Package ‘QRank’

July 21, 2025

Type Package

Title A Novel Quantile Regression Approach for eQTL Discovery

Version 1.0

Date 2016-12-25

Author Xiaoyu Song

Maintainer Xiaoyu Song <`xs2148@cumc.columbia.edu`>

Description
A Quantile Rank-score based test for the identification of expression quantitative trait loci.

License GPL (>= 2)

Imports quantreg

NeedsCompilation no

Repository CRAN

Date/Publication 2017-01-11 18:47:57

Contents

`QRank-package` *A Novel Quantile Regression Tool for eQTL Discovery*

Description

A Quantile Rank-score (QRank) based test for the identification of expression quantitative trait loci (eQTLs).

Details

We use a Quantile Rank-score (QRank) based test to identify the expression quantitative trait loci (eQTLs) that are associated with the conditional quantile function of gene expressions.

Author(s)

Xiaoyu Song

Maintainer: Xiaoyu Song <xs2148@cumc.columbia.edu>

References

Xiaoyu Song, Gen Li, Zhenwei Zhou, Xianling Wang, Iuliana Ionita-Laza and Ying Wei (2016). QRank: A Novel Quantile Regression Tool for eQTL Discovery. *Under revision for Bioinformatics.*

Examples

```
set.seed(123)                                                              #
n=300                                                                      #
x=rbinom(n, 2, 0.2)                                                        #
y=rnorm(n, mean=0, sd=1)                                                   #
z=cbind(rbinom(n, 1, 0.3), rnorm(n, mean=2, sd=2))                        #
taus=c( 0.25, 0.5, 0.75)                                                   #

# - run the proposed QRank approach                                        #
QRank(gene=y, snp=x, cov=z, tau=taus)                                      #

# - output                                                                 #
#Composite.pvalue:                                                         #
#[1] 0.2241873                                                             #

#Quantile.specific.pvalue:                                                 #
#    0.25       0.5      0.75                                              #
#0.5452044 0.1821452 0.5938421                                             #
```

`heter.QRank` *Heterogeneity index for "QRank"*

Description

Calculate the heterogeneity index of quantile regression coefficients at multiple quantile levels. It measures the variation of the quantile coefficients across quantile levels.

$$heterogeneity = log(sd(\beta)/abs(mean(\beta)))$$

where β is the vector of quantile regression coefficients at multiple quantile levels.

Usage

```
heter.QRank(object, newtaus=NULL)
```

Arguments

`object`	Object returned from "QRank"
`newtaus`	a vector of quantile levels based on which heterogeneity index are calculated. Default is NULL, in which case the quantile levels inherited from "QRank" will be used.

Value

`heterogeneity index`	one hetergeneity index

See Also

`QRank`

Examples

```
# continuted from "QRank"

taus=c( 0.25, 0.5, 0.75)
q = QRank(gene=y, snp=x, cov=z, tau=taus)
heter.QRank(q) # default uses taus inherited from "QRank"
# - output
#Heterogeneity index:
#[1] 2.474184
heter.QRank(q,newtaus = 1:9/10) # calculate based on new taus values
# - output
#Heterogeneity index:
#[1] 2.69242
```

`print.QRank` *Print a QRank object*

Description

Print the object of QRank

Usage

```
## S3 method for class 'QRank'
print(x, ...)
```

Arguments

`x`	Object returned from QRank.
`...`	Optional arguments

See Also

`QRank`

`print.QRank.heter` *Print a QRank.heter object*

Description

Print the object of heter.QRank

Usage

```
## S3 method for class 'QRank.heter'
print(x, ...)
```

Arguments

`x`	Object returned from heter.QRank.
`...`	Optional arguments

See Also

`heter.QRank`

`QRank` *A new Quantile Rank-score (QRank) based test for the eQTL identification.*

Description

A function to obtain the p-value on the association between a gene expression and a genetic variant based on quantile rank-score test.

Usage

```
QRank(gene, snp, cov = NULL, tau)
```

Arguments

`gene`	a gene expression level from a selected gene. No parametric assumption is needed for underlying distribution.
`snp`	a selected SNP.
`cov`	a vector or matrix of covariates. Default is NULL.
`tau`	the quantile levels to be estimated. Tau can be a single value or a vector of quantile levels.

Details

This function conducts Quantile Rank-score (QRank) based test for the continuous traits. It can be used to identify expression quantitative trait loci (eQTLs) that are associated with the conditional quantile functions of gene expression.

Value

`composite.pvalue`
: a single p-value for across all quantile levels under consideration, testing H0: No genetic association at the selected quantile levels.

`quantile.specific.pvalue`
: p-values of each quantile level, testing H_0: The genetic variant and gene expression are not associated at this quantile level.

Author(s)

Xiaoyu Song

References

Xiaoyu Song, Gen Li, Zhenwei Zhou, Xianling Wang, Iuliana Ionita-Laza and Ying Wei (2016). QRank: A Novel Quantile Regression Tool for eQTL Discovery. *Under revision for Bioinformatics.*

Examples

```
set.seed(123)                                                         #
n=300                                                                 #
x=rbinom(n, 2, 0.2)                                                   #
y=rnorm(n, mean=0, sd=1)                                              #
z=cbind(rbinom(n, 1, 0.3), rnorm(n, mean=2, sd=2))                    #
taus=c( 0.25, 0.5, 0.75)                                              #

# - run the proposed QRank approach                                   #
QRank(gene=y, snp=x, cov=z, tau=taus)                                 #

# - output                                                            #
#Composite.pvalue:                                                    #
#[1] 0.2241873                                                        #

#Quantile.specific.pvalue:                                            #
#    0.25       0.5      0.75                                         #
#0.5452044 0.1821452 0.5938421                                        #
```

Index