr3).*

Description

Maximum entropy bootstrap (meboot) package is used for statistical inference using the sum of three signs sg1 to sg3 from the three criteria Cr1 to Cr3 to assess preponderance of evidence in favor of a sign. (+1, 0, -1). The bootstrap output can be analyzed to assess approximate preponderance of a particular sign which determines the causal direction.

Usage

```
bootPairs0(mtx, ctrl = 0, n999 = 9)
```

Arguments

`mtx`	data matrix with two or more columns
`ctrl`	data matrix having control variable(s) if any
`n999`	Number of bootstrap replications (default=9)

Value

out When `mtx` has p columns, `out` of `bootPairs(mtx)` is a matrix of n999 rows and p-1 columns each containing resampled 'sum' values summarizing the weighted sums associated with all three criteria from the function `silentPairs(mtx)` applied to each bootstrap sample separately.

Note

This computation is computer intensive and generally very slow. It may be better to use it at a later stage in the investigation when a preliminary causal determination is already made. A positive sign for j-th weighted sum reported in the column 'sum' means that the first variable listed in the argument matrix `mtx` is the 'kernel cause' of the variable in the (j+1)-th column of `mtx`.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Zheng, S., Shi, N.-Z., and Zhang, Z. (2012). Generalized measures of correlation for asymmetry, nonlinearity, and beyond. Journal of the American Statistical Association, vol. 107, pp. 1239-1252.

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See Also silentPairs0, bootPairs has the version with later version of Cr1.

Examples

```
## Not run:
options(np.messages = FALSE)
set.seed(34);x=sample(1:10);y=sample(2:11)
bb=bootPairs0(cbind(x,y),n999=29)
apply(bb,2,summary) #gives summary stats for n999 bootstrap sum computations

bb=bootPairs0(airquality,n999=999);options(np.messages=FALSE)
apply(bb,2,summary) #gives summary stats for n999 bootstrap sum computations

data('EuroCrime')
attach(EuroCrime)
bootPairs0(cbind(crim,off),n999=29)#First col. crim causes officer deployment,
#hence positives signs are most sensible for such call to bootPairs
#note that n999=29 is too small for real problems, chosen for quickness here.

## End(Not run)
```

---

bootQuantile — *Compute confidence intervals [quantile(s)] of indexes from bootPairs output*

Description

Begin with the output of bootPairs function, a (n999 by p-1) matrix when there are p columns of data, `bootQuantile` produces a (k by p-1) mtx of quantile(s) of bootstrap ouput assuming that there are k quantiles needed.

Usage

```
bootQuantile(out, probs = c(0.025, 0.975), per100 = TRUE)
```

Arguments

`out`	output from bootPairs with p-1 columns and n999 rows
`probs`	quantile evaluation probabilities. The default is k=2, probs=c(.025,0.975) for a 95 percent confidence interval. Note that there are k=2 quantiles desired for each column with this specification
`per100`	logical (default per100=TRUE) to change the range of 'sum' to [-100, 100] values which are easier to interpret

Value

CI k quantiles evaluated at probs as a matrix with k rows and quantile of pairwise p-1 indexes representing p-1 column pairs (fixing the first column in each pair) This function summarizes the output of of `bootPairs(mtx)` (a n999 by p-1 matrix) each containing resampled 'sum' values summarizing the weighted sums associated with all three criteria from the function `silentPairs(mtx)` applied to each bootstrap sample separately. #'

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See Also `silentPairs`.

Examples

```
## Not run:
options(np.messages = FALSE)
set.seed(34);x=sample(1:10);y=sample(2:11)
bb=bootPairs(cbind(x,y),n999=29)
bootQuantile(bb) #gives summary stats for n999 bootstrap sum computations

bb=bootPairs(airquality,n999=999);options(np.messages=FALSE)
bootQuantile(bb,tau=0.476)#signs for n999 bootstrap sum computations

data('EuroCrime')
attach(EuroCrime)
bb=bootPairs(cbind(crim,off),n999=29) #col.1= crim causes off
#hence positive signs are more intuitively meaningful.
#note that n999=29 is too small for real problems, chosen for quickness here.
bootQuantile(bb)# quantile matrix for n999 bootstrap sum computations

## End(Not run)
```

bootSign *Probability of unambiguously correct (+ or -) sign from bootPairs output*

Description

If there are p columns of data, `bootSign` produces a p-1 by 1 vector of probabilities of correct signs assuming that the mean of n999 values has the correct sign and assuming that m of the 'sum' index values inside the range [-tau, tau] are neither positive nor negative but indeterminate or ambiguous (being too close to zero). That is, the denominator of P(+1) or P(-1) is (n999-m) if m signs are too close to zero. Thus it measures the bootstrap success rate in identifying the correct sign, when the sign of the average of n999 bootstraps is assumed to be correct.

Usage

```
bootSign(out, tau = 0.476)
```

Arguments

`out`	output from bootPairs with p-1 columns and n999 rows
`tau`	threshold to determine what value is too close to zero, default tau=0.476 is equivalent to 15 percent threshold for the unanimity index ui

Value

sgn When `mtx` has p columns, `sgn` reports pairwise p-1 signs representing (fixing the first column in each pair) the average sign after averaging the output of of `bootPairs(mtx)` (a n999 by p-1 matrix) each containing resampled 'sum' values summarizing the weighted sums associated with all three criteria from the function `silentPairs(mtx)` applied to each bootstrap sample separately. #'

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See Also `silentPairs`, `bootQuantile`, `bootSignPcent`.

Examples

```
## Not run:
options(np.messages = FALSE)
set.seed(34);x=sample(1:10);y=sample(2:11)
bb=bootPairs(cbind(x,y),n999=29)
bootSign(bb,tau=0.476) #gives success rate in n999 bootstrap sum computations

bb=bootPairs(airquality,n999=999);options(np.messages=FALSE)
bootSign(bb,tau=0.476)#signs for n999 bootstrap sum computations

data('EuroCrime');options(np.messages=FALSE)
attach(EuroCrime)
bb=bootPairs(cbind(crim,off),n999=29) #col.1= crim causes off
#hence positive signs are more intuitively meaningful.
#note that n999=29 is too small for real problems, chosen for quickness here.
bootSign(bb,tau=0.476)#gives success rate in n999 bootstrap sum computations

## End(Not run)
```

`bootSignPcent` *Probability of unambiguously correct (+ or -) sign from bootPairs output transformed to percentages.*

Description

If there are p columns of data, `bootSignPcent` produces a p-1 by 1 vector of probabilities of correct signs assuming that the mean of n999 values has the correct sign and assuming that m of the 'ui' index values inside the range [-tau, tau] are neither positive nor negative but indeterminate or ambiguous (being too close to zero). That is, the denominator of P(+1) or P(-1) is (n999-m) if m signs are too close to zero. Thus it measures the bootstrap success rate in identifying the correct sign, when the sign of the average of n999 bootstraps is assumed to be correct.

Usage

```
bootSignPcent(out, tau = 5)
```

Arguments

`out`	output from bootPairs with p-1 columns and n999 rows
`tau`	threshold to determine what value is too close to zero, default tau=5 is 5 percent threshold for the unanimity index ui

Value

sgn When `mtx` has p columns, `sgn` reports pairwise p-1 signs representing (fixing the first column in each pair) the average sign after averaging the output of of `bootPairs(mtx)` (a n999 by p-1 matrix) each containing resampled 'sum' values summarizing the weighted sums associated with all three criteria from the function `silentPairs(mtx)` applied to each bootstrap sample separately. #'

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See Also `silentPairs`, `bootQuantile`, `bootSign`.

Examples

```
## Not run:
options(np.messages = FALSE)
set.seed(34);x=sample(1:10);y=sample(2:11)
bb=bootPairs(cbind(x,y),n999=29)
bootSignPcent(bb,tau=5) #gives success rate in n999 bootstrap sum computations

bb=bootPairs(airquality,n999=999);options(np.messages=FALSE)
bootSignPcent(bb,tau=5)#success rate for signs from n999 bootstraps

data('EuroCrime');options(np.messages=FALSE)
attach(EuroCrime)
bb=bootPairs(cbind(crim,off),n999=29) #col.1= crim causes off
#hence positive signs are more intuitively meaningful.
#note that n999=29 is too small for real problems, chosen for quickness here.
bootSignPcent(bb,tau=5)#successful signs from n999 bootstraps

## End(Not run)
```

`bootSummary` *Compute usual summary stats of 'sum' indexes from bootPairs output*

Description

Begin with the output of bootPairs function, a (n999 by p-1) matrix when there are p columns of data, `bootSummary` produces a (6 by p-1) mtx of summary of bootstrap ouput (Min, 1st Qu,Median, Mean, 3rd Qi.,Max)

Usage

```
bootSummary(out, per100 = TRUE)
```

Arguments

`out`	output from bootPairs with p-1 columns and n999 rows in input here
`per100`	logical (default per100=TRUE) to change the range of 'sum' to [-100, 100] values which are easier to interpret

Value

summ summary output from the (n999 by p-1) matrix output of `bootPairs(mtx)` each containing resampled 'sum' values summarizing the weighted sums associated with all three criteria from the function `silentPairs(mtx)` applied to each bootstrap sample separately.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, `http://dx.doi.org/10.1080/03610918.2015.1122048`

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: `https://ssrn.com/abstract=2982128`

See Also

See Also `silentPairs`.

Examples

```
## Not run:
options(np.messages = FALSE)
set.seed(34);x=sample(1:10);y=sample(2:11)
bb=bootPairs(cbind(x,y),n999=29)
bootSummary(bb) #gives summary stats for n999 bootstrap sum computations

bb=bootPairs(airquality,n999=999);options(np.messages=FALSE)
bootSummary(bb)#signs for n999 bootstrap sum computations

data('EuroCrime')
attach(EuroCrime)
bb=bootPairs(cbind(crim,off),n999=29) #col.1= crim causes off
#hence positive signs are more intuitively meaningful.
#note that n999=29 is too small for real problems, chosen for quickness here.
bootSummary(bb)#signs for n999 bootstrap sum computations

## End(Not run)
```

causeSummary *Kernel causality summary of evidence for causal paths from three criteria*

Description

Allowing input matrix of control variables, this function produces a 5 column matrix summarizing the results where the estimated signs of stochastic dominance order values, (+1, 0, -1), are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, -1). The final range for the unanimity of sign index is [–100, 100].

Usage

```
causeSummary(mtx, nam = colnames(mtx), ctrl = 0, dig = 6,
  wt = c(1.2, 1.1, 1.05, 1), sumwt = 4)
```

Arguments

`mtx`	The data matrix with many columns, y the first column is fixed and then paired with all columns, one by one, and still called x for the purpose of flipping.
`nam`	vector of column names for `mtx`. Default: colnames(mtx)
`ctrl`	data matrix for designated control variable(s) outside causal paths
`dig`	Number of digits for reporting (default `dig`=6).
`wt`	Allows user to choose a vector of four alternative weights for SD1 to SD4.
`sumwt`	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The reason for slightly declining sampling unreliability of higher moments is simply that SD4 involves fourth power of the deviations from the mean and SD3 involves 3rd power, etc. The summary results for all three criteria are reported in one matrix called `out`:

Value

If there are p columns in the input matrix, x1, x2, .., xp, say, and if we keep x1 as a common member of all causal direction pairs (x1, x(1+j)) for (j=1, 2, .., p-1) which can be flipped. That is, either x1 is the cause or x(1+j) is the cause in a chosen pair. The control variables are not flipped. The printed output of this function reports the results for p-1 pairs indicating which variable (by name) causes which other variable (also by name). It also prints strength or signed summary strength index in range [-100,100]. A positive sign of the strength index means x1 kernel causes x(1+j), whereas negative strength index means x(1+j) kernel causes x1. The function also prints Pearson correlation and its p-value. This function also returns a matrix of p-1 rows and 5 columns entitled: “cause", "response", "strength", "corr." and "p-value", respectively with self-explanatory titles. The first two columns have names of variables x1 or x(1+j), depending on which is the cause. The 'strength' column has absolute value of summary index in range [0,100] providing summary of causal results based on preponderance of evidence from Cr1 to Cr3 from four orders of stochastic dominance, etc. The order of input columns matters. The fourth column 'corr.' reports the Pearson correlation coefficient while the fifth column has the p-value for testing the null of zero Pearson coeff. This function calls `silentPairs` allowing for control variables. The output of this function can be sent to 'xtable' for a nice Latex table.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. Since Cr1 to Cr3 near unanimously suggest 'crim' as the cause of 'off', strength index 100 suggests unanimity. `attach(EuroCrime); causeSummary(cbind(crim,off))`

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'New exogeneity tests and causal paths,' Chapter 2 in 'Handbook of Statistics: Conceptual Econometrics Using R', Vol.32, co-editors: H. D. Vinod and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2019, pp. 33-64.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See `bootPairs`, `causeSummary0` has an older version of this function.

See `someCPairs`

`silentPairs`

Examples

```
## Not run:
mtx=as.matrix(mtcars[,1:3])
ctrl=as.matrix(mtcars[,4:5])
 causeSummary(mtx,ctrl,nam=colnames(mtx))

## End(Not run)

options(np.messages=FALSE)
```

```
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
causeSummary(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

`causeSummary0` *Older Kernel causality summary of evidence for causal paths from three criteria*

Description

Allowing input matrix of control variables, this function produces a 5 column matrix summarizing the results where the estimated signs of stochastic dominance order values, (+1, 0, -1), are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, -1). The final range for the unanimity of sign index is [–100, 100].

Usage

```
causeSummary0(mtx, nam = colnames(mtx), ctrl = 0, dig = 6,
  wt = c(1.2, 1.1, 1.05, 1), sumwt = 4)
```

Arguments

`mtx`	The data matrix with many columns, y the first column is fixed and then paired with all columns, one by one, and still called x for the purpose of flipping.
`nam`	vector of column names for `mtx`. Default: colnames(mtx)
`ctrl`	data matrix for designated control variable(s) outside causal paths
`dig`	Number of digits for reporting (default `dig`=6).
`wt`	Allows user to choose a vector of four alternative weights for SD1 to SD4.
`sumwt`	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The reason for slightly declining sampling unreliability of higher moments is simply that SD4 involves fourth power of the deviations from the mean and SD3 involves 3rd power, etc. The summary results for all three criteria are reported in one matrix called `out`:

Value

If there are p columns in the input matrix, x1, x2, .., xp, say, and if we keep x1 as a common member of all causal direction pairs (x1, x(1+j)) for (j=1, 2, .., p-1) which can be flipped. That is, either x1 is the cause or x(1+j) is the cause in a chosen pair. The control variables are not flipped. The printed output of this function reports the results for p-1 pairs indicating which variable (by name) causes which other variable (also by name). It also prints strength or signed summary strength index in range [-100,100]. A positive sign of the strength index means x1 kernel causes x(1+j), whereas negative strength index means x(1+j) kernel causes x1. The function also prints Pearson correlation and its p-value. This function also returns a matrix of p-1 rows and 5 columns entitled: “cause", “response", “strength", “corr." and “p-value", respectively with self-explanatory titles. The first two columns have names of variables x1 or x(1+j), depending on which is the cause. The ‘strength’ column has absolute value of summary index in range [0,100] providing summary of causal results based on preponderance of evidence from Cr1 to Cr3 from four orders of stochastic dominance, etc. The order of input columns matters. The fourth column ‘corr.’ reports the Pearson correlation coefficient while the fifth column has the p-value for testing the null of zero Pearson coeff. This function calls `silentPairs0` (the older version) allowing for control variables. The output of this function can be sent to ‘xtable’ for a nice Latex table.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. Since Cr1 to Cr3 near unanimously suggest ‘crim’ as the cause of ‘off’, strength index 100 suggests unanimity. `attach(EuroCrime); causeSummary0(cbind(crim,off))`. Both versions give identical result for this example. Old version of Cr1 using gradients was also motivated by the same Hausman-Wu test statistic.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. ‘Generalized Correlation and Kernel Causality with Applications in Development Economics’ in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See `bootPairs`

See `someCPairs`

`silentPairs`

Examples

```
## Not run:
mtx=as.matrix(mtcars[,1:3])
ctrl=as.matrix(mtcars[,4:5])
 causeSummary0(mtx,ctrl,nam=colnames(mtx))

## End(Not run)

options(np.messages=FALSE)
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
causeSummary0(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

`causeSummBlk` *Block Version Kernel causality summary causal paths from three criteria*

Description

Allowing input matrix of control variables, this function produces a 5 column matrix summarizing the results where the estimated signs of stochastic dominance order values, (+1, 0, -1), are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, -1). The final range for the unanimity of sign index is [–100, 100].

Usage

```
causeSummBlk(mtx, nam = colnames(mtx), blksiz = 10, ctrl = 0,
  dig = 6, wt = c(1.2, 1.1, 1.05, 1), sumwt = 4)
```

Arguments

`mtx`	The data matrix with many columns, y the first column is a fixed target and then it is paired with all other columns, one by one, and still called x for the purpose of flipping.
`nam`	vector of column names for `mtx`. Default: colnames(mtx)
`blksiz`	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done
`ctrl`	data matrix for designated control variable(s) outside causal paths
`dig`	Number of digits for reporting (default `dig`=6).
`wt`	Allows user to choose a vector of four alternative weights for SD1 to SD4.
`sumwt`	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The reason for slightly declining sampling unreliability of higher moments is simply that SD4 involves fourth power of the deviations from the mean and SD3 involves 3rd power, etc. The summary results for all three criteria are reported in one matrix called `out`:

Value

If there are p columns in the input matrix, x1, x2, .., xp, say, and if we keep x1 as a common member of all causal-direction-pairs (x1, x(1+j)) for (j=1, 2, .., p-1) which can be flipped. That is, either x1 is the cause or x(1+j) is the cause in a chosen pair. The control variables are not flipped. The printed output of this function reports the results for p-1 pairs indicating which variable (by name) causes which other variable (also by name). It also prints strength or signed summary strength index in range [-100,100]. A positive sign of the strength index means x1 kernel causes x(1+j), whereas negative strength index means x(1+j) kernel causes x1. The function also prints Pearson correlation and its p-value. This function also returns a matrix of p-1 rows and 5 columns entitled: “cause", “response", “strength", “corr." and “p-value", respectively with self-explanatory titles. The first two columns have names of variables x1 or x(1+j), depending on which is the cause. The ‘strength’ column has absolute value of summary index in range [0,100] providing summary of causal results based on preponderance of evidence from Cr1 to Cr3 from four orders of stochastic dominance, etc. The order of input columns matters. The fourth column ‘corr.’ reports the Pearson correlation coefficient while the fifth column has the p-value for testing the null of zero Pearson coeff. This function calls `siPairsBlk` allowing for control variables. The output of this function can be sent to ‘xtable’ for a nice Latex table.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. Since Cr1 to Cr3 near unanimously suggest ‘crim’ as the cause of ‘off’, strength index 100 suggests unanimity. `attach(EuroCrime); causeSummary(cbind(crim,off))`

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. ‘Generalized Correlation and Kernel Causality with Applications in Development Economics’ in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. ’New exogeneity tests and causal paths,’ Chapter 2 in ’Handbook of Statistics: Conceptual Econometrics Using R’, Vol.32, co-editors: H. D. Vinod and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2019, pp. 33-64.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See bootPairs, causeSummary0 has an older version of this function.

See someCPairs

siPairsBlk, causeSummary

Examples

```
## Not run:
mtx=as.matrix(mtcars[,1:3])
ctrl=as.matrix(mtcars[,4:5])
 causeSummBlk(mtx,ctrl,nam=colnames(mtx))

## End(Not run)

options(np.messages=FALSE)
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
causeSummBlk(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

`cofactor` *Compute cofactor of a matrix based on row r and column c.*

Description

Compute cofactor of a matrix based on row r and column c.

Usage

```
cofactor(x, r, c)
```

Arguments

x	matrix whose cofactor is desired to be computed
r	row number
c	column number

Value

cofactor of x, w.r.t. row r and column c.

Note

needs the function ‘minor” in memory. attaches sign (-1)^(r+c) to the minor.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

See Also

`minor(x,r,c)`

Examples

```
## The function is currently defined as
function (x, r, c)
{
    out = minor(x, r, c) * ((-1)^(r + c))
    return(out)
  }
```

`comp_portfo2` *Compares two vectors (portfolios) using stochastic dominance of orders 1 to 4.*

Description

Given two vectors of portfolio returns this function calls the internal function wtdpapb to report the simple means of four sophisticated measures of stochastic dominance. as explained in Vinod (2008).

Usage

```
comp_portfo2(xa, xb)
```

Arguments

xa	Data on returns for portfolio A in the form of a T by 1 vector
xb	Data on returns for portfolio B in the form of a T by 1 vector

Value

Returns four numbers which are averages of four sophisticated measures of stochastic dominance measurements called SD1 to SD4.

Note

It is possible to modify this function to report the median or standard deviation or any other descriptive statistic by changing the line in the code 'oumean = apply(outb,2,mean)' toward the end of this function. A trimmed mean may be of interest when outliers are suspected.

require(np)

Make sure that functions wtdpapb, bigfp, stochdom2 are in the memory. and options(np.messages=FALSE)

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.", "Hands-On Intermediate Econometrics Using R" (2008) World Scientific Publishers: Hackensack, NJ. (Chapter 4) https://www.worldscientific.com/worldscibooks/10.1142/6895

See Also

stochdom2

Examples

```
set.seed(30)
xa=sample(20:30)#generally lower returns
xb=sample(32:40)# higher returns in xb
gp = comp_portfo2(xa, xb)#all Av(sdi) positive means xb dominates
##positive SD1 to SD4 means xb dominates xa as it should
```

---

da internal da

Description

intended for internal use only

Usage

```
da
```

`da2Lag` *internal da2Lag*

Description

intended for internal use

Usage

```
data(da2Lag)
```

Format

The format is: int 4

`depMeas` *depMeas Measure dependence between two vectors.*

Description

An infant may depend on the mother for survival, but not vice versa. Dependence relations need not be symmetric, yet correlation coefficients are symmetric. One way to measure the extent of dependence is to find the max of the absolute values of the two asymmetric correlations using Vinod (2015) definition of generalized (asymmetric) correlation coefficients. It requires a kernel regression of x on y obtained by using the 'np' package and its flipped version. We use a block version of 'gmcmtx0' called 'gmcmtxBlk' to admit several bandwidths.

Usage

```
depMeas(x, y, blksiz = length(x))
```

Arguments

`x`	Vector of data on first variable
`y`	Vector of data on second variable
`blksiz`	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done

Value

A measure of dependence.

Note

This function needs the gmcmtxBlk function which in turn needs the np package.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

See Also

See Also `gmcmtx0` and `gmcmtxBlk`

Examples

```
library(generalCorr)
options(np.messages = FALSE)
x=1:20;y=sin(x)
depMeas(x,y,blksiz=20)
```

`diff.e0` *Internal diff.e0*

Description

Internal diff.e0

Usage

```
data(diff.e0)
```

`dig` *Internal dig*

Description

Intended for internal use

Usage

```
data(dig)
```

Format

The format digs: int 78

`e0` *internal e0*

Description

intended for internal use only

Usage

```
e0
```

`EuroCrime` *European Crime Data*

Description

This data set refers to crime in European countries during 2008. The sources are World Bank and Eurostat. The crime statistics refers to homicides. It avoids possible reporting bias from the presence of police officers, because homicide reporting in most countries is standardized. Typical usage is: `data(EuroCrime);attach(EuroCrime)`. The secondary source 'quandl.com' was used for collecting these data.

Details

The variables included in the dataset are:

- `Country` Name of the European country
- `crim` Per capita crime rate
- `off` Per capita deployment of police officers

generalCorrInfo *generalCorr package description:*

Description

This package provides convenient software tools for causal path determinations using Vinod (2014, 2015) and extends them. A matrix of asymmetric generalized correlations r*(x|y) is reported by the functions `rstar` and `gmcmtx0`. The r*(x|y) measures the strength of the dependence of x on y. If |r*(x|y)|> |r*(y|x)| it suggests that y is more likely the "kernel cause" of x. This package refers to the r* based criterion as criterion 3 (Cr3) and further adds two additional ways of comparing two kernel regressions helping identify the 'cause' called criterion 1 and 2 (Cr1 and Cr2) using absolute values of gradients and residuals, respectively. See references below. The package has one-line commands summarizing all three criteria leading to high (over 70 %) success rates in causal path identifications.

Details

The usual partial correlations are generalized for the asymmetric matrix of r*'s. Partial correlations help asses the effect of x on y after removing the effect of a set of (control) variables. See `parcor_ijk` and `parcor_ridg`. Another way of generalizing partial correlations by using incremental R-square values in kernel regressions are provided in functions `mag_ctrl` and `someMagPairs`.

The package provides additional tools for causal assessment, for printing the causal detections in a clear, comprehensive compact summary form, such as `somePairs`, `some0Pairs`, `someCPairs` for matrix algebra, such as `cofactor`, for outlier detection `get0outlier`, for numerical integration by the trapezoidal rule, stochastic dominance `stochdom2` and `comp_portfo2`, etc. The function `causeSummary` gives an overall summary of causal path results. The compact function `silentPairs` gives one-line summary of causal path strengths, where negative strength means that variable 'causes' the variable in the first column.

The package has a function `pcause` for bootstrap-based statistical inference and another one for a heuristic t-test called `heurist`. Pairwise deletion of missing data is done in `napair`, while triplet-wise deletion is in `naTriplet` intended for use when control variable(s) are also present. If one has panel data, functions `PanelLag` and `Panel2Lag` are relevant. `pillar3D` provides 3-dimensional plots of data which look more like surfaces, than usual plots with vertical pins.

In simultaneous equation models where endogeneity of regressors is feared, we suggest using Prof. Koopmans' method which suggests ignoring endogeneity issues for all variables "causing" the dependent variable assessed by our three criteria. Weighted summary of all three criteria is in `someCPairs`.

Note

A vignette1 provided with this package generalCorr at CRAN describes the usage of the package with examples. Type the following command: `vignette("generalCorr-vignette",package="generalCorr")` to read the vignette. See also additional citations in the vignette, the references here and their citations for further details.

References

Vinod, H. D.'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in 'Handbook of Statistics: Computational Statistics with R', Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

Zheng, S., Shi, N.-Z., and Zhang, Z. (2012). 'Generalized measures of correlation for asymmetry, nonlinearity, and beyond,' Journal of the American Statistical Association, vol. 107, pp. 1239-1252.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

Vinod, H. D. 'New exogeneity tests and causal paths,' Chapter 2 in 'Handbook of Statistics: Conceptual Econometrics Using R', Vol.32, co-editors: H. D. Vinod and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2019, pp. 33-64.

`get0outliers` *Function to compute outliers and their count using Tukey method using 1.5 times interquartile range (IQR) to define boundaries.*

Description

Function to compute outliers and their count using Tukey method using 1.5 times interquartile range (IQR) to define boundaries.

Usage

```
get0outliers(x, verbo = TRUE, mult = 1.5)
```

Arguments

`x`	vector of data.
`verbo`	set to TRUE(default) assuming printed details are desired.
`mult`	=1.5(default), the number of times IQR is used in defining outlier boundaries.

Value

`below`	which items are lower than the lower limit
`above`	which items are larger than the upper limit
`low.lim`	the lower boundary for outlier detection
`up.lim`	the upper boundary for outlier detection
`nUP`	count of number of data points above upper boundary
`nLO`	count of number of data points below lower boundary

Note

The function removes the missing data before checking for outliers.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
set.seed(101);x=sample(1:100)[1:15];x[16]=150;x[17]=NA
get0outliers(x)#correctly identifies outlier=150
```

---

getSeq — *Two sequences: starting+ending values from n and blocksize (internal use)*

Description

This is an auxiliary function for gmcmtxBlk. It gives sequences of starting and ending values

Usage

```
getSeq(n, blksiz)
```

Arguments

`n`	length of the range
`blksiz`	blocksize

Value

two vectors sqLO and sqUP

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

See Also

gmcmtxBlk

Examples

```
getSeq(n=99, blksiz=10)
```

`gmc0` *internal gmc0*

Description

intended for internal use only

Usage

```
gmc0
```

`gmc1` *internal gmc1*

Description

intended for internal use only

Usage

```
gmc1
```

`gmcmtx0` *Matrix R* of generalized correlation coefficients captures nonlinearities.*

Description

This function checks for missing data for each pair individually. It then uses the `kern` function to kernel regress x on y, and conversely y on x. It needs the library 'np' which reports R-squares of each regression. This function reports their square roots after assigning them the observed sign of the Pearson correlation coefficient. Its advantages are: (i) It is asymmetric yielding causal direction information, by relaxing the assumption of linearity implicit in usual correlation coefficients. (ii) The r* correlation coefficients are generally larger upon admitting arbitrary nonlinearities. (iii) max(|R*ij|, |R*ji|) measures (nonlinear) dependence. For example, let x=1:20 and y=sin(x). This y has a perfect (100 percent) nonlinear dependence on x and yet Pearson correlation coefficient r(xy) -0.0948372 is near zero and usual confidence interval (-0.516, 0.363) includes zero, implying that it is not different from zero. This shows a miserable failure of traditional r(x,y) to measure dependence when nonlinearities are present.

Usage

```
gmcmtx0(mym, nam = colnames(mym))
```

Arguments

`mym`	A matrix of data on variables in columns
`nam`	Column names of the variables in the data matrix

Value

A non-symmetric R* matrix of generalized correlation coefficients

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in 'Handbook of Statistics: Computational Statistics with R', Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

Vinod, H. D. 'New exogeneity tests and causal paths,' Chapter 2 in 'Handbook of Statistics: Conceptual Econometrics Using R', Vol.32, co-editors: H. D. Vinod and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2019, pp. 33-64.

Zheng, S., Shi, N.-Z., and Zhang, Z. (2012). 'Generalized measures of correlation for asymmetry, nonlinearity, and beyond,' Journal of the American Statistical Association, vol. 107, pp. 1239-1252.

See Also

See Also as `gmcmtxBlk` for a more general version using blocking.

Examples

```
gmcmtx0(mtcars[,1:3])

## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
gmcmtx0(x)
## End(Not run)
```

gmcmtxBlk *Matrix R* of generalized correlation coefficients captures nonlinearities using blocks.*

Description

The algorithm uses two auxiliary functions, `getSeq` and `NLhat`. The latter uses the `kern` function to kernel regress x on y, and conversely y on x. It needs the package 'np,' which reports residuals and allows one to compute fitted values (xhat, yhat). Unlike `gmcmtx0`, this function considers blocks of blksiz=10 (default) pairs of data points separately with distinct bandwidths for each block, usually creating superior local fits.

Usage

```
gmcmtxBlk(mym, nam = colnames(mym), blksiz = 10)
```

Arguments

`mym`	A matrix of data on selected variables arranged in columns
`nam`	Column names of the variables in the data matrix
`blksiz`	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done

Details

This function does pairwise checks of missing data for all pairs. Assume that there are n rows in the input matrix 'mym' with some missing rows. If the columns of mym are denoted (X1, X2, ...Xp), we are considering all pairs (Xi, Xj), treated as (x, y), with 'nv' number of valid (non-missing) rows Note that each x and y is an (nv by 1) vector. This function further splits these (x, y) vectors into as many subgroups or blocks as are needed for the nv paired valid data points for the chosen block length (blksiz)

Next, the algorithm strings together various blocks of fitted value vectors (xhat, yhat) also of dimension nv by 1. Now for each pair of Xi Xj (column Xj= cause, row Xi=response, treated as x and y), the algorithm computes R*ij the simple Pearson correlation coefficient between (x, xhat) and as R*ji the correlation coeff. between (y, yhat). Next, it assigns |R*ij| and |R*ji| the observed sign of the Pearson correlation coefficient between x and y.

Its advantages discussed in Vinod (2015, 2019) are: (i) It is asymmetric yielding causal direction information, by relaxing the assumption of linearity implicit in usual correlation coefficients. (ii) The R* correlation coefficients are generally larger upon admitting arbitrary nonlinearities. (iii) max(|R*ij|, |R*ji|) measures (nonlinear) dependence. For example, let x=1:20 and y=sin(x). This y has a perfect (100 percent) nonlinear dependence on x and yet Pearson correlation coefficient r(x y)= -0.0948372 is near zero, and its 95% confidence interval (-0.516, 0.363) includes zero, implying that the population r(x,y) is not significantly different from zero. This example highlights a serious failure of the traditional r(x,y) in measuring dependence between x and y when nonlinearities are present. `gmcmtx0` without blocking does work if x=1:n, and y=f(x)=sin(x) is used with n<20. But for larger n, the fixed bandwidth used by the `kern` function becomes a problem. The block version has additional bandwidths for each block, and hence it correctly quantifies the presence of high dependence even when x=1:n, and y=f(x) are defined for large n and complicated nonlinear functional forms for f(x).

Value

A non-symmetric R* matrix of generalized correlation coefficients

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, `http://dx.doi.org/10.1080/03610918.2015.1122048`

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in 'Handbook of Statistics: Computational Statistics with R', Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

Vinod, H. D. 'New exogeneity tests and causal paths,' Chapter 2 in 'Handbook of Statistics: Conceptual Econometrics Using R', Vol.32, co-editors: H. D. Vinod and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2019, pp. 33-64.

Zheng, S., Shi, N.-Z., and Zhang, Z. (2012). 'Generalized measures of correlation for asymmetry, nonlinearity, and beyond,' Journal of the American Statistical Association, vol. 107, pp. 1239-1252.

Examples

```
## Not run:
x=1:20; y=sin(x)
gmcmtxBlk(cbind(x,y),blksiz=10)
## End(Not run)
```

`gmcmtxZ` *compute the matrix R* of generalized correlation coefficients.*

Description

This function checks for missing data separately for each pair using `kern` function to kernel regress x on y, and conversely y on x. It needs the library 'np' which reports R-squares of each regression. This function reports their square roots with the sign of the Pearson correlation coefficients. Its appeal is that it is asymmetric yielding causal direction information. It avoids the assumption of linearity implicit in the usual correlation coefficients.

Usage

```
gmcmtxZ(mym, nam = colnames(mym))
```

Arguments

mym	A matrix of data on variables in columns
nam	Column names of the variables in the data matrix

Value

A non-symmetric R* matrix of generalized correlation coefficients

Note

This allows the user to change `gmcmtx0` and further experiment with my code.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Examples

```
## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
gmcmtxZ(x)

## End(Not run)
```

---

gmcxy_np   *Function to compute generalized correlation coefficients r*(x|y) and r*(y|x) from two vectors (not matrices)*

Description

This function uses the 'np' package and assumes that there are no missing data.

Usage

```
gmcxy_np(x, y)
```

Arguments

x	vector of x data
y	vector of y data

Value

corxy	r*(x|y) from regressing x on y, where y is the kernel cause.
coryx	r*(y|x) from regressing y on x, where x is the cause.

Note

This is provided if the user want to avoid calling `kern`.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R,' Chapter 4 in 'Handbook of Statistics: Computational Statistics with R,' Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

Examples

```
## Not run:
set.seed(34);x=sample(1:10);y=sample(2:11)
gmcxy_np(x,y)
## End(Not run)
```

---

goodCol *internal goodCol*

Description

intended for internal use only

Usage

```
goodCol
```

heurist — *Heuristic t test of the difference between two generalized correlations.*

Description

Function to run a heuristic t test of the difference between two generalized correlations.

Usage

```
heurist(rxy, ryx, n)
```

Arguments

rxy	generalized correlation r*(x|y) where y is the kernel cause.
ryx	generalized correlation r*(y|x) where x is the kernel cause.
n	Sample size needed to determine the degrees of freedom for the t test.

Value

Prints the t statistics and p-values.

Note

This function requires Revele's R package called 'psych' in memory. This test is known to be conservative (i.e., often fails to reject the null hypothesis of zero difference between the two generalized correlation coefficients.)

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
set.seed(34);x=sample(1:10);y=sample(2:11)
g1=gmcxy_np(x,y)
n=length(x)
h1=heurist(g1$corxy,g1$coryx,n)
print(h1)
print(h1$t) #t statistic
print(h1$p) #p-value
```

`i` *internal i*

Description

intended for internal use

Usage

```
data(i)
```

Format

The format is: int 78

`ibad` *internal object*

Description

intended for internal use

`ii` *internal ii*

Description

intended for internal use

`j` *internal j*

Description

intended for internal use

Usage

```
data(j)
```

Format

The format is: int 4

kern *Kernel regression with options for residuals and gradients.*

Description

Function to run kernel regression with options for residuals and gradients asssuming no missing data.

Usage

```
kern(dep.y, reg.x, tol = 0.1, ftol = 0.1, gradients = FALSE,
  residuals = FALSE)
```

Arguments

dep.y	Data on the dependent (response) variable
reg.x	Data on the regressor (stimulus) variables
tol	Tolerance on the position of located minima of the cross-validation function (default =0.1)
ftol	Fractional tolerance on the value of cross validation function evaluated at local minima (default =0.1)
gradients	Make this TRUE if gradients computations are desired
residuals	Make this TRUE if residuals are desired

Value

Creates a model object 'mod' containing the entire kernel regression output. Type `names(mod)` to reveal the variety of outputs produced by 'npreg' of the 'np' package. The user can access all of them at will by using the dollar notation of R.

Note

This is a work horse for causal identification.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See kern_ctrl.

Examples

```
## Not run:
set.seed(34);x=matrix(sample(1:600)[1:50],ncol=2)
require(np); options(np.messages=FALSE)
k1=kern(x[,1],x[,2])
print(k1$R2) #prints the R square of the kernel regression

## End(Not run)
```

---

kern_ctrl *Kernel regression with control variables and optional residuals and gradients.*

Description

Allowing matrix input of control variables, this function runs kernel regression with options for residuals and gradients.

Usage

```
kern_ctrl(dep.y, reg.x, ctrl, tol = 0.1, ftol = 0.1,
  gradients = FALSE, residuals = FALSE)
```

Arguments

dep.y	Data on the dependent (response) variable
reg.x	Data on the regressor (stimulus) variable
ctrl	Data matrix on the control variable(s) kept outside the causal paths. A constant vector is not allowed as a control variable.
tol	Tolerance on the position of located minima of the cross-validation function (default=0.1)
ftol	Fractional tolerance on the value of cross validation function evaluated at local minima (default=0.1)
gradients	Set to TRUE if gradients computations are desired
residuals	Set to TRUE if residuals are desired

Value

Creates a model object 'mod' containing the entire kernel regression output. If this function is called as `mod=kern_ctrl(x,y,ctrl=z)`, the researcher can simply type `names(mod)` to reveal the large variety of outputs produced by 'npreg' of the 'np' package. The user can access all of them at will using the dollar notation of R.

Note

This is a work horse for causal identification.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See kern.

Examples

```
## Not run:
set.seed(34);x=matrix(sample(1:600)[1:50],ncol=5)
require(np)
k1=kern_ctrl(x[,1],x[,2],ctrl=x[,4:5])
print(k1$R2) #prints the R square of the kernel regression

## End(Not run)
```

---

mag *Approximate overall magnitudes of kernel regression partials dx/dy and dy/dx.*

Description

Uses Vinod (2015) and runs kernel regression of x on y, and also of y on x by using the 'np' package. The function goes on to compute a summary magnitude of the overall approximate partial derivative dx/dy (and dy/dx), after adjusting for units by using an appropriate ratio of standard deviations. Of course, the real partial derivatives of nonlinear functions are generally distinct for each observation.

Usage

```
mag(x, y)
```

Arguments

x	Vector of data on the dependent variable
y	Vector of data on the regressor

Value

vector of two magnitudes of kernel regression partials dx/dy and dy/dx.

Note

This function is intended for use only after the direction of causal path is already determined by various functions in this package (e.g. `somePairs`). For example, if the researcher knows that x causes y, then only dy/dx denoted by dydx is relevant. The other output of the function dxdy is to be ignored. Similarly, only 'dxdy' is relevant if y is known to be the cause of x.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

See Also

See `mag_ctrl`.

Examples

```
set.seed(123);x=sample(1:10);y=1+2*x+rnorm(10)
mag(x,y)#dxdy approx=.5 and dydx approx=2 will be nice.
```

`mag_ctrl` *After removing control variables, magnitude of effect of x on y, and of y on x.*

Description

Uses Vinod (2015) and runs kernel regressions: `x~ y + ctrl` and `x~ ctrl` to evaluate the 'incremental change' in R-squares. Let (rxy;ctrl) denote the square root of that 'incremental change' after its sign is made the same as that of the Pearson correlation coefficient from `cor(x,y)`. One can interpret (rxy;ctrl) as a generalized partial correlation coefficient when x is regressed on y after removing the effect of control variable(s) in `ctrl`. It is more general than the usual partial correlation coefficient, since this one allows for nonlinear relations among variables. Next, the function computes 'dxdy' obtained by multiplying (rxy;ctrl) by the ratio of standard deviations, `sd(x)/sd(y)`. Now our ‘dxdy’ approximates the magnitude of the partial derivative (dx/dy) in a causal model where y is the cause and x is the effect. The function also reports entirely analogous ‘dydx’ obtained by interchanging x and y.

Usage

```
mag_ctrl(x, y, ctrl)
```

Arguments

`x`	Vector of data on the dependent variable.
`y`	Vector of data on the regressor.
`ctrl`	data matrix for designated control variable(s) outside causal paths. A constant vector is not allowed as a control variable.

Value

vector of two magnitudes ‘dxdy’ (effect when x is regressed on y) and ‘dydx’ for reverse regression. Both regressions remove the effect of control variable(s).

Note

This function is intended for use only after the causal path direction is already determined by various functions in this package (e.g. `someCPairs`). That is, after the researcher knows whether x causes y or vice versa. The output of this function is a vector of two numbers: (dxdy, dydx), in that order, representing the magnitude of effect of one variable on the other. We expect the researcher to use only ‘dxdy’ if y is the known cause, or ‘dydx’ if x is the cause. These approximate overall measures may not be well-defined in some applications, because the real partial derivatives of nonlinear functions are generally distinct for each evaluation point.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. ‘Generalized Correlation and Kernel Causality with Applications in Development Economics’ in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. ’Matrix Algebra Topics in Statistics and Economics Using R’, Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C. R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

See Also

See `mag`

Examples

```
set.seed(123);x=sample(1:10); z=runif(10); y=1+2*x+3*z+rnorm(10)
options(np.messages=FALSE)
mag_ctrl(x,y,z)#dx/dy=0.47 is approximately 0.5, but dy/dx=1.41 is not approx=2,
```

`min.e0` *internal min.e0*

Description

intended for internal use only

Usage

```
min.e0
```

`minor` *Function to do compute the minor of a matrix defined by row r and column c.*

Description

Function to do compute the minor of a matrix defined by row r and column c.

Usage

```
minor(x, r, c)
```

Arguments

x	The input matrix
r	The row number
c	The column number

Value

The appropriate 'minor' matrix defined from the input matrix.

Note

This function is needed by the cofactor function.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
 x=matrix(1:20,ncol=4)
minor(x,1,2)
## End(Not run)
```

`mtx` *internal mtx*

Description

intended for internal use only

Usage

```
mtx
```

`mtx0` *internal mtx0*

Description

intended for internal use only

Usage

```
mtx0
```

`mtx2` *internal mtx2*

Description

intended for internal use only

Usage

```
mtx2
```

`n` *internal n*

Description

intended for internal use

Usage

```
n
```

Format

The format is: int 78

`nall` *internal nall*

Description

intended for internal use only

Usage

```
nall
```

`nam.badCol` *internal nam.badCol*

Description

intended for internal use only

Usage

```
nam.badCol
```

nam.goodCol *internal nam.goodCol*

Description

intended for internal use only

Usage

```
nam.goodCol
```

nam.mtx0 *internal nam.mtx0*

Description

intended for internal use only

Usage

```
nam.mtx0
```

napair *Function to do pairwise deletion of missing rows.*

Description

The aim in pair-wise deletions is to retain the largest number of available data pairs with all non-missing data.

Usage

```
napair(x, y)
```

Arguments

x	Vector of x data
y	Vector of y data

Value

newx	A new vector x after removing pairwise missing data
newy	A new vector y after removing pairwise missing data

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
x=sample(1:10);y=sample(1:10);x[2]=NA; y[3]=NA
napair(x,y)
## End(Not run)
```

---

naTriplet — *Function to do matched deletion of missing rows from x, y and control variable(s).*

Description

The aim in three-way deletions is to retain only the largest number of available data triplets with all non-missing data.

Usage

```
naTriplet(x, y, ctrl)
```

Arguments

`x`	Vector of x data
`y`	Vector of y data
`ctrl`	Data matrix on the control variable(s) kept beyond causal path determinations

Value

`newx`	A new vector x after removing triplet-wise missing data
`newy`	A new vector or matrix y after removing triplet-wise missing data
`newctrl`	A new vector or matrix ctrl after removing triplet-wise missing data

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

See Also

See `napair`.

Examples

```
## Not run:
x=sample(1:10);y=sample(1:10);x[2]=NA; y[3]=NA
w=sample(2:11)
naTriplet(x,y,w)
## End(Not run)
```

NLhat *Compute fitted values from kernel regression of x on y and y on x*

Description

This is an auxiliary function for 'gmcmtxBlk.' It uses two numerical vectors (x, y) of same length to create two vectors (xhat, yhat) of fitted values using nonlinear kernel regressions. It uses package 'np' called by `kern` function to kernel regress x on y, and conversely y on x. It uses the option 'residuals=TRUE' of 'kern'

Usage

```
NLhat(x, y)
```

Arguments

x	A column vector of x data
y	A column vector of y data

Value

two vectors named xhat and yhat for fitted values

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

See Also

See Also as `gmcmtxBlk`.

Examples

```
## Not run:
set.seed(34);x=sample(1:15);y=sample(1:15)
NLhat(x,y)
## End(Not run)
```

`out1` *internal out1*

Description

intended for internal use only

Usage

```
out1
```

`p1` *internal p1*

Description

intended for internal use only

Usage

```
p1
```

`Panel2Lag` *Function to compute a vector of 2 lagged values of a variable from panel data.*

Description

The panel data have a set of time series for each entity (e.g. country) arranged such that all time series data for one entity is together. The data for the second entity should be below the entire data for first entity. When a variable is lagged twice, special care is needed to insert NA's for the first two time points (e.g. weeks) for each entity (country).

Usage

```
Panel2Lag(ID, xj)
```

Arguments

`ID`	Location of the column having time identities (e.g. the week number)
`xj`	Data on variable to be lagged linked to ID

Value

Vector containing 2 lagged values of xj.

Note

This function is provided for convenient user modifications.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

See Also

A more general function `PanelLag` has examples.

`PanelLag` *Function for computing a vector of one-lagged values of xj, a variable from panel data.*

Description

Panel data have a set of time series for each entity (e.g. country) arranged such that all time series data for one entity is together, and the data for the second entity should be below the entire data for first entity and so on for entities. In such a data setup, When a variable is lagged once, special care is needed to insert an NA for the first time point in the data (e.g. week) for each entity.

Usage

```
PanelLag(ID, xj, lag = 1)
```

Arguments

`ID`	Location of the column having time identities (e.g. week number).
`xj`	Data vector of variable to be lagged and is linked with the ID.
`lag`	Number of lags desired (lag=1 is the default).

Value

Vector containing one-lagged values of variable xj.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
indiv=gl(6,12,labels=LETTERS[1:6])
#creates A,A,A 12 times B B B also 12 times etc.
set.seed(99);cost=sample(30:90, 72, replace=TRUE)
revenu=sample(50:110, 72, replace=TRUE); month=rep(1:12,6)
df=data.frame(indiv,month,cost,revenu);head(df);tail(df)
L2cost=PanelLag(ID=month,xj=df[,'cost'], lag=2)
head(L2cost)
tail(L2cost)

gmcmtx0(cbind(revenu,cost,L2cost))

gmcxy_np(revenu,cost)

## End(Not run)
```

---

parcorBijk — *Block version of generalized partial correlation coefficients between Xi and Xj, after removing the effect of xk, via nonparametric regression residuals.*

Description

This function uses data on two column vectors, xi, xj and a third xk which can be a vector or a matrix, usually of the remaining variables in the model, including control variables, if any. It first removes missing data from all input variables. Then, it computes residuals of kernel regression (xi on xk) and (xj on xk). This is a block version of parcor_ijk.

Usage

```
parcorBijk(xi, xj, xk, blksiz = 10)
```

Arguments

`xi`	Input vector of data for variable xi
`xj`	Input vector of data for variable xj
`xk`	Input data for variables in xk, usually control variables
`blksiz`	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done

Value

`ouij`	Generalized partial correlation Xi with Xj (=cause) after removing xk
`ouji`	Generalized partial correlation Xj with Xi (=cause) after removing xk

allowing for control variables.

Note

This function calls kern,

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

See Also

See parcor_linear.

Examples

```
## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
options(np.messages=FALSE)
parcorBijk(x[,1], x[,2], x[,3], blksi=10)

## End(Not run)#'
```

---

parcorBMany	*Block version reports many generalized partial correlation coefficients allowing control variables.*

Description

This function calls a block version parcorBijk of the function which uses original data to compute generalized partial correlations between X_{idep} and X_j where j can be any one of the remaining variables in the input matrix mtx. Partial correlations remove the effect of variables X_k other than X_i and X_j. Calculation further allows for the presence of control variable(s) (if any) to remain always outside the input matrix and whose effect is also removed in computing partial correlations.

Usage

```
parcorBMany(mtx, ctrl = 0, dig = 4, idep = 1, blksiz = 10,
  verbo = FALSE)
```

Arguments

mtx	Input data matrix with at least 3 columns.
ctrl	Input vector or matrix of data for control variable(s), default is ctrl=0 when control variables are absent
dig	The number of digits for reporting (=4, default)
idep	The column number of the first variable (=1, default)
blksiz	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done
verbo	Make this TRUE for detailed printing of computational steps

Value

A five column ‘out’ matrix containing partials. The first column has the name of the `idep` variable. The second column has the name of the j variable, while the third column has partial correlation coefficients r*(i,j | k).The last column reports the absolute difference between two partial correlations.

Note

This function reports all partial correlation coefficients, while avoiding ridge type adjustment.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlations and Instantaneous Causality for Data Pairs Benchmark,' (March 8, 2015) http://ssrn.com/abstract=2574891

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

See Also

See Also `parcor_ijk`, `parcorMany`.

Examples

```
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10  #x is partly indep and partly affected by z
y=1+2*x+3*z+rnorm(10)# y depends on x and z not vice versa
mtx=cbind(x,y,z)
parcorBMany(mtx, blksiz=10)


## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
parcorBMany(x, idep=1)

## End(Not run)
```

parcorMany *Report many generalized partial correlation coefficients allowing control variables.*

Description

This function calls parcor_ijk function which uses original data to compute generalized partial correlations between X_{idep} and X_j where j can be any one of the remaining variables in the input matrix mtx. Partial correlations remove the effect of variables x_k other than X_i and X_j. Calculation further allows for the presence of control variable(s) (if any) to remain always outside the input matrix and whose effect is also removed in computing partial correlations.

Usage

```
parcorMany(mtx, ctrl = 0, dig = 4, idep = 1, verbo = FALSE)
```

Arguments

mtx	Input data matrix with at least 3 columns.
ctrl	Input vector or matrix of data for control variable(s), default is ctrl=0 when control variables are absent
dig	The number of digits for reporting (=4, default)
idep	The column number of the first variable (=1, default)
verbo	Make this TRUE for detailed printing of computational steps

Value

A five column 'out' matrix containing partials. The first column has the name of the idep variable. The second column has the name of the j variable, while the third column has partial correlation coefficients r*(i,j | k). The last column reports the absolute difference between two partial correlations.

Note

This function reports all partial correlation coefficients, while avoiding ridge type adjustment.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlations and Instantaneous Causality for Data Pairs Benchmark,' (March 8, 2015) http://ssrn.com/abstract=2574891

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

See Also

See Also parcor_ijk.

Examples

```
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10  #x is partly indep and partly affected by z
y=1+2*x+3*z+rnorm(10)# y depends on x and z not vice versa
mtx=cbind(x,y,z)
parcorMany(mtx)


## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
parcorMany(x, idep=1)

## End(Not run)
```

`parcorMtx` *Matrix of generalized partial correlation coefficients, always leaving out control variables, if any.*

Description

This function calls parcor_ijk function which uses original data to compute generalized partial correlations between X_i and X_j where j can be any one of the remaining variables in the input matrix mtx. Partial correlations remove the effect of variables x_k other than X_i and X_j. Calculation further allows for the presence of control variable(s) (if any) to remain always outside the input matrix and whose effect is also removed in computing partial correlations.

Usage

```
parcorMtx(mtx, ctrl = 0, dig = 4, verbo = FALSE)
```

Arguments

mtx	Input data matrix with p columns. p is at least 3 columns.
ctrl	Input vector or matrix of data for control variable(s), default is ctrl=0 when control variables are absent
dig	The number of digits for reporting (=4, default)
verbo	Make this TRUE for detailed printing of computational steps

Value

A p by p 'out' matrix containing partials r*(i,j | k). and r*(j,i | k).

Note

We want to get all partial correlation coefficient pairs removing other column effects. Vinod (2018) shows why one needs more than one criterion to decide the causal paths or exogeneity.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlations and Instantaneous Causality for Data Pairs Benchmark,' (March 8, 2015) http://ssrn.com/abstract=2574891

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

Vinod, H. D. 'New Exogeneity Tests and Causal Paths,' (June 30, 2018). Available at SSRN: https://ssrn.com/abstract=3206096

See Also

See Also parcor_ijk.

Examples

```
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10  #x is partly indep and partly affected by z
y=1+2*x+3*z+rnorm(10)# y depends on x and z not vice versa
mtx=cbind(x,y,z)
parcorMtx(mtx)


## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
parcorMtx(x)

## End(Not run)
```

---

parcorSilent *Silently compute generalized (ridge-adjusted) partial correlation coefficients from matrix R*.*

Description

This function calls `parcor_ijkOLD` function which uses a generalized correlation matrix R* as input to compute generalized partial correlations between X_i and X_j where j can be any one of the remaining variables. Computation removes the effect of all other variables in the matrix. It further adjusts the resulting partial correlation coefficients to be in the appropriate [-1,1] range by using an additive constant in the fashion of ridge regression.

Usage

```
parcorSilent(gmc0, dig = 4, idep = 1, verbo = FALSE, incr = 3)
```

Arguments

`gmc0`	This must be a p by p matrix R* of generalized correlation coefficients.
`dig`	The number of digits for reporting (=4, default)
`idep`	The column number of the first variable (=1, default)
`verbo`	Make this TRUE for detailed printing of computational steps
`incr`	incremental constant for iteratively adjusting 'ridgek' where ridgek is the constant times the identity matrix used to make sure that the gmc0 matrix is positive definite. If not, this function iteratively increases the `incr` till relevant partial correlations are within the [-1,1] interval.

Value

A five column 'out' matrix containing partials. The first column has the name of the `idep` variable. The second column has the name of the j variable, while the third column has r*(i,j | k). The 4-th column has r*(j,i | k) (denoted partji), and the 5-th column has rijMrji, that is the difference in absolute values (abs(partij) - abs(partji)).

Note

The ridgek constant created by the function during the first round may not be large enough to make sure that that other pairs of r*(i,j | k) are within the [-1,1] interval. The user may have to choose a suitably larger input `incr` to get all relevant partial correlation coefficients in the correct [-1,1] interval.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlations and Instantaneous Causality for Data Pairs Benchmark,' (March 8, 2015) http://ssrn.com/abstract=2574891

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

Vinod, H. D. "A Survey of Ridge Regression and Related Techniques for Improvements over Ordinary Least Squares," Review of Economics and Statistics, Vol. 60, February 1978, pp. 121-131.

See Also

See Also parcor_ijk for a better version using original data as input.

Examples

```
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10  #x is partly indep and partly affected by z
y=1+2*x+3*z+rnorm(10)# y depends on x and z not vice versa
mtx=cbind(x,y,z)
g1=gmcmtx0(mtx)
parcor_ijkOLD(g1,1,2) # ouji> ouij implies i=x is the cause of j=y
parcor_ridg(g1,idep=1)
parcorSilent(g1,idep=1)


## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
gm1=gmcmtx0(x)
parcorSilent(gm1, idep=1)

## End(Not run)
```

---

parcor_ijk — *Generalized partial correlation coefficients between Xi and Xj, after removing the effect of xk, via nonparametric regression residuals.*

Description

This function uses data on two column vectors, xi, xj and a third xk which can be a vector or a matrix, usually of the remaining variables in the model, including control variables, if any. It first removes missing data from all input variables. Then, it computes residuals of kernel regression (xi on xk) and (xj on xk). This version avoids ridge type adjustment present in an older version.

Usage

```
parcor_ijk(xi, xj, xk)
```

Arguments

xi	Input vector of data for variable xi
xj	Input vector of data for variable xj
xk	Input data for variables in xk, usually control variables

Value

`ouij`	Generalized partial correlation Xi with Xj (=cause) after removing xk
`ouji`	Generalized partial correlation Xj with Xi (=cause) after removing xk

allowing for control variables.

Note

This function calls `kern`,

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

See Also

See `parcor_linear`.

Examples

```
## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
options(np.messages=FALSE)
parcor_ijk(x[,1], x[,2], x[,3])

## End(Not run)#'
```

`parcor_ijkOLD` *Generalized partial correlation coefficient between Xi and Xj after removing the effect of all others. (older version, deprecated)*

Description

This function uses a generalized correlation matrix R* as input to compute generalized partial correlations between X_i and X_j where j can be any one of the remaining variables. Computation removes the effect of all other variables in the matrix. The user is encouraged to remove all known irrelevant rows and columns from the R* matrix before submitting it to this function.

Usage

```
parcor_ijkOLD(x, i, j)
```

Arguments

`x`	Input a p by p matrix R* of generalized correlation coefficients.
`i`	A column number identifying the first variable.
`j`	A column number identifying the second variable.

Value

ouij	Partial correlation Xi with Xj (=cause) after removing all other X's
ouji	Partial correlation Xj with Xi (=cause) after removing all other X's
myk	A list of column numbers whose effect has been removed

Note

This function calls `minor`, and `cofactor` and is called by `parcor_ridge`.

Examples

```
## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
gm1=gmcmtx0(x)
parcor_ijkOLD(gm1, 2,3)

## End(Not run)#'
```

---

parcor_linear — *Partial correlation coefficient between Xi and Xj after removing the linear effect of all others.*

Description

This function uses a symmetric correlation matrix R as input to compute usual partial correlations between X_i and X_j where j can be any one of the remaining variables. Computation removes the effect of all other variables in the matrix. The user is encouraged to remove all known irrelevant rows and columns from the R matrix before submitting it to this function.

Usage

```
parcor_linear(x, i, j)
```

Arguments

x	Input a p by p matrix R of symmetric correlation coefficients.
i	A column number identifying the first variable.
j	A column number identifying the second variable.

Value

ouij	Partial correlation Xi with Xj after removing all other X's
ouji	Partial correlation Xj with Xi after removing all other X's
myk	A list of column numbers whose effect has been removed

Note

This function calls minor, and cofactor

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

See Also

See parcor_ijk for generalized partial correlation coefficients useful for causal path determinations.

Examples

```
## Not run:
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
c1=cor(x)
parcor_linear(c1, 2,3)

## End(Not run)
```

---

parcor_ridg — *Compute generalized (ridge-adjusted) partial correlation coefficients from matrix R*. (deprecated)*

Description

This function calls `parcor_ijkOLD` function which uses a generalized correlation matrix R* as input to compute generalized partial correlations between X_i and X_j where j can be any one of the remaining variables. Computation removes the effect of all other variables in the matrix. It further adjusts the resulting partial correlation coefficients to be in the appropriate [-1,1] range by using an additive constant in the fashion of ridge regression.

Usage

```
parcor_ridg(gmc0, dig = 4, idep = 1, verbo = FALSE, incr = 3)
```

Arguments

gmc0	This must be a p by p matrix R* of generalized correlation coefficients.
dig	The number of digits for reporting (=4, default)
idep	The column number of the first variable (=1, default)
verbo	Make this TRUE for detailed printing of computational steps

`incr` incremental constant for iteratively adjusting 'ridgek' where ridgek is the constant times the identity matrix used to make sure that the gmc0 matrix is positive definite. If not iteratively increas the `incr` till all partial correlations are within the [-1,1] interval.

Value

A five column 'out' matrix containing partials. The first column has the name of the `idep` variable. The second column has the name of the j variable, while the third column has r*(i,j | k). The 4-th column has r*(j,i | k) (denoted partji), and the 5-th column has rijMrji, that is the difference in absolute values (abs(partij) - abs(partji)).

Note

The ridgek constant created by the function during the first round may not be large enough to make sure that that other pairs of r*(i,j | k) are within the [-1,1] interval. The user may have to choose a suitably larger input `incr` to get all relevant partial correlation coefficients in the correct [-1,1] interval.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlations and Instantaneous Causality for Data Pairs Benchmark,' (March 8, 2015) http://ssrn.com/abstract=2574891

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

Vinod, H. D. "A Survey of Ridge Regression and Related Techniques for Improvements over Ordinary Least Squares," Review of Economics and Statistics, Vol. 60, February 1978, pp. 121-131.

See Also

See Also `parcor_ijkOLD`.

Examples

```
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10  #x is partly indep and partly affected by z
y=1+2*x+3*z+rnorm(10)# y depends on x and z not vice versa
mtx=cbind(x,y,z)
g1=gmcmtx0(mtx)
parcor_ijkOLD(g1,1,2) # ouji> ouij implies i=x is the cause of j=y
parcor_ridg(g1,idep=1)


## Not run:
```

```
set.seed(34);x=matrix(sample(1:600)[1:99],ncol=3)
colnames(x)=c('V1', 'v2', 'V3')
gm1=gmcmtx0(x)
parcor_ridg(gm1, idep=1)

## End(Not run)
```

---

pcause *Compute the bootstrap probability of correct causal direction.*

Description

Maximum entropy bootstrap ('meboot') package is used for statistical inference regarding δ which equals GMC(X|Y)-GMC(Y|X) defined by Zheng et al (2012). The bootstrap provides an approximation to chances of correct determination of the causal direction.

Usage

```
pcause(x, y, n999 = 999)
```

Arguments

x	Vector of x data
y	Vector of y data
n999	Number of bootstrap replications (default=999)

Value

P(cause) the bootstrap proportion of correct causal determinations.

Note

'pcause' is computer intensive and generally slow. It is better to use it at a later stage in the investigation when a preliminary causal determination is already made. Its use may slow the exploratory phase. In my experience, if P(cause) is less than 0.55, there is a cause for concern.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Zheng, S., Shi, N.-Z., and Zhang, Z. (2012). Generalized measures of correlation for asymmetry, nonlinearity, and beyond. Journal of the American Statistical Association, vol. 107, pp. 1239-1252.

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Examples

```
## Not run:
set.seed(34);x=sample(1:10);y=sample(2:11)
pcause(x,y,n999=29)

data('EuroCrime')
attach(EuroCrime)
pcause(crim,off,n999=29)

## End(Not run)
```

---

pillar3D *Create a 3D pillar chart to display (x, y, z) data coordinate surface.*

Description

Give data on x, y, z coordinate values of a 3D surface, this function plots them after making pillars near each z value by adding and subtracting small amounts dz. Instead of pins of the height z this creates pillars which better resemble a surface. It uses the wireframe() function of 'lattice' package to do the plotting.

Usage

```
pillar3D(z = c(657, 936, 1111, 1201), x = c(280, 542, 722, 1168),
  y = c(162, 214, 186, 246), drape = TRUE, xlab = "y", ylab = "x",
  zlab = "z", mymain = "Pillar Chart")
```

Arguments

z	z-coordinate values
x	x-coordinate values
y	y-coordinate values
drape	logical value, default drape=TRUE to give color to heights
xlab	default "x" label on the x axis

ylab	default "y" label on the y axis
zlab	default "z" label on the z axis
mymain	default "Pillar Chart" main label on the plot

Details

For additional plotting features type 'pillar3D()' on the R console to get my code and adjust wireframe() function defaults.

Value

A 3D plot

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
pillar3D())
## End(Not run)
```

---

prelec2 — *Intermediate weighting function giving Non-Expected Utility theory weights.*

Description

Computes cumulative probabilities and difference between consecutive cumulative probabilities described in Vinod (2008) textbook. This is a simpler version of the version in the book without mapping to non-expected utility theory weights as explained in Vinod (2008).

Usage

```
prelec2(n)
```

Arguments

n	A (usually small) integer.

Value

x	sequence 1:n
p	probabilities p= x[i]/n
pdif	consecutive differences p[i] - p[i - 1]

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Hands-On Intermediate Econometrics Using R' (2008) World Scientific Publishers: Hackensack, NJ. https://www.worldscientific.com/worldscibooks/10.1142/6895

Examples

```
## Not run: prelec2(10)
```

`probSign` *Compute probability of positive or negative sign from bootPairs output*

Description

If there are p columns of data, `probSign` produces a p-1 by 1 vector of probabilities of correct signs assuming that the mean of n999 values has the correct sign and assuming that m of the 'sum' index values inside the range [-tau, tau] are neither positive nor negative but indeterminate or ambiguous (being too close to zero). That is, the denominator of P(+1) or P(-1) is (n999-m) if m signs are too close to zero.

Usage

```
probSign(out, tau = 0.476)
```

Arguments

`out`	output from bootPairs with p-1 columns and n999 rows
`tau`	threshold to determine what value is too close to zero, default tau=0.476 is equivalent to 15 percent threshold for the unanimity index ui

Value

sgn When `mtx` has p columns, `sgn` reports pairwise p-1 signs representing (fixing the first column in each pair) the average sign after averaging the output of of `bootPairs(mtx)` (a n999 by p-1 matrix) each containing resampled 'sum' values summarizing the weighted sums associated with all three criteria from the function `silentPairs(mtx)` applied to each bootstrap sample separately. #'

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. and Lopez-de-Lacalle, J. (2009). 'Maximum entropy bootstrap for time series: The meboot R package.' Journal of Statistical Software, Vol. 29(5), pp. 1-19.

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See Also silentPairs.

Examples

```
## Not run:
options(np.messages = FALSE)
set.seed(34);x=sample(1:10);y=sample(2:11)
bb=bootPairs(cbind(x,y),n999=29)
probSign(bb,tau=0.476) #gives summary stats for n999 bootstrap sum computations

bb=bootPairs(airquality,n999=999);options(np.messages=FALSE)
probSign(bb,tau=0.476)#signs for n999 bootstrap sum computations

data('EuroCrime')
attach(EuroCrime)
bb=bootPairs(cbind(crim,off),n999=29) #col.1= crim causes off
#hence positive signs are more intuitively meaningful.
#note that n999=29 is too small for real problems, chosen for quickness here.
probSign(bb,tau=0.476)#signs for n999 bootstrap sum computations

## End(Not run)
```

---

rhs.lag2 *internal rhs.lag2*

Description

intended for internal use only

Usage

```
rhs.lag2
```

rhs1 *internal rhs1*

Description

intended for internal use only

Usage

```
rhs1
```

ridgek *internal ridgek*

Description

intended for internal use only

Usage

```
ridgek
```

rij *internal rij*

Description

intended for internal use only

Usage

```
rij
```

rijMrji *internal rijMrji*

Description

intended for internal use only

Usage

```
rijMrji
```

`rji` *internal rji*

Description

intended for internal use only

Usage

```
rji
```

`rrij` *internal rrij*

Description

intended for internal use only

Usage

```
rrij
```

`rrji` *internal rrji*

Description

intended for internal use only

Usage

```
rrji
```

rstar *Function to compute generalized correlation coefficients r*(x,y).*

Description

Uses Vinod (2015) definition of generalized (asymmetric) correlation coefficients. It requires kernel regression of x on y obtained by using the 'np' package. It also reports usual Pearson correlation coefficient r and p-value for testing the null hypothesis that (population r)=0.

Usage

```
rstar(x, y)
```

Arguments

x	Vector of data on the dependent variable
y	Vector of data on the regressor

Value

Four objects created by this function are:

corxy	r*x|y or regressing x on y
coryx	r*y|x or regressing y on x
pearson.r	Pearson's product moment correlation coefficient
pv	The p-value for testing the Pearson r

Note

This function needs the kern function which in turn needs the np package.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C.R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

See Also

See Also gmcmtx0 and gmcmtxBlk.

Examples

```
x=sample(1:30);y=sample(1:30); rstar(x,y)
```

`sales2Lag` *internal sales2Lag*

Description

intended for internal use only

Usage

```
sales2Lag
```

`salesLag` *internal salesLag*

Description

intended for internal use only

Usage

```
salesLag
```

`seed` *internal seed*

Description

intended for internal use only

Usage

```
seed
```

`sgn.e0` *internal sgn.e0*

Description

intended for internal use only

Usage

```
sgn.e0
```

`silentMtx` *No-print kernel-causality unanimity score matrix with optional control variables*

Description

Allowing input matrix of control variables and missing data, this function produces a p by p matrix summarizing the results, where the estimated signs of stochastic dominance order values (+1, 0, –1) are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, –1). Final weighted index is always in the range [–3.175, 3.175]. It is converted to the more intuitive range [–100, 100].

Usage

```
silentMtx(mtx, ctrl = 0, dig = 6, wt = c(1.2, 1.1, 1.05, 1),
  sumwt = 4)
```

Arguments

`mtx`	The data matrix with p columns. Denote `x1` as the first column which is fixed and then paired with all other columns, say: `x2`, `x3`, .., `xp`, one by one for the purpose of flipping with `x1`. p must be 2 or more
`ctrl`	data matrix for designated control variable(s) outside causal paths
`dig`	Number of digits for reporting (default `dig=6`).
`wt`	Allows user to choose a vector of four alternative weights for SD1 to SD4.
`sumwt`	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. Why are higher moment estimates less reliable? The higher power of the deviations from the mean needed in their computations lead to greater sampling variability. The summary results for all three criteria are reported in a vector of numbers internally called `crall`:

Value

With p columns in `mtx` argument to this function, `x1` can be paired with a total of p-1 columns (x2, x3, .., xp). Note we never flip any of the control variables with `x1`. This function produces i=1,2,..,p-1 numbers representing the summary sign, or 'sum' from the signs sg1 to sg3 associated with the three criteria: Cr1, Cr2 and Cr3. Note that sg1 and sg2 themselves are weighted signs using weighted sum of signs from four orders of stochastic dominance. In general, a positive sign in the i-th location of the 'sum' output of this function means that `x1` is the kernel cause while the variable in (i+1)-th column of `mtx` is the 'effect' or 'response' or 'endogenous.' The magnitude represents the strength (unanimity) of the evidence for a particular sign. Conversely a negative sign in the i-th location of the 'sum' output of this function means that that the first variable listed as the input to this function is the 'effect,' while the variable in (i+1)-th column of `mtx` is the exogenous kernel cause. This function is a summary of `someCPairs` allowing for control variables.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. The command `attach(EuroCrime); silentPairs(cbind(crim,off))` returns only one number: 3.175, implying a high unanimity strength. The index 3.175 is the highest. The positive sign of the index suggests that 'crim' variable in the first column of the matrix input to this function kernel causes 'off' in the second column of the matrix argument `mtx` to this function.

Interpretation of the output matrix produced by this function is as follows. A negative index means the variable named in the column kernel-causes the variable named in the row. A positive index means the row name variable kernel-causes the column name variable. The abs(index) measures unanimity by three criteria, Cr1 to Cr3 representing the strength of evidence for the identified causal path.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

H. D. Vinod 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See `silentPairs`.

See `someCPairs`, `some0Pairs`

Examples

```
## Not run:
options(np.messages=FALSE)
colnames(mtcars[2:ncol(mtcars)])
silentMtx(mtcars[,1:3],ctrl=mtcars[,4:5]) # mpg paired with others

## End(Not run)

options(np.messages=FALSE)
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
silentMtx(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

`silentMtx0` *Older kernel-causality unanimity score matrix with optional control variables*

Description

Allowing input matrix of control variables and missing data, this function produces a p by p matrix summarizing the results, where the estimated signs of stochastic dominance order values (+1, 0, –1) are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, –1). Final weighted index is always in the range [–3.175, 3.175]. It is converted to the more intuitive range [–100, 100].

Usage

```
silentMtx0(mtx, ctrl = 0, dig = 6, wt = c(1.2, 1.1, 1.05, 1),
  sumwt = 4)
```

Arguments

`mtx`	The data matrix with p columns. Denote x1 as the first column which is fixed and then paired with all other columns, say: x2, x3, .., xp, one by one for the purpose of flipping with x1. p must be 2 or more
`ctrl`	data matrix for designated control variable(s) outside causal paths
`dig`	Number of digits for reporting (default `dig`=6).
`wt`	Allows user to choose a vector of four alternative weights for SD1 to SD4.
`sumwt`	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. Why are higher moment estimates less reliable? The higher power of the deviations from the mean needed in their computations lead to greater sampling variability. The summary results for all three criteria are reported in a vector of numbers internally called `crall`:

Value

With p columns in `mtx` argument to this function, `x1` can be paired with a total of p-1 columns (x2, x3, .., xp). Note we never flip any of the control variables with `x1`. This function produces i=1,2,..,p-1 numbers representing the summary sign, or 'sum' from the signs sg1 to sg3 associated with the three criteria: Cr1, Cr2 and Cr3. Note that sg1 and sg2 themselves are weighted signs using weighted sum of signs from four orders of stochastic dominance. In general, a positive sign in the i-th location of the 'sum' output of this function means that `x1` is the kernel cause while the variable in (i+1)-th column of `mtx` is the 'effect' or 'response' or 'endogenous.' The magnitude represents the strength (unanimity) of the evidence for a particular sign. Conversely a negative sign in the i-th location of the 'sum' output of this function means that that the first variable listed as the input to this function is the 'effect,' while the variable in (i+1)-th column of `mtx` is the exogenous kernel cause. This function allows for control variables.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. The command `attach(EuroCrime); silentPairs(cbind(crim,off))` returns only one number: 3.175, implying a high unanimity strength. The index 3.175 is the highest. The positive sign of the index suggests that 'crim' variable in the first column of the matrix input to this function kernel causes 'off' in the second column of the matrix argument `mtx` to this function.

Interpretation of the output matrix produced by this function is as follows. A negative index means the variable named in the column kernel-causes the variable named in the row. A positive index means the row name variable kernel-causes the column name variable. The abs(index) measures unanimity by three criteria, Cr1 to Cr3 representing the strength of evidence for the identified causal path.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

H. D. Vinod 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See silentPairs0 using older Cr1 criterion based on kernel regression local gradients.

See someCPairs, some0Pairs

Examples

```
## Not run:
options(np.messages=FALSE)
colnames(mtcars[2:ncol(mtcars)])
silentMtx0(mtcars[,1:3],ctrl=mtcars[,4:5]) # mpg paired with others

## End(Not run)

options(np.messages=FALSE)
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
silentMtx0(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

---

silentPairs — *No-print kernel causality scores with control variables Hausman-Wu Criterion 1*

Description

Allowing input matrix of control variables and missing data, this function produces a 3 column matrix summarizing the results where the estimated signs of stochastic dominance order values (+1, 0, -1) are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, -1), always in the range [–3.175, 3.175].

Usage

```
silentPairs(mtx, ctrl = 0, dig = 6, wt = c(1.2, 1.1, 1.05, 1),
  sumwt = 4)
```

Arguments

mtx	The data matrix with p columns. Denote x1 as the first column which is fixed and then paired with all other columns, say: x2, x3, .., xp, one by one for the purpose of flipping with x1. p must be 2 or more

ctrl	data matrix for designated control variable(s) outside causal paths default ctrl=0 which means that there are no control variables used.
dig	Number of digits for reporting (default dig=6).
wt	Allows user to choose a vector of four alternative weights for SD1 to SD4.
sumwt	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The source of slightly declining sampling unreliability of higher moments is the higher power of the deviations from the mean needed in their computations. The summary results for all three criteria are reported in a vector of numbers internally called `crall`:

Value

With p columns in `mtx` argument to this function, `x1` can be paired with a total of p-1 columns (x2, x3, .., xp). Note we never flip any of the control variables with `x1`. This function produces i=1,2,..,p-1 numbers representing the summary sign, or 'sum' from the signs sg1 to sg3 associated with the three criteria: Cr1, Cr2 and Cr3. Note that sg1 and sg2 themselves are weighted signs using weighted sum of signs from four orders of stochastic dominance. In general, a positive sign in the i-th location of the 'sum' output of this function means that `x1` is the kernel cause while the variable in (i+1)-th column of `mtx` is the 'effect' or 'response' or 'endogenous.' The magnitude represents the strength (unanimity) of the evidence for a particular sign. Conversely a negative sign in the i-th location of the 'sum' output of this function means that that the first variable listed as the input to this function is the 'effect,' while the variable in (i+1)-th column of `mtx` is the exogenous kernel cause.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. The command `attach(EuroCrime); silentPairs(cbind(crim,off))` returns only one number: 3.175, implying the highest unanimity strength index, with the positive sign suggesting 'crim' in the first column kernel causes 'off' in the second column of the argument `mtx` to this function.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

H. D. Vinod 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See `bootPairs`, `silentMtx`

See `someCPairs`, `some0Pairs`

Examples

```
## Not run:
options(np.messages=FALSE)
colnames(mtcars[2:ncol(mtcars)])
silentPairs(mtcars[,1:3],ctrl=mtcars[,4:5]) # mpg paired with others

## End(Not run)

options(np.messages=FALSE)
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
silentPairs(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

`silentPairs0` *Older version, kernel causality weighted sum allowing control variables*

Description

Allowing input matrix of control variables and missing data, this function produces a 3 column matrix summarizing the results where the estimated signs of stochastic dominance order values (+1, 0, -1) are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, -1), always in the range [–3.175, 3.175].

Usage

```
silentPairs0(mtx, ctrl = 0, dig = 6, wt = c(1.2, 1.1, 1.05, 1),
  sumwt = 4)
```

Arguments

`mtx` The data matrix with p columns. Denote x1 as the first column which is fixed and then paired with all other columns, say: x2, x3, .., xp, one by one for the purpose of flipping with x1. p must be 2 or more

ctrl	data matrix for designated control variable(s) outside causal paths default ctrl=0 which means that there are no control variables used.
dig	Number of digits for reporting (default dig=6).
wt	Allows user to choose a vector of four alternative weights for SD1 to SD4.
sumwt	Sum of weights can be changed here =4(default).

Details

This uses an older version of the first criterion Cr1 based on absolute values of local gradients of kernel regressions, not absolute Hausman-Wu statistic (RHS variable times kernel residuals). It calls `abs_stdapd` and `abs_stdapdC` The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The source of slightly declining sampling unreliability of higher moments is the higher power of the deviations from the mean needed in their computations. The summary results for all three criteria are reported in a vector of numbers internally called `crall`:

Value

With p columns in `mtx` argument to this function, `x1` can be paired with a total of p-1 columns (x2, x3, .., xp). Note we never flip any of the control variables with `x1`. This function produces i=1,2,..,p-1 numbers representing the summary sign, or 'sum' from the signs sg1 to sg3 associated with the three criteria: Cr1, Cr2 and Cr3. Note that sg1 and sg2 themselves are weighted signs using weighted sum of signs from four orders of stochastic dominance. In general, a positive sign in the i-th location of the 'sum' output of this function means that `x1` is the kernel cause while the variable in (i+1)-th column of `mtx` is the 'effect' or 'response' or 'endogenous.' The magnitude represents the strength (unanimity) of the evidence for a particular sign. Conversely a negative sign in the i-th location of the 'sum' output of this function means that that the first variable listed as the input to this function is the 'effect,' while the variable in (i+1)-th column of `mtx` is the exogenous kernel cause. This function is a summary of `someCPairs` allowing for control variables.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. The command `attach(EuroCrime); silentPairs(cbind(crim,off))` returns only one number: 3.175, implying the highest unanimity strength index, with the positive sign suggesting 'crim' in the first column kernel causes 'off' in the second column of the argument `mtx` to this function.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

H. D. Vinod 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See bootPairs, silentMtx

See someCPairs, some0Pairs

See silentPairs for newer version using more direct Hausman-Wu exogeneity test statistic.

Examples

```
## Not run:
options(np.messages=FALSE)
colnames(mtcars[2:ncol(mtcars)])
silentPairs0(mtcars[,1:3],ctrl=mtcars[,4:5]) # mpg paired with others

## End(Not run)

options(np.messages=FALSE)
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
silentPairs0(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

---

siPairsBlk *Block Version of silentPairs for causality scores with control variables*

Description

Allowing input matrix of control variables and missing data, this function produces a 3 column matrix summarizing the results where the estimated signs of stochastic dominance order values (+1, 0, -1) are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2 and added to the Cr3 estimate as: (+1, 0, -1), always in the range [–3.175, 3.175].

Usage

```
siPairsBlk(mtx, ctrl = 0, dig = 6, blksiz = 10, wt = c(1.2, 1.1,
  1.05, 1), sumwt = 4)
```

Arguments

`mtx`	The data matrix with p columns. Denote `x1` as the first column which is fixed and then paired with all other columns, say: x2, x3, .., xp, one by one for the purpose of flipping with `x1`. p must be 2 or more
`ctrl`	data matrix for designated control variable(s) outside causal paths default ctrl=0 which means that there are no control variables used.
`dig`	Number of digits for reporting (default `dig`=6).
`blksiz`	block size, default=10, if chosen blksiz >n, where n=rows in matrix then blksiz=n. That is, no blocking is done
`wt`	Allows user to choose a vector of four alternative weights for SD1 to SD4.
`sumwt`	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The source of slightly declining sampling unreliability of higher moments is the higher power of the deviations from the mean needed in their computations. The summary results for all three criteria are reported in a vector of numbers internally called `crall`:

Value

With p columns in `mtx` argument to this function, `x1` can be paired with a total of p-1 columns (x2, x3, .., xp). Note we never flip any of the control variables with `x1`. This function produces i=1,2,..,p-1 numbers representing the summary sign, or 'sum' from the signs sg1 to sg3 associated with the three criteria: Cr1, Cr2 and Cr3. Note that sg1 and sg2 themselves are weighted signs using weighted sum of signs from four orders of stochastic dominance. In general, a positive sign in the i-th location of the 'sum' output of this function means that `x1` is the kernel cause while the variable in (i+1)-th column of `mtx` is the 'effect' or 'response' or 'endogenous.' The magnitude represents the strength (unanimity) of the evidence for a particular sign. Conversely a negative sign in the i-th location of the 'sum' output of this function means that that the first variable listed as the input to this function is the 'effect,' while the variable in (i+1)-th column of `mtx` is the exogenous kernel cause.

Note

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers. The command `attach(EuroCrime); silentPairs(cbind(crim,off))` returns only one number: 3.175, implying the highest unanimity strength index, with the positive sign suggesting 'crim' in the first column kernel causes 'off' in the second column of the argument `mtx` to this function.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

H. D. Vinod 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. Causal Paths and Exogeneity Tests in Generalcorr Package for Air Pollution and Monetary Policy (June 6, 2017). Available at SSRN: https://ssrn.com/abstract=2982128

See Also

See bootPairs, silentMtx

See someCPairs, some0Pairs

Examples

```
## Not run:
options(np.messages=FALSE)
colnames(mtcars[2:ncol(mtcars)])
siPairsBlk(mtcars[,1:3],ctrl=mtcars[,4:5]) # mpg paired with others

## End(Not run)

options(np.messages=FALSE)
set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10 #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
siPairsBlk(mtx=cbind(x2,y2), ctrl=cbind(z,w2))
```

`some0Pairs` *Function reporting detailed kernel causality results in a 7-column matrix (uses deprecated criterion 1, no longer recommended but may be useful for second and third criterion typ=2,3)*

Description

The seven columns produced by this function summarize the results where the signs of stochastic dominance order values (+1 or -1) are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2. The weighting is obviously not needed for the third criterion Cr3.

Usage

```
some0Pairs(mtx, dig = 6, verbo = TRUE, rnam = FALSE, wt = c(1.2,
  1.1, 1.05, 1), sumwt = 4)
```

Arguments

mtx	The data matrix in the first column is paired with all others.
dig	Number of digits for reporting (default dig=6).
verbo	Make verbo= TRUE for printing detailed steps.
rnam	Make rnam= TRUE if cleverly created row-names are desired.
wt	Allows user to choose a vector of four alternative weights for SD1 to SD4.
sumwt	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The source of slightly declining sampling unreliability of higher moments is the higher power of the deviations from the mean needed in their computations. The summary results for all three criteria are reported in one matrix called `outVote`:

typ=1 reports ('Y', 'X', 'Cause', 'SD1apd', 'SD2apd', 'SD3apd', 'SD4apd') naming variables identifying 'cause' and measures of stochastic dominance using absolute values of kernel regression gradients (or amorphous partial derivatives, apd-s) being minimized by the kernel regression algorithm while comparing the kernel regression of X on Y with that of Y on X.

typ=2 reports ('Y', 'X', 'Cause', 'SD1res', 'SD2res', 'SD3res', 'SD4res') and measures of stochastic dominance using absolute values of kernel regression residuals comparing regression of X on Y with that of Y on X.

typ=3 reports ('Y', 'X', 'Cause', 'r*x|y', 'r*y|x', 'r', 'p-val') containing generalized correlation coefficients r*, 'r' refers to. Pearson correlation coefficient p-val is the p-value for testing the significance of 'r'

Value

Prints three matrices detailing results for Cr1, Cr2 and Cr3. It also returns a grand summary matrix called 'outVote' which summarizes all three criteria. In general, a positive sign for weighted sum reported in the column 'sum' means that the first variable listed as the input to this function is the 'kernel cause.' For example, crime 'kernel causes' police officer deployment (not vice versa) is indicated by the positive sign of 'sum' (=3.175) reported for that example included in this package.

Note

The output matrix last column for 'mtcars' example has the sum of the scores by the three criteria combined. If 'sum' is positive, then variable X (mpg) is more likely to have been engineered to kernel cause the response variable Y, rather than vice versa.

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See Also somePairs

Examples

```
## Not run:
some0Pairs(mtcars) # first variable is mpg and effect on mpg is of interest

## End(Not run)

## Not run:
data(EuroCrime)
attach(EuroCrime)
some0Pairs(cbind(crim,off))

## End(Not run)
```

---

someCPairs *Kernel causality computations admitting control variables reporting a 7-column matrix (has older Cr1)*

Description

Allowing input matrix of control variables, produce 7 column matrix summarizing the results where the signs of stochastic dominance order values (+1 or -1) are weighted by `wt=c(1.2,1.1,1.05,1)` to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2. The weighting is obviously not needed for the third criterion Cr3.

Usage

```
someCPairs(mtx, ctrl, dig = 6, verbo = TRUE, rnam = FALSE,
  wt = c(1.2, 1.1, 1.05, 1), sumwt = 4)
```

Arguments

mtx	The data matrix with many columns where the first column is fixed and then paired with all other columns, one by one.
ctrl	data matrix for designated control variable(s) outside causal paths
dig	Number of digits for reporting (default dig=6).
verbo	Make verbo= TRUE for printing detailed steps.
rnam	Make rnam= TRUE if cleverly created rownames are desired.
wt	Allows user to choose a vector of four alternative weights for SD1 to SD4.
sumwt	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The source of slightly declining sampling unreliability of higher moments is the higher power of the deviations from the mean needed in their computations. The summary results for all three criteria are reported in one matrix called `outVote`:

typ=1 reports ('Y', 'X', 'Cause', 'SD1apdC', 'SD2apdC', 'SD3apdC', 'SD4apdC') naming variables identifying 'cause' and measures of stochastic dominance using absolute values of kernel regression gradients (or amorphous partial derivatives, apd-s) being minimized by the kernel regression algorithm while comparing the kernel regression of X on Y with that of Y on X. The letter C in the titles reminds presence of control variable(s).

typ=2 reports ('Y', 'X', 'Cause', 'SD1resC', 'SD2resC', 'SD3resC', 'SD4resC') and measures of stochastic dominance using absolute values of kernel regression residuals comparing regression of X on Y with that of Y on X.

typ=3 reports ('Y', 'X', 'Cause', 'r*x|yC', 'r*y|xC', 'r', 'p-val') containing generalized correlation coefficients r*, 'r' refers to. Pearson correlation coefficient p-val is the p-value for testing the significance of 'r'. The letter C in the titles reminds the presence of control variable(s).

Value

Prints three matrices detailing results for Cr1, Cr2 and Cr3. It also returns a grand summary matrix called 'outVote' which summarizes all three criteria. In general, a positive sign for weighted sum reported in the column 'sum' means that the first variable listed as the input to this function is the 'kernel cause.' This function is an extension of `some0Pairs` to allow for control variables. For example, crime 'kernel causes' police officer deployment (not vice versa) is indicated by the positive sign of 'sum' (=3.175) reported for that example included in this package.

Note

The output matrix last column for 'mtcars' example has the sum of the scores by the three criteria combined. If 'sum' is positive, then variable X (mpg) is more likely to have been engineerd to kernel cause the response variable Y, rather than vice versa.

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See Also somePairs, some0Pairs

Examples

```
## Not run:
someCPairs(mtcars[,1:3],ctrl=mtcars[4:5]) # first variable is mpg and effect on mpg is of interest

## End(Not run)

set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10  #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
someCPairs(cbind(x2,y2), cbind(z,w2)) #yields x2 as correct cause
```

---

someCPairs2 — *Kernel causality computations admitting control variables reporting a 7-column matrix, version 2.*

Description

Second version of someCPairs also allows input matrix of control variables, produce 7 column matrix summarizing the results where the signs of stochastic dominance order values (+1 or -1) are weighted by wt=c(1.2,1.1,1.05,1) to compute an overall result for all orders of stochastic dominance by a weighted sum for the criteria Cr1 and Cr2. The weighting is obviously not needed for the third criterion Cr3.

Usage

```
someCPairs2(mtx, ctrl, dig = 6, verbo = TRUE, rnam = FALSE,
  wt = c(1.2, 1.1, 1.05, 1), sumwt = 4)
```

Arguments

`mtx`	The data matrix with many columns where the first column is fixed and then paired with all other columns, one by one.
`ctrl`	data matrix for designated control variable(s) outside causal paths
`dig`	Number of digits for reporting (default `dig=6`).
`verbo`	Make `verbo= TRUE` for printing detailed steps.
`rnam`	Make `rnam= TRUE` if cleverly created rownames are desired.
`wt`	Allows user to choose a vector of four alternative weights for SD1 to SD4.
`sumwt`	Sum of weights can be changed here =4(default).

Details

The reason for slightly declining weights on the signs from SD1 to SD4 is simply that the local mean comparisons implicit in SD1 are known to be more reliable than local variance implicit in SD2, local skewness implicit in SD3 and local kurtosis implicit in SD4. The source of slightly declining sampling unreliability of higher moments is the higher power of the deviations from the mean needed in their computations. The summary results for all three criteria are reported in one matrix called `outVote`:

(typ=1) reports ('Y', 'X', 'Cause', 'SD1.rhserr', 'SD2.rhserr', 'SD3.rhserr', 'SD4.rhserr') naming variables identifying the 'cause' and measures of stochastic dominance using absolute values of kernel regression abs(RHS first regressor*residual) values comparing flipped regressions X on Y versus Y on X. The letter C in the titles reminds presence of control variable(s).

typ=2 reports ('Y', 'X', 'Cause', 'SD1resC', 'SD2resC', 'SD3resC', 'SD4resC') and measures of stochastic dominance using absolute values of kernel regression residuals comparing regression of X on Y with that of Y on X.

typ=3 reports ('Y', 'X', 'Cause', 'r*x|yC', 'r*y|xC', 'r', 'p-val') containing generalized correlation coefficients r*, 'r' refers to. Pearson correlation coefficient p-val is the p-value for testing the significance of 'r'. The letter C in the titles reminds the presence of control variable(s).

Value

Prints three matrices detailing results for Cr1, Cr2 and Cr3. It also returns a grand summary matrix called 'outVote' which summarizes all three criteria. In general, a positive sign for weighted sum reported in the column 'sum' means that the first variable listed as the input to this function is the 'kernel cause.' This function is an extension of `some0Pairs` to allow for control variables. For example, crime 'kernel causes' police officer deployment (not vice versa) is indicated by the positive sign of 'sum' (=3.175) reported for that example included in this package.

Note

The output matrix last column for 'mtcars' example has the sum of the scores by the three criteria combined. If 'sum' is positive, then variable X (mpg) is more likely to have been engineered to kernel cause the response variable Y, rather than vice versa.

The European Crime data has all three criteria correctly suggesting that high crime rate kernel causes the deployment of a large number of police officers.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY.

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

See Also somePairs, some0Pairs

Examples

```
## Not run:
someCPairs2(mtcars[,1:3],ctrl=mtcars[4:5]) # first variable is mpg and effect on mpg is of interest

## End(Not run)

set.seed(234)
z=runif(10,2,11)# z is independently created
x=sample(1:10)+z/10  #x is somewhat indep and affected by z
y=1+2*x+3*z+rnorm(10)
w=runif(10)
x2=x;x2[4]=NA;y2=y;y2[8]=NA;w2=w;w2[4]=NA
someCPairs2(cbind(x2,y2), cbind(z,w2)) #yields x2 as correct cause
```

---

someMagPairs — *Summary magnitudes after removing control variables in several pairs where dependent variable is fixed.*

Description

This builds on the function `mag_ctrl`, where the input matrix `mtx` has p columns. The first column is present in each of the (p-1) pairs. Its output is a matrix with four columns containing the names of variables and approximate overall estimates of the magnitudes of partial derivatives (dy/dx) and (dx/dy) for a distinct (x,y) pair in a row. The estimated overall derivatives are not always well-defined, because the real partial derivatives of nonlinear functions are generally distinct for each observation point.

Usage

```
someMagPairs(mtx, ctrl, dig = 6, verbo = TRUE)
```

Arguments

- `mtx` The data matrix with many columns where the first column is fixed and then paired with all other columns, one by one.
- `ctrl` data matrix for designated control variable(s) outside causal paths. A constant vector is not allowed as a control variable.
- `dig` Number of digits for reporting (default `dig=6`).
- `verbo` Make `verbo= TRUE` for printing detailed steps.

Details

The function `mag_ctrl` has kernel regressions: `x~ y + ctrl` and `x~ ctrl` to evaluate the'incremental change' in R-squares. Let (rxy;ctrl) denote the square root of that 'incremental change' after its sign is made the same as that of the Pearson correlation coefficient from `cor(x,y))`. One can interpret (rxy;ctrl) as a generalized partial correlation coefficient when x is regressed on y after removing the effect of control variable(s) in `ctrl`. It is more general than the usual partial correlation coefficient, since this one allows for nonlinear relations among variables. Next, the function computes 'dxdy' obtained by multiplying (rxy;ctrl) by the ratio of standard deviations, `sd(x)/sd(y)`. Now our 'dxdy' approximates the magnitude of the partial derivative (dx/dy) in a causal model where y is the cause and x is the effect. The function also reports entirely analogous 'dydx' obtained by interchanging x and y.

`someMegPairs` function runs the function `mag_ctrl` on several column pairs in a matrix input `mtx` where the first column is held fixed and all others are changed one by one, reporting two partial derivatives for each row.

Value

Table containing names of Xi and Xj and two magnitudes: (dXidXj, dXjdXi). dXidXj is the magnitude of the effect on Xi when Xi is regressed on Xj (i.e., when Xj is the cause). The analogous dXjdXi is the magnitude when Xj is regressed on Xi.

Note

This function is intended for use only after the causal path direction is already determined by various functions in this package (e.g. `someCPairs`). That is, after the researcher knows whether Xi causes Xj or vice versa. The output of this function is a matrix of 4 columns, where first columns list the names of Xi and Xj and the next two numbers in each row are dXidXj, dXjdXi, respectively, representing the magnitude of effect of one variable on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Vinod, H. D. 'Matrix Algebra Topics in Statistics and Economics Using R', Chapter 4 in Handbook of Statistics: Computational Statistics with R, Vol.32, co-editors: M. B. Rao and C. R. Rao. New York: North Holland, Elsevier Science Publishers, 2014, pp. 143-176.

See Also

See mag_ctrl, someCPairs

Examples

```
set.seed(34);x=sample(1:10);y=1+2*x+rnorm(10);z=sample(2:11)
w=runif(10)
ss=someMagPairs(cbind(y,x,z),ctrl=w)
```

---

somePairs — *Function reporting kernel causality results as a 7-column matrix.(deprecated)*

Description

This function lets the user choose one of three criteria to determine causal direction by setting `typ` as 1, 2 or 3. This function reports results for only one criterion at a time unlike the function `some0Pairs` which summarizes the resulting causal directions for all criteria with suitable weights. If some variables are 'control' variables, use `someCPairs`, C=control.

Usage

```
somePairs(mtx, dig = 6, verbo = FALSE, typ = 1, rnam = FALSE)
```

Arguments

`mtx`	The data matrix in the first column is paired with all others.
`dig`	Number of digits for reporting (default `dig`=6).
`verbo`	Make `verbo= TRUE` for printing detailed steps.
`typ`	Must be 1 (default), 2 or 3 for the three criteria.
`rnam`	Make `rnam= TRUE` if cleverly created rownames are desired.

Details

(typ=1) reports ('Y', 'X', 'Cause', 'SD1apd', 'SD2apd', 'SD3apd', 'SD4apd') nameing variables identifying 'cause' and measures of stochastic dominance using absolute values of kernel regression gradients comparing regresson of X on Y with that of Y on X.

(typ=2) reports ('Y', 'X', 'Cause', 'SD1res', 'SD2res', 'SD3res', 'SD4res') and measures of stochastic dominance using absolute values of kernel regression residuals comparing regresson of X on Y with that of Y on X.

(typ=3) reports ('Y', 'X', 'Cause', 'r*X|Y', 'r*Y|X', 'r', 'p-val') containing generalized correlation coefficients r*, 'r' refers to the Pearson correlation coefficient and p-val column has the p-values for testing the significance of Pearson's 'r'.

Value

A matrix containing causal identification results for one criterion. The first column of the input `mtx` having p columns is paired with (p-1) other columns The output matrix headings are self-explanatory and distinct for each criterion Cr1 to Cr3.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

H. D. Vinod 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

The related function `some0Pairs` may be more useful, since it reports on all three criteria (by choosing typ=1,2,3) and further summarizes their results by weighting to help choose causal paths.

Examples

```
## Not run:
data(mtcars)
somePairs(mtcars)

## End(Not run)
```

`somePairs2` *Function reporting kernel causality results as a 7-column matrix, version 2.*

Description

This function is an alternative implementation of `somePairs` which also lets the user choose one of three criteria to determine causal direction by setting `typ` as 1, 2 or 3. This function reports results for only one criterion at a time unlike the function `some0Pairs` which summarizes the resulting causal directions for all criteria with suitable weights. If some variables are 'control' variables, use `someCPairs`, where notation C=control.

Usage

```
somePairs2(mtx, dig = 6, verbo = FALSE, typ = 1, rnam = FALSE)
```

Arguments

`mtx`	The data matrix in the first column is paired with all others.
`dig`	Number of digits for reporting (default `dig`=6).
`verbo`	Make `verbo= TRUE` for printing detailed steps.
`typ`	Must be 1 (default), 2 or 3 for the three criteria.
`rnam`	Make `rnam= TRUE` if cleverly created rownames are desired.

Details

(typ=1) reports ('Y', 'X', 'Cause', 'SD1.rhserr', 'SD2.rhserr', 'SD3.rhserr', 'SD4.rhserr') naming variables identifying the 'cause,' using Hausman-Wu criterion. It measures of stochastic dominance using absolute values of kernel regression abs(RHS first regressor*residual), comparing flipped regressions X on Y versus Y on X.

(typ=2) reports ('Y', 'X', 'Cause', 'SD1res', 'SD2res', 'SD3res', 'SD4res') and measures of stochastic dominance using absolute values of kernel regression residuals comparing regression of X on Y with that of Y on X.

(typ=3) reports ('Y', 'X', 'Cause', 'r*X|Y', 'r*Y|X', 'r', 'p-val') containing generalized correlation coefficients r*, 'r' refers to the Pearson correlation coefficient and p-val column has the p-values for testing the significance of Pearson's 'r'.

Value

A matrix containing causal identification results for one criterion. The first column of the input `mtx` having p columns is paired with (p-1) other columns The output matrix headings are self-explanatory and distinct for each criterion Cr1 to Cr3.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

H. D. Vinod 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

See Also

The related function `some0Pairs` may be more useful, since it reports on all three criteria (by choosing typ=1,2,3) and further summarizes their results by weighting to help choose causal paths.

Alternative and revised function `somePairs2` implements the Cr1 (first criterion) with a direct estimate of the Hausman-Wu statistic for testing exogeneity.

Examples

```
## Not run:
data(mtcars)
somePairs2(mtcars)

## End(Not run)
```

`sort.abse0` *internal sort.abse0*

Description

intended for internal use only

Usage

```
sort.abse0
```

`sort.e0` *internal sort.e0*

Description

intended for internal use only

Usage

```
sort.e0
```

`sort_matrix` *Sort all columns of matrix x with respect to the j-th column.*

Description

This function can use the sort.list function in R. The reason for using it is that one wants the sort to carry along all columns.

Usage

```
sort_matrix(x, j)
```

Arguments

x	An input matrix with several columns
j	The column number with reference to which one wants to sort

Value

A sorted matrix

Examples

```
set.seed(30)
x=matrix(sample(1:50),ncol=5)
y=sort_matrix(x,3);y
```

---

stdres — *Residuals of kernel regressions of x on y when both x and y are standardized.*

Description

1) Standardize the data to force mean zero and variance unity, 2) kernel regress x on y, with the option 'residuals = TRUE' and finally 3) compute the residuals.

Usage

```
stdres(x, y)
```

Arguments

x	vector of data on the dependent variable
y	data on the regressors which can be a matrix

Details

The first argument is assumed to be the dependent variable. If `stdres(x,y)` is used, you are regressing x on y (not the usual y on x). The regressors can be a matrix with 2 or more columns. The missing values are suitably ignored by the standardization.

Value

kernel regression residuals are returned after standardizing the data on both sides so that the magnitudes of residuals are comparable between regression of x on y on the one hand and regression of y on x on the other.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D. 'Generalized Correlation and Kernel Causality with Applications in Development Economics' in Communications in Statistics -Simulation and Computation, 2015, http://dx.doi.org/10.1080/03610918.2015.1122048

Examples

```
## Not run:
set.seed(330)
x=sample(20:50)
y=sample(20:50)
stdres(x,y)

## End(Not run)
```

`stdz_xy` *Standardize x and y vectors to achieve zero mean and unit variance.*

Description

Standardize x and y vectors to achieve zero mean and unit variance.

Usage

```
stdz_xy(x, y)
```

Arguments

`x`	Vector of data which can have NA's
`y`	Vector of data which can have NA's

Value

`stdx`	standardized values of x
`stdy`	standardized values of y

Note

This works even if there are missing x or y values.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

Examples

```
## Not run:
set.seed(30)
x=sample(20:30)
y=sample(21:31)
stdz_xy(x,y)
## End(Not run)
```

---

stochdom2 — *Compute vectors measuring stochastic dominance of four orders.*

Description

Stochastic dominance originated as a sophisticated comparison of two distributions of stock market returns. The dominating distribution is superior in terms of local mean, variance, skewness and kurtosis respectively, representing dominance orders 1 to 4, without simply computing the four moment summary measures for the entire data. Vinod (2008, sec. 4.3) explains the details. This function uses the output of 'wtdpapb'.

Usage

```
stochdom2(dj, wpa, wpb)
```

Arguments

`dj`	Vector of (unequal) distances of consecutive intervals defined on common support of two probability distributions being compared
`wpa`	Vector of the first set of (weighted) probabilities
`wpb`	Vector of the second set of (weighted) probabilities

Value

`sd1b`	Vector measuring stochastic dominance of order 1, SD1
`sd2b`	Vector measuring stochastic dominance of order 2, SD2
`sd3b`	Vector measuring stochastic dominance of order 3, SD3
`sd4b`	Vector measuring stochastic dominance of order 4, SD4

Note

The input to this function is the output of the function `wtdpapb`.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.', 'Hands-On Intermediate Econometrics Using R' (2008) World Scientific Publishers: Hackensack, NJ. https://www.worldscientific.com/worldscibooks/10.1142/6895

Vinod, H. D. 'Ranking Mutual Funds Using Unconventional Utility Theory and Stochastic Dominance,' Journal of Empirical Finance Vol. 11(3) 2004, pp. 353-377.

See Also

See Also wtdpapb

Examples

```
 ## Not run:
 set.seed(234);x=sample(1:30);y=sample(5:34)
 w1=wtdpapb(x,y) #y should dominate x with mostly positive SDs
 stochdom2(w1$dj, w1$wpa, w1$wpb)
## End(Not run)
```

---

wtdpapb *Creates input for the stochastic dominance function stochdom2*

Description

Stochastic dominance is a sophisticated comparison of two distributions of stock market returns. The dominating distribution is superior in terms of mean, variance, skewness and kurtosis respectively, representing dominance orders 1 to 4, without directly computing four moments. Vinod(2008) sec. 4.3 explains the details. The 'wtdpapb' function creates the input for stochdom2 which in turn computes the stochastic dominance. See Vinod (2004) for details about quantitative stochastic dominance.

Usage

```
wtdpapb(xa, xb)
```

Arguments

xa	Vector of (excess) returns for the first investment option A or values of any random variable being compared to another.
xb	Vector of returns for the second option B

Value

wpa	Weighted vector of probabilities for option A
wpb	Weighted vector of probabilities for option B
dj	Vector of interval widths (distances) when both sets of data are forced on a common support

Note

Function is needed before using stochastic dominance

In Vinod (2008) where the purpose of `wtdpapb` is to map from standard ‘expected utility theory’ weights to more sophisticated ‘non-expected utility theory’ weights using Prelec’s (1998, Econometrica, p. 497) method. These weights are not needed here. Hence we provide the function `prelec2` which does not use Prelec weights at all, thereby simplifying and speeding up the R code provided in Vinod (2008). This function avoids sophisticated ‘non-expected’ utility theory which incorporates commonly observed human behavior favoring loss aversion and other anomalies inconsistent with precepts of the expected utility theory. Such weighting is not needed for our application.

Author(s)

Prof. H. D. Vinod, Economics Dept., Fordham University, NY

References

Vinod, H. D.’, ’Hands-On Intermediate Econometrics Using R’ (2008) World Scientific Publishers: Hackensack, NJ. https://www.worldscientific.com/worldscibooks/10.1142/6895

Vinod, H. D. ’Ranking Mutual Funds Using Unconventional Utility Theory and Stochastic Dominance,’ Journal of Empirical Finance Vol. 11(3) 2004, pp. 353-377.

See Also

See Also `stochdom2`

Examples

```
 ## Not run:
 set.seed(234);x=sample(1:30);y=sample(5:34)
 wtdpapb(x,y)
## End(Not run)
```

Index