

# Package: depcoeff (via r-universe)

September 1, 2024

**Title** Dependency Coefficients

**Type** Package

**Version** 0.0.1

**Date** 2019-12-23

**Author** Eckhard Liebscher

**Maintainer** Eckhard Liebscher <eckhard.liebscher@hs-merseburg.de>

**Description** Functions to compute coefficients measuring the dependence of two or more than two variables. The functions can be deployed to gain information about functional dependencies of the variables with emphasis on monotone functions. The statistics describe how well one response variable can be approximated by a monotone function of other variables. In regression analysis the variable selection is an important issue. In this framework the functions could be useful tools in modeling the regression function. Detailed explanations on the subject can be found in papers Liebscher (2014) <[doi:10.2478/demo-2014-0004](https://doi.org/10.2478/demo-2014-0004)>; Liebscher (2017) <[doi:10.1515/demo-2017-0012](https://doi.org/10.1515/demo-2017-0012)>; Liebscher (2019, submitted).

**Depends** R (>= 3.5.0)

**Suggests** MASS

**License** GPL-2

**Encoding** UTF-8

**LazyData** true

**LinkingTo** Rcpp

**Imports** Rcpp, copula

**RoxygenNote** 7.0.2

**NeedsCompilation** yes

**Date/Publication** 2020-01-23 16:20:05 UTC

**Repository** <https://eliebs.r-universe.dev>

**RemoteUrl** <https://github.com/cran/depcoeff>

**RemoteRef** HEAD

**RemoteSha** 8849413fa66559b67aa990f9fbaeb655f4947c02

## Contents

kendr	2
kendr	3
spearr	4
spearrs	5
zetac	6
zetaci	7
zetapm	8

<b>Index</b>	<b>10</b>
--------------	-----------

---

kendr	<i>Kendall regression coefficient</i>
-------	---------------------------------------

---

### Description

The function `kendr` evaluates the multivariate Kendall regression coefficient. It describes how well the target variable  $y$  can be fit by a function of regressor variables which is increasing w.r.t. some regressors and decreasing w.r.t. the other regressors.

### Usage

```
kendr(x,y,direction=NULL,out=0)
```

### Arguments

<code>x</code>	data matrix of regressor variables
<code>y</code>	data vector of the target variable
<code>direction</code>	vector of length $d$ ( $d$ is number of regressors), value 1 refers to regressors leading to increasing $y$ whenever this regressor increases, value -1 refers to regressors leading to decreasing $y$ whenever this regressor increases. If <code>direction=NULL</code> , then all coefficients are computed.
<code>out</code>	value 1: full output, value 0: reduced output, only coefficients that are largest in absolute value

### Value

list of Kendall regression coefficients for several directions

### References

Eckhard Liebscher (2019). Kendall regression Coefficient. submitted

### Examples

```
library(MASS)
data <- gilgais
kendr(data[,1:3],data[,4],out=1)
```

---

`kendrs`*Kendall regression coefficient for split domains*

---

## Description

The function `kendrs` evaluates the multivariate Kendall regression coefficient for two regressors and split regressor region. It describes how well the target variable can be fit in each split region by a function which is increasing w.r.t. some regressors and decreasing w.r.t. the other regressors.

## Usage

```
kendrs(x,y,splitp=NULL)
```

## Arguments

<code>x</code>	datamatrix of regressor variables with two columns,
<code>y</code>	data vector of the target variable
<code>splitp</code>	vector of length 2 of the splitting points, If <code>p1</code> is the first component of this vector, then the point splits the domain of the first regressor into a left region of fraction <code>p1</code> of data items and a right region of the remaining data items. The same is done for the second regressor. As the result we obtain 4 subregions of the regressor domain. default= <code>c(0.5,0.5)</code>

## Value

list of Kendall regression coefficients for the 4 split regions and the total coefficient together with the corresponding optimal directions. direction `++` means that `y` increases whenever both regressors increases direction `+-` means that `y` increases whenever the first regressor increases and the other regressor decreases..etc.

## References

Eckhard Liebscher (2019). Kendall regression coefficient. submitted

## Examples

```
library(MASS)
data<- gilgais
kendrs(data[,1:2],data[,3],splitp=c(0.4,0.6))
```

---

spearr	<i>Spearman regression coefficient</i>
--------	--

---

### Description

The function `spearr` evaluates the multivariate Spearman regression coefficient. It describes how well the target variable `y` can be fit by a function of regressor variables which is increasing w.r.t. some regressors and decreasing w.r.t. the other regressors.

### Usage

```
spearr(x,y,direction=NULL,out=0)
```

### Arguments

<code>x</code>	data matrix of regressor variables
<code>y</code>	data vector of the target variable
<code>direction</code>	vector of length <code>d</code> ( <code>d</code> is number of regressors), value 1 refers to regressors leading to increasing <code>y</code> whenever this regressor increases, value -1 refers to regressors leading to decreasing <code>y</code> whenever this regressor increases. If <code>direction=NULL</code> , then all coefficients are computed.
<code>out</code>	value 1: full output, value 0: reduced output, only coefficients that are largest in absolute value

### Value

list of Spearman regression coefficients for several directions

### References

Eckhard Liebscher (2019). A copula-based dependence measure for regression analysis. submitted

### Examples

```
library(MASS)
data <- gilgais
spearr(data[,1:3],data[,4],out=1)
```

---

`spearrs`*Spearman regression coefficient for split domains*

---

## Description

The function `spearrs` evaluates the multivariate Spearman regression coefficient for two regressors and split regressor region. It describes how well the target variable can be fit in each split region by a function which is increasing w.r.t. some regressors and decreasing w.r.t. the other regressors.

## Usage

```
spearrs(x,y,splitp=NULL)
```

## Arguments

<code>x</code>	datamatrix of regressor variables with two columns,
<code>y</code>	data vector of the target variable
<code>splitp</code>	vector of length 2 of the splitting points, If <code>p1</code> is the first component of this vector, then the point splits the domain of the first regressor into a left region of fraction <code>p1</code> of data items and a right region of the remaining data items. The same is done for the second regressor. As the result we obtain 4 subregions of the regressor domain. default= <code>c(0.5,0.5)</code>

## Value

list of Kendall regression coefficients for the 4 split regions and the total coefficient together with the corresponding optimal directions. direction `++` means that `y` increases whenever both regressors increases direction `+-` means that `y` increases whenever the first regressor increases and the other regressor decreases..etc.

## References

Eckhard Liebscher (2019). A copula-based dependence measure for regression analysis. submitted

## Examples

```
library(MASS)
data<- gilgais
spearrs(data[,1:2],data[,3],splitp=c(0.4,0.6))
```

zetac

*Zeta dependence coefficient***Description**

zetac is a function to evaluate the zeta dependence coefficient (one interval) of two random variables  $x$  and  $y$  which is based on the copula. Four specific coefficients are available: the Spearman coefficient, Spearman's footrule, the power coefficient and the Huber function coefficient.

**Usage**

```
zetac(x,y,method="Spearman",methodF=1,parH=0.5,parp=1.5)
```

**Arguments**

<code>x, y</code>	data vectors of the two variables whose dependence is analysed.
<code>method</code>	list of names of the coefficients: "Spearman" stands for the Spearman coefficient, "footrule" means Spearman's footrule, "power" stands for the power function coefficient, "Huber" means the Huber function coefficient. If "all" is assigned to method then all methods are used.
<code>methodF</code>	value 1,2 or 3 refers to several methods for computation of the distribution function values, 1 is the default value.
<code>parH</code>	parameter of the Huber function (default 0.5). Valid values for parH are between 0 and 1.
<code>parp</code>	parameter of the power function (default 1.5). The parameter has to be positive.

**Details**

Let  $X_1, \dots, X_n$  be the sample of the  $X$  variable. Formulas for the estimators of values  $F(X_i)$  of the distribution function:  $\text{methodF} = 1 \rightarrow \hat{F}(X_i) = \frac{1}{n}\text{rank}(X_i)$   $\text{methodF} = 2 \rightarrow \hat{F}^1(X_i) = \frac{1}{n+1}\text{rank}(X_i)$   $\text{methodF} = 3 \rightarrow \hat{F}^2(X_i) = \frac{1}{\sqrt{n^2-1}}\text{rank}(X_i)$  The values of the distribution function of  $Y$  are treated analogously.

**Value**

zeta dependence coefficient of two random variables. This coefficient is bounded by 1. The higher the value the stronger is the dependence.

**References**

Eckhard Liebscher (2014). Copula-based dependence measures. *Dependence Modeling* 2 (2014), 49-64

**Examples**

```
library(MASS)
data<- gilgais
zetac(data[,1],data[,2])
```

---

zetaci	<i>Zeta coefficient of piecewise monotonicity with split domain</i>
--------	---

---

**Description**

The function `zetaci` evaluates the coefficient of piecewise monotonicity of variables `x` and `y` where the `x`-domain is split into a fixed number of intervals.

**Usage**

```
zetaci(x,y,a,method="Spearman",methodF=1,parH=0.5,parp=1.5)
```

**Arguments**

<code>x, y</code>	data vectors of the two variables whose dependence is analysed.
<code>a</code>	vector of fractions $a_i, 0 < a_i < a_{i+1} < 1$ for the splitting. A fraction of $a_1, a_2 - a_1, a_3 - a_2 \dots$ of data points are in the corresponding split region. The number of split regions is equal to the length of <code>a</code> plus 1.
<code>method</code>	value (default "Spearman")
<code>methodF</code>	value 1,2 or 3 refers to several methods for computation of the distribution function values, 1 is the default value.
<code>parH</code>	parameter of the Huber function (default 0.5). Valid values for <code>parH</code> are between 0 and 1.
<code>parp</code>	parameter of the power function (default 1.5). The parameter has to be positive.

**Details**

Let  $X_1, \dots, X_n$  be the sample of the  $X$  variable. Formulas for the estimators of values  $F(X_i)$  of the distribution function:  $\text{methodF} = 1 \rightarrow \hat{F}(X_i) = \frac{1}{n} \text{rank}(X_i)$   $\text{methodF} = 2 \rightarrow \hat{F}^1(X_i) = \frac{1}{n+1} \text{rank}(X_i)$   $\text{methodF} = 3 \rightarrow \hat{F}^2(X_i) = \frac{1}{\sqrt{n^2-1}} \text{rank}(X_i)$  The values of the distribution function of  $Y$  are treated analogously.

**Value**

list of zeta dependence coefficients of piecewise monotonicity of two random variables containing the following elements: Spearman...Spearman coefficient footrule...Spearman's footrule power...power coefficient Huber...Huber function coefficient

**References**

Eckhard Liebscher (2017). Copula-based dependence measures for piecewise monotonicity. *Dependence Modeling* 5 (2017), 198-220

**Examples**

```
library(MASS)
data<- gilgais
zetaci(data[, 1], data[, 2], a=c(0.25, 0.5, 0.75))
```

---

 zetapm

*Zeta dependence coefficient of piecewise monotonicity*


---

**Description**

zetapm is a function to evaluate the zeta dependence coefficients of piecewise monotonicity of two random variables  $x$  and  $y$  which is based on the copula. The regressor domain (domain of  $x$ ) is split into two parts. The function searches for the optimal splitting point to obtain maximum dependence. The main part of the function is coded as C++ procedure

**Usage**

```
zetapm(x,y,amin=0.25,method="all",methodF=1,parp=1.5,parH=0.5)
```

**Arguments**

<code>x, y</code>	data vectors of the two variables whose dependence is analysed.
<code>amin</code>	minimum fraction of sample items to be used for one split region
<code>method</code>	vector of chosen special coefficients: Spearman...Spearman coefficient footrule...Spearman's footrule power...power coefficient Huber...Huber function coefficient, "all" refers to all coefficients
<code>methodF</code>	value 1,2 or 3 refers to several methods for computation of the distribution function values, 1 is the default value.
<code>parp</code>	parameter of the power function (default 1.5). The parameter has to be positive.
<code>parH</code>	parameter of the Huber function (default 0.5). Valid values for parH are between 0 and 1.

**Details**

Let  $X_1, \dots, X_n$  be the sample of the  $X$  variable. Formulas for the estimators of values  $F(X_i)$  of the distribution function:  $\text{methodF} = 1 \rightarrow \hat{F}(X_i) = \frac{1}{n}\text{rank}(X_i)$   $\text{methodF} = 2 \rightarrow \hat{F}^1(X_i) = \frac{1}{n+1}\text{rank}(X_i)$   $\text{methodF} = 3 \rightarrow \hat{F}^2(X_i) = \frac{1}{\sqrt{n^2-1}}\text{rank}(X_i)$  The values of the distribution function of  $Y$  are treated analogously.

**Value**

list of zeta dependence coefficients (plusminus coefficient and minusplus one) of piecewise monotonicity of two random variables containing the following elements or a subset of it in this order: Spearman coefficient, footrule, power coefficient, Huber function coefficient. `position1` and `position2` indicate the number of the sample items where the optimized split point is located



## **References**

Eckhard Liebscher (2017). Copula-based dependence measures for piecewise monotonicity. *Dependence Modeling* 5 (2017), 198-220

## **Examples**

```
library(MASS)
data<- gilgais
zetapm(data[,1],data[,2])
```

# Index

kendr, 2  
kendsr, 3

spearr, 4  
spearrs, 5

zetac, 6  
zetaci, 7  
zetapm, 8