Package ‘r2redux’

August 2, 2022

Title R2 Statistic

Version 1.0.10

Description R2 statistic for significance test. Variance and covariance of R2 values used to assess the 95% CI and p-value of the R2 difference.

License GPL (>=3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

NeedsCompilation no

Depends R (>= 2.10)

LazyData true

R topics documented:

`cc_trf` *cc_trf function*

Description

This function transforms the predictive ability (R2) and its standard error (se) between the observed scale and liability scale

Usage

```
cc_trf(R2, se, K, P)
```

Arguments

`R2`	R2 or coefficient of determination on the observed or liability scale
`se`	Standard error of R2
`K`	Population prevalence
`P`	The ratio of cases in the study samples

Value

This function will transform the R2 and its s.e between observed scale and liability scale.Output from the command is the lists of outcomes.

`R2l`	Transformed R2 on the liability scale
`sel`	Transformed se on the liability scale
`R2O`	Transformed R2 on the observed scale
`seO`	Transformed se on the observed scale

References

Lee, S. H., Goddard, M. E., Wray, N. R., and Visscher, P. M. A better coefficient of determination for genetic profile analysis. Genetic epidemiology,(2012). 36(3): p. 214-224.

Examples

```
#To get the transformed R2

output=cc_trf(0.06, 0.002, 0.05, 0.05)
output

#output$R2l (transformed R2 on the liability scale)
#0.2679337

#output$sel (transformed se on the liability scale)
#0.008931123

#output$R2O (transformed R2 on the observed scale)
#0.01343616

#output$seO (transformed se on the observed scale)
#0.000447872
```

`dat1` *Phenotypes and 10 sets of PGSs*

Description

A dataset containing phenotypes and multiple PGSs estimated from 10 sets of SNPs according to GWAS p-value thresholds

Usage

```
dat1
```

Format

A data frame with 1000 rows and 11 variables:

V1 Phenotype, value

V2 PGS1, for p value threshold <=1

V3 PGS2, for p value threshold <=0.5

V4 PGS3, for p value threshold <=0.4

V5 PGS4, for p value threshold <=0.3

V6 PGS5, for p value threshold <=0.2

V7 PGS6, for p value threshold <=0.1

V8 PGS7, for p value threshold <=0.05

V9 PGS8, for p value threshold <=0.01

V10 PGS9, for p value threshold <=0.001

V11 PGS10, for p value threshold <=0.0001

`dat2` *Phenotypes and 2 sets of PGSs*

Description

A dataset containing phenotypes and 2 sets of PGSs estimated from 2 sets of SNPs from regulatroy and non-regulatory genomic regions

Usage

```
dat2
```

Format

A data frame with 1000 rows and 3 variables:

V1 Phenotype

V2 PGS1, regulatory region

V3 PGS2, non-regulatory region

olkin12_1 *olkin12_1 function*

Description

olkin12_1 function

Usage

```
olkin12_1(omat, nv)
```

Arguments

omat	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
nv	Sample size

Value

This function will be used as source code

olkin12_13 *olkin12_13 function*

Description

olkin12_13 function

Usage

```
olkin12_13(omat, nv)
```

Arguments

omat	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
nv	Sample size

Value

This function will be used as source code

olkin12_3 *olkin12_3 function*

Description

olkin12_3 function

Usage

```
olkin12_3(omat, nv)
```

Arguments

omat	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
nv	Sample size

Value

This function will be used as source code

olkin12_34 *olkin12_34 function*

Description

olkin12_34 function

Usage

```
olkin12_34(omat, nv)
```

Arguments

omat	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
nv	Sample size

Value

This function will be used as source code olkin1_2 *olkin1_2 function*

Description

olkin1_2 function

Usage

```
olkin1_2(omat, nv)
```

Arguments

omat	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
nv	Sample size

Value

This function will be used as source code

olkin_beta1_2 *olkin_beta1_2 function*

Description

This function derives Information matrix for beta1^2 and beta2^2 where beta1 and 2 are regression coefficients from a multiple regression model, i.e. y = x1•beta1 + x2•beta2 + e, where y, x1 and x2 are column-standardised, (i.e. in the context of correlation coefficients,see Olkin and Finn 1995).

Usage

```
olkin_beta1_2(omat, nv)
```

Arguments

omat	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
nv	Sample size

Value

This function will give information (variance-covariance) matrix of beta1^2 and beta2^2.To get information (variance-covariance) matrix of beta1^2 and beta2^2. Where beta1 and beta2 are regression coefficients from a multiple regression model. The outputs are listed as follows.

info	2x2 information (variance-covariance) matrix
var1	Variance of beta1^2
var2	Variance of beta2^2
var1_2	Variance of difference between beta1^2 and beta2^2

References

Olkin, I. and Finn, J.D. Correlations redux. Psychological Bulletin, 1995. 118(1): p. 155.

Examples

```
#To get information (variance-covariance) matrix of beta1_2 and beta2_2 where
#beta1 and 2 are regression coefficients from a multiple regression model.
dat=dat1
omat=cor(dat)[1:3,1:3]
#omat
#1.0000000 0.1958636 0.1970060
#0.1958636 1.0000000 0.9981003
#0.1970060 0.9981003 1.0000000

nv=length(dat$V1)
output=olkin_beta1_2(omat,nv)
output

#output$info (2x2 information (variance-covariance) matrix)
#0.04146276 0.08158261
#0.08158261 0.16111124

#output$var1 (variance of beta1^2)
#0.04146276

#output$var2 (variance of beta2^2)
#0.1611112

#output$var1_2 (variance of difference between beta1^2 and beta2^2)
#0.03940878
```

`olkin_beta_inf` *olkin_beta_inf function*

Description

This function derives Information matrix for beta1 and beta2 where beta1 and 2 are regression coefficients from a multiple regression model, i.e. y = x1 * beta1 + x2 * beta2 + e, where y, x1 and x2 are column-standardised (see Olkin and Finn 1995).

Usage

```
olkin_beta_inf(omat, nv)
```

Arguments

`omat`	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
`nv`	Sample size

Value

This function will generate information (variance-covariance) matrix of beta1 and beta2.The outputs are listed as follows.

`info`	2x2 information (variance-covariance) matrix
`var1`	Variance of beta1
`var2`	Variance of beta2
`var1_2`	Variance of difference between beta1 and beta2

References

Olkin, I. and Finn, J.D. Correlations redux. Psychological Bulletin, 1995. 118(1): p. 155.

Examples

```
#To get information (variance-covariance) matrix of beta1 and beta2 where
#beta1 and 2 are regression coefficients from a multiple regression model.
dat=dat1
omat=cor(dat)[1:3,1:3]
#omat
#1.0000000 0.1958636 0.1970060
#0.1958636 1.0000000 0.9981003
#0.1970060 0.9981003 1.0000000

nv=length(dat$V1)
output=olkin_beta_inf(omat,nv)
output

#output$info (2x2 information (variance-covariance) matrix)
#0.2531406 -0.2526212
#-0.2526212  0.2530269

#output$var1 (variance of beta1)
#0.2531406

#output$var2 (variance of beta2)
#0.2530269

#output$var1_2 (variance of difference between beta1 and beta2)
#1.01141
```

`olkin_beta_ratio` *olkin_beta_ratio function*

Description

This function derives variance of beta1^2 / R^2 where beta1 and 2 are regression coefficients from a multiple regression model, i.e. y = x1*beta1 + x2*beta2 + e, where y, x1 and x2 are column-standardised (see Olkin and Finn 1995).

Usage

```
olkin_beta_ratio(omat, nv)
```

Arguments

`omat`	3 by 3 matrix having the correlation coefficients between y, x1 and x2, i.e. omat=cor(dat) where dat is N by 3 matrix having variables in the order of cbind (y,x1,x2)
`nv`	sampel size

Value

This function will generate the variance of the proportion, i.e. beta1^2/R^2.The outputs are listed as follows.

`ratio_var`	Variance of ratio

References

Olkin, I. and Finn, J.D. Correlations redux. Psychological Bulletin, 1995. 118(1): p. 155.

Examples

```
#To get information (variance-covariance) matrix of beta1 and beta2 where
#beta1 and 2 are regression coefficients from a multiple regression model.
dat=dat1
omat=cor(dat)[1:3,1:3]
#omat
#1.0000000 0.1958636 0.1970060
#0.1958636 1.0000000 0.9981003
#0.1970060 0.9981003 1.0000000

nv=length(dat$V1)
output=olkin_beta_ratio(omat,nv)
output

#r2redux output

#output$ratio_var (Variance of ratio)
#27.20206
```

`r2_beta_var` *r2_beta_var*

Description

This function estimates var(beta1^2) and (beta2^2), and beta1 and 2 are regression coefficients from a multiple regression model, i.e. y = x1*beta1 + x2*beta2 +e, y, x1 and x2 are column-standardised (see Olkin and Finn 1995). y is N by 1 matrix having the dependent variable, x1 is N by 1 matrix having the ith explanatory variable. x2 is N by 1 matrix having the jth explanatory variable. v1 and v2 indicates the ith and jth column in the data (v1 or v2 should be a single interger between 1 - M, see Arguments below).

Usage

```
r2_beta_var(dat, v1, v2, nv)
```

Arguments

`dat`	N by (M+1) matrix having variables in the order of cbind(y,x)
`v1`	This can be set as v1=1, v1=2, v1=3 or any value between 1 - M based on combination
`v2`	This can be set as v2=1, v2=2, v2=3, or any value between 1 - M based on combination
`nv`	Sample size

Value

This function will estiamte the variance of beta1^2 and beta2^2 and, and the covariance between beta1^2 and beta2^2, i.e. the information matrix of squared regression coefficients. beta1 and beta2 are regression coefficients from a multiple regression model, i.e. y = x1*beta1 + x2*beta2 +e, where y, x1 and x2 are column-standardised. The outputs are listed as follows.

`beta1_sq`	beta1_sq
`beta2_sq`	beta2_sq
`var1`	Variance of beta1_sq
`var2`	Variance of beta2_sq
`var1_2`	Variance of difference between t1 and t2
`cov`	Covariance between t1 and t2
`upper_beta1_sq`	upper limit of 95% CI for beta1_sq
`lower_beta1_sq`	lower limit of 95% CI for beta1_sq
`upper_beta2_sq`	upper limit of 95% CI for beta2_sq
`lower_beta2_sq`	lower limit of 95% CI for beta2_sq

References

Olkin, I. and Finn, J.D. Correlations redux. Psychological Bulletin, 1995. 118(1): p. 155.

Examples

```
#To get the 95% CI of beta1_sq and beta2_sq
#beta1 and beta2 are regression coefficients from a multiple regression model,
#i.e. y = x1.beta1 + x2.beta2 +e, where y, x1 and x2 are column-standardised.

dat=dat2
nv=length(dat$V1)
v1=c(1)
v2=c(2)
output=r2_beta_var(dat,v1,v2,nv)
output
#r2redux output
#output$beta1_sq (beta1_sq)
#0.01118301

#output$beta2_sq (beta2_sq)
```

```
#0.004980285

#output$var1 (variance of beta1_sq)
#7.072931e-05

#output$var2 (variance of beta2_sq)
#3.161929e-05

#output$var1_2 (variance of difference between beta1_sq and beta2_sq)
#0.000162113

#output$cov (covariance between t1 and t2)
#-2.988221e-05

#output$upper_beta1_sq (upper limit of 95% CI for beta1_sq)
#0.03037793

#output$lower_beta1_sq (lower limit of 95% CI for beta1_sq)
#-0.00123582

#output$upper_beta2_sq (upper limit of 95% CI for beta2_sq)
#0.02490076

#output$lower_beta2_sq (lower limit of 95% CI for beta2_sq)
#-0.005127546
```

`r2_diff` *r2_diff function*

Description

This function estimates var(R2(y~x[,v1]) - R2(y~x[,v2])) where R2 is the R squared value of the model, y is N by 1 matrix having the dependent variable, and x is N by M matrix having M explanatory variables. v1 or v2 indicates the ith column in the x matrix (v1 or v2 can be multiple values between 1 - M, see Arguments below)

Usage

```
r2_diff(dat, v1, v2, nv)
```

Arguments

`dat`	N by (M+1) matrix having variables in the order of cbind(y,x)
`v1`	This can be set as v1=c(1) or v1=c(1,2)
`v2`	This can be set as v2=c(2), v2=c(3), v2=c(1,3) or v2=c(3,4)
`nv`	Sample size

Value

This function will estimate significant difference between two PGS (either dependent or independent and joint or single). To get the test statistics for the difference between R2(y~x[,v1]) and R2(y~x[,v2]). (here we define R2_1=R2(y~x[,v1])) and R2_2=R2(y~x[,v2]))). The outputs are listed as follows.

rsq1	R2_1
rsq2	R2_2
var1	Variance of R2_1
var2	variance of R2_2
var_diff	Variance of difference between R2_1 and R2_2
r2_based_p	two tailed P-value for significant difference between R2_1 and R2_2
r2_based_p_one_tail	one tailed P-value for significant difference
mean_diff	Differences between R2_1 and R2_2
upper_diff	Upper limit of 95% CI for the difference
lower_diff	Lower limit of 95% CI for the difference

Examples

```
#To get the test statistics for the difference between R2(y~x[,v1]) and
#R2(y~x[,v2]). (here we define R2_1=R2(y~x[,v1])) and R2_2=R2(y~x[,v2])))

dat=dat1
nv=length(dat$V1)
v1=c(1)
v2=c(2)
output=r2_diff(dat,v1,v2,nv)
output

#r2redux output

#output$rsq1 (R2_1)
#0.03836254

#output$rsq2 (R2_2)
#0.03881135

#output$var1 (variance of R2_1)
#0.0001436128

#output$var2 (variance of R2_2)
#0.0001451358

#output$var_diff (variance of difference between R2_1 and R2_2)
#5.678517e-07

#output$r2_based_p (two tailed p-value for significant difference between R2_1 and R2_2)
#0.5514562

#output$r2_based_p_one_tail(one tailed p-value for significant difference)
#0.2757281

#output$mean_diff (differences between R2_1 and R2_2)
#-0.0004488044

#output$upper_diff (upper limit of 95% CI for the difference)
#0.001028172
```

```
#output$lower_diff (lower limit of 95% CI for the difference)
#-0.001925781


#To get the test statistics for the difference between R2(y~x[,v1]+x[,v2]) and
#R2(y~x[,v2]). (here R2_1=R2(y~x[,v1]+x[,v2]) and R2_2=R2(y~x[,v1]))

dat=dat1
nv=length(dat$V1)
v1=c(1,2)
v2=c(1)
output=r2_diff(dat,v1,v2,nv)

#r2redux output

#output$rsq1 (R2_1)
#0.03896678

#output$rsq2 (R2_2)
#0.03836254

#output$var1 (variance of R2_1)
#0.0001473686

#output$var2 (variance of R2_2)
#0.0001436128

#output$var_diff (variance of difference between R2_1 and R2_2)
#2.321425e-06

#output$r2_based_p (p-value for significant difference between R2_1 and R2_2)
#0.4366883

#output$mean_diff (differences between R2_1 and R2_2)
#0.0006042383

#output$upper_diff (upper limit of 95% CI for the difference)
#0.00488788

#output$lower_diff (lower limit of 95% CI for the difference)
#-0.0005576171
```

`r2_enrich_beta` *r2_enrich_beta*

Description

This function estimates var(beta1^2/R^2), beta1 and R^2 are regression coefficient and the coefficient of determination from a multiple regression model, i.e. y = x1*beta1 + x2*beta2 +e, where y, x1 and x2 are column-standardised (see Olkin and Finn 1995). y is N by 1 matrix having the dependent variable, and x1 is N by 1 matrix having the ith explanatory variables. x2 is N by 1 matrix having the jth explanatory variables. v1 and v2 indicates the ith and jth column in the data (v1 or v2 should be a single interger between 1 - M, see Arguments below).

Usage

```
r2_enrich_beta(dat, v1, v2, nv, exp1)
```

Arguments

`dat`	N by (M+1) matrix having variables in the order of cbind(y,x)
`v1`	These can be set as v1=1, v1=2, v1=3 or any value between 1 - M based on combination
`v2`	These can be set as v2=1, v2=2, v2=3, or any value between 1 - M based on combination
`nv`	Sample size
`exp1`	The expectation of the ratio (e.g. ratio of # SNPs in genomic partitioning)

Value

This function will test the ratio which is significantly different from the expectation.To get the test statistic for the ratio which is significantly different from the expectation. var[(t1/exp)-(t2/(1-exp))], where t1 = beta1^2 and t2 = beta2^2. beta1 and beta2 are regression coefficients from a multiple regression model, i.e. y = x1.beta1 + x2.beta2 +e, where y, x1 and x2 are column-standardised. The outputs are listed as follows.

`beta1_sq`	beta1_sq
`beta2_sq`	beta2_sq
`ratio1`	beta1_sq/R^2
`ratio2`	t1/R^2
`ratio_var1`	variance of ratio 1
`ratio_var2`	variance of ratio 2
`upper_ratio1`	upper limit of 95% CI for ratio 1
`lower_ratio1`	lower limit of 95% CI for ratio 1
`upper_ratio2`	upper limit of 95% CI for ratio 2
`lower_ratio2`	lower limit of 95% CI for ratio 2
`enrich_p1`	two tailed P-value for beta1_sq/R^2 is significantly different from exp1
`enrich_p1_one_tail`	one tailed P-value for beta1_sq/R^2 is significantly different from exp1
`enrich_p2`	P-value for beta2_sq/R2 is significantly different from (1-exp1)
`enrich_p2_one_tail`	one tailed P-value for beta2_sq/R2 is significantly different from (1-exp1)

References

Olkin, I. and Finn, J.D. Correlations redux. Psychological Bulletin, 1995. 118(1): p. 155.

Examples

```
#To get the test statistic for the ratio which is significantly
#different from the expectation, this function estiamtes
#var (beta1^2/R^2), where
#beta1^2 and R^2 are regression coefficients and the
#coefficient of dterminationfrom a multiple regression model,
#i.e. y = x1*beta1 + x2*beta2 +e, where y, x1 and x2 are
#column-standardised.

dat=dat2
nv=length(dat$V1)
v1=c(1)
v2=c(2)
expected_ratio=0.04
output=r2_enrich_beta(dat,v1,v2,nv,expected_ratio)
output

#r2redux output

#output$beta1_sq (beta1_sq)
#0.01118301

#output$beta2_sq (beta2_sq)
#0.004980285

#output$ratio1 (beta1_sq/R^2)
#0.4392572

#output$ratio2 (beta2_sq/R^2)
#0.1956205

#output$ratio_var1 (variance of ratio 1)
#0.08042288

#output$ratio_var2 (variance of ratio 2)
#0.0431134

#output$upper_ratio1 (upper limit of 95% CI for ratio 1)
#0.9950922

#output$lower_ratio1 (lower limit of 95% CI for ratio 1)
#-0.1165778

#output$upper_ratio2 upper limit of 95% CI for ratio 2)
#0.6025904

#output$lower_ratio2 (lower limit of 95% CI for ratio 2)
#-0.2113493

#output$enrich_p1 (two tailed P-value for beta1_sq/R^2 is
#significantly different from exp1)
#0.1591692

#output$enrich_p1_one_tail (one tailed P-value for beta1_sq/R^2
#is significantly different from exp1)
#0.07958459
```

```
#output$enrich_p2 (P-value for beta2_sq/R2 is significantly
#different from (1-exp1))
#0.000232035

#output$enrich_p2_one_tail (one tailed P-value for beta2_sq/R2 is
#significantly different from (1-exp1))
#0.0001160175
```

---

r2_var *r2_var function*

Description

This function estimates var(R2(y~x[,v1])) where R2 is the R squared value of the model, where R2 is the R squared value of the model, y is N by 1 matrix having the dependent variable, and x is N by M matrix having M explanatory variables. v1 indicates the ith column in the x matrix (v1 can be multiple values between 1 - M, see Arguments below)

Usage

```
r2_var(dat, v1, nv)
```

Arguments

dat	N by (M+1) matrix having variables in the order of cbind(y,x)
v1	This can be set as v1=c(1), v1=c(1,2) or possibly with more values
nv	Sample size

Value

This function will test the null hypothesis for R2. To get the test statistics for R2(y~x[,v1]). The outputs are listed as follows.

rsq	R2
var	Variance of R2
r2_based_p	P-value under the null hypothesis, i.e. R2=0
upper_r2	Upper limit of 95% CI for R2
lower_r2	Lower limit of 95% CI for R2

Examples

```
#To get the test statistics for R2(y~x[,v1])
dat=dat1
nv=length(dat$V1)
v1=c(1)
output=r2_var(dat,v1,nv)
output

#r2redux output
```

```
#output$rsq (R2)
#0.03836254

#output$var (variance of R2)
#0.0001436128

#output$r2_based_p (P-value under the null hypothesis, i.e. R2=0)
#1.188162e-10

#output$upper_r2 (upper limit of 95% CI for R2)
#0.06433782

#output$lower_r2 (lower limit of 95% CI for R2)
#0.01764252


#To get the test statistic for R2(y~x[,v1]+x[,v2]+x[,v3])

dat=dat1
nv=length(dat$V1)
v1=c(1,2,3)
r2_var(dat,v1,nv)

#r2redux output

#output$rsq (R2)
#0.03836254

#output$var (variance of R2)
#0.0001436128

#output$r2_based_p (R2 based P-value)
#1.188162e-10

#output$upper_r2 (upper limit of 95% CI for R2)
#0.06433782

#output$lower_r2 (lower limit of 95% CI for R2)
#0.01764252
```

`r_diff` *r_diff function*

Description

This function estimates var(R(y~x[,v1]) - R(y~x[,v2])) where R is the correlation between y and x, y is N by 1 matrix having the dependent variable, and x is N by M matrix having M explanatory variables. v1 or v2 indicates the ith column in the x matrix (v1 or v2 can be multiple values between 1 - M, see Arguments below)

Usage

```
r_diff(dat, v1, v2, nv)
```

Arguments

`dat`	N by (M+1) matrix having variables in the order of cbind(y,x)
`v1`	This can be set as v1=c(1) or v1=c(1,2)
`v2`	This can be set as v2=c(2), v2=c(3), v2=c(1,3) or v2=c(3,4)
`nv`	Sample size

Value

This function will estimate significant difference between two PGS (either dependent or independent and joint or single). To get the test statistics for the difference between R(y~x[,v1]) and R(y~x[,v2]). (here we define R_1=R(y~x[,v1])) and R_2=R(y~x[,v2]))). The outputs are listed as follows.

`r1`	R_1
`r2`	R_2
`var1`	Variance of R_1
`var2`	variance of R_2
`var_diff`	Variance of difference between R_1 and R_2
`r2_based_p`	P-value for significant difference between R_1 and R_2 for two tailed test
`r_based_p_one_tail`	P-value for significant difference between R_1 and R_2 for one tailed test
`mean_diff`	Differences between R_1 and R_2
`upper_diff`	Upper limit of 95% CI for the difference
`lower_diff`	Lower limit of 95% CI for the difference

Examples

```
#To get the test statistics for the difference between R(y~x[,v1]) and
#R(y~x[,v2]). (here we define R_1=R(y~x[,v1])) and R_2=R(y~x[,v2])))

dat=dat1
nv=length(dat$V1)
v1=c(1)
v2=c(2)
output=r_diff(dat,v1,v2,nv)
output

#r2redux output

#output$r1 (R_1)
#0.1958636

#output$r2 (R_2)
#0.197006

#output$var1 (variance of R_1)
#0.0009247466

#output$var2 (variance of R_1)
#0.0001451358
```

```
#output$var_diff (variance of difference between R_1 and R_2)
#3.65286e-06

#output$r_based_p (two tailed p-value for significant difference between R_1 and R_2)
#0.5500319

#output$r_based_p_one_tail (one tailed p-value for significant difference between R_1 and
#0.2750159

#output$mean_diff
#-0.001142375 (differences between R2_1 and R2_2)

#output$upper_diff (upper limit of 95% CI for the difference)
#0.002603666

#output$lower_diff (lower limit of 95% CI for the difference)
#-0.004888417


#To get the test statistics for the difference between R(y~x[,v1]+[,v2]) and
#R(y~x[,v2]). (here R_1=R(y~x[,v1]+x[,v2]) and R_2=R(y~x[,v1]))

nv=length(dat$V1)
v1=c(1,2)
v2=c(2)
output=r_diff(dat,v1,v2,nv)
output

#output$r1
#0.1974001

#output$r2
#0.197006

#output$var1
#0.0009235848

#output$var2
#0.0009238836

#output$var_diff
#3.837451e-06

#output$r2_based_p
#0.8405593

#output$mean_diff
#0.0003940961

#output$upper_diff
#0.004233621

#output$lower_diff
#-0.003445429
```

Index