Package 'gwrpvr'

October 13, 2022

Type Package

Title Genome-Wide Regression P-Value (Gwrpv)

Version 1.0

Date 2017-10-13

Author Gregory Connor [aut],
Michael O'Neill [trl, aut, cre]

Maintainer Michael O'Neill <m.oneill@ucd.ie>

Description Computes the sample probability value (p-value)
for the estimated coefficient from a standard
genome-wide univariate regression. It computes the exact
finite-sample p-value under the assumption that the measured
phenotype (the dependent variable in the regression) has
a known Bernoulli-normal mixture distribution.
Finite-sample genome-wide regression p-values (Gwrpv) with
a non-normally distributed phenotype (Gregory Connor and
Michael O'Neill, bioRxiv 204727 <doi:10.1101/204727>).

URL https://doi.org/10.1101/204727

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-10-19 17:47:54 UTC

R topics documented:

`calc_pvalue` *calc_pvalue()*

Description

calculate the pvalue : called from loop_calc_pvalue()

Usage

```
calc_pvalue(n0a, n1a, n2a, n0, n1, n2, pa, pb, x, mua, mub, sumsqx, siga, sigb,
  vary, beta, skipiter, pvalue)
```

Arguments

`n0a`	outer loop index
`n1a`	middle loop index
`n2a`	inner loop index
`n0`	the major allele homozygotes
`n1`	the major allele heterozygotes
`n2`	the minor allele zygotes
`pa`	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
`pb`	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
`x`	a zero mean explanatory variable from the SNP data set
`mua`	parameter of the mixture distribution, can be any real number
`mub`	parameter of the mixture distribution, can be any real number
`sumsqx`	sum of the squares of x
`siga`	parameter of the mixture distribution, can be any real number
`sigb`	parameter of the mixture distribution, can be any real number
`vary`	vary <- pa*(mua^2+siga^2)+pb*(mub^2+sigb^2)-(pa*mua+pb*mub)^2
`beta`	the beta from the regression being tested
`skipiter`	flag to determine if we can skip some calculations
`pvalue`	the input pvalue prior to calculating new improved pvalue

Value

pvalue

close_to_normal *This is a CLT-linked run-time control.*

Description

If the number of observations is large enough that a normality approximation holds for the y average across the major homozygote subsample, then the code skips the time-consuming loop over n0, n1 and n2 and and uses the normal approximation for the average y for the major homozygote subsample. The remaining loop is only over n1 and n2. The only new input/output variables are input lognearnorm (the magnitude of maximum allowed tolerance (in log 10 format) for the sum of squared deviation of skewness and kurtosis from their normal values and output stopiter (a zero if the code does not mandate a stop to the iterative estimation and a one if it does). The input variable lognearnorm has a default value set so that users only have to enter it if they want to over-ride the default value.

Usage

```
close_to_normal(totnobs, n0, n1, n2, pa, pb, mua, mub, siga, sigb, beta,
  nearnorm)
```

Arguments

totnobs	the sum of n0, n1, n2
n0	the major allele homozygotes
n1	the major allele heterozygotes
n2	the minor allele zygotes
pa	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
pb	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
mua	parameter of the mixture distribution, can be any real number
mub	parameter of the mixture distribution, can be any real number
siga	parameter of the mixture distribution, can be any real number
sigb	parameter of the mixture distribution, can be any real number
beta	the beta from the regression being tested
nearnorm	must be in log base 10 format, with default value set to -5

Value

list(skewbeta = skewbeta, kurtbeta = kurtbeta, sigbeta = sigbeta, skipiter = skipiter)

gwrpv *Genome-Wide Regression P-Value (gwrpv) in R*

Description

Computes the sample probability value (p-value) for the estimated coefficient from a standard genome-wide univariate regression. It computes the exact finite-sample p-value under the assumption that the measured phenotype (the dependent variable in the regression) has a known Bernoulli-normal mixture distribution.

Usage

```
gwrpv(beta, n0, n1, n2, mua, siga, mub, sigb, pa, pb, logdelta = -16,
  lognearnorm = -5, logtopsum = 8)
```

Arguments

beta	the beta being tested
n0	number of major allele homozygotes
n1	number of major allele heterozygotes
n2	number of minor allele zygotes
mua	parameter of the mixture distribution, can be any real number
siga	parameter of the mixture distribution, can be any real number
mub	parameter of the mixture distribution, can be any real number
sigb	parameter of the mixture distribution, can be any real number
pa	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
pb	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
logdelta	must be in log base 10 format, with default value set to -16
lognearnorm	must be in log base 10 format, with default value set to -5
logtopsum	must be in log base 10 format, with default value set to 8

Value

gwrpv returns a list containing:

$pvalue p-value of a two-sided hypothesis test for a true coefficient of zero

$skew skewness

$kurt kurtosis of the coefficient estimate under assumed model

$skiptype type of trimming/skip which took place (zero means no trimming)

$totnobs total number of observations

$loopruns number of sums in the main computation for each regression case

.

Examples

```
beta <- 6.05879
n0 <- 499
n1 <- 1
n2 <- 0
mua <- 13.87226
siga <- 2.58807
mub <- 4.62829
sigb <- 2.51803
pa <- 0.96544
pb <- 0.03456    # alternatively: pb <- 1.0 - pa
gwrpv(beta,n0,n1,n2,mua,siga,mub,sigb,pa,pb)

# note default values have been used for the trim parameters above
# in the following example we explicitly set the trim parameters
#
g <- gwrpv(beta,n0,n1,n2,mua,siga,mub,sigb,pa,pb,logdelta=-16,lognearnorm=-5,logtopsum=8)
g$pvalue
```

`gwrpvr` *gwrpvr: A package for calculating Genome-Wide Regression P-Values (gwrpv) in R*

Description

Computes the sample probability value (p-value) for the estimated coefficient from a standard genome-wide univariate regression. It computes the exact finite-sample p-value under the assumption that the measured phenotype (the dependent variable in the regression) has a known Bernoulli-normal mixture distribution.

Details

The gwrpvr package provides two functions: gwrpv and gwrpv_batch.

`gwrpv_batch` *Batch computation of a list of pvalues of GWA regression beta statistics using a bernoulli-normal mixture distribution*

Description

Batch computation of a list of pvalues of GWA regression beta statistics using a bernoulli-normal mixture distribution

Usage

```
gwrpv_batch(regresults, mua, siga, mub, sigb, pa, pb, logdelta = -16,
  lognearnorm = -5, logtopsum = 8)
```

Arguments

regresults	a list of four lists. **$beta** the list of betas being tested **$n0** the list of major allele homozygotes **$n1** the list of major allele heterozygotes **$n2** the list of minor allele zygotes
mua	parameter of the mixture distribution, can be any real number
siga	parameter of the mixture distribution, can be any real number
mub	parameter of the mixture distribution, can be any real number
sigb	parameter of the mixture distribution, can be any real number
pa	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
pb	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
logdelta	must be in log base 10 format, with default value set to -16
lognearnorm	must be in log base 10 format, with default value set to -5
logtopsum	must be in log base 10 format, with default value set to 8

Value

gwrpv_batch returns a list of lists containing the lists:

$pvalue p-value of a two-sided hypothesis test for a true coefficient of zero

$skew skewness

$kurt kurtosis of the coefficient estimate under assumed model

$skiptype type of trimming/skip which took place (zero means no trimming)

$totnobs total number of observations

$loopruns number of sums in the main computation for each regression case

.

Examples

```
beta <- c(6.05879, -6.05879, 2.72055, -2.72055, 1.93347,
         -1.93347, 0.88288, -0.88288, 4.28421, -4.28421)
n0 <- c(499, 499, 495, 495, 490, 490, 451, 451, 998, 998)
n1 <- c(1, 1, 5, 5, 10, 10, 48, 48, 2, 2)
n2 <- c(0, 0, 0, 0, 0, 0, 1, 1, 0, 0)
myregresults <- list(beta = beta, n0 = n0, n1 = n1, n2 = n2)
mua <- 13.87226
siga <- 2.58807
mub <- 4.62829
sigb <- 2.51803
pa <- 0.96544
pb <- 1.0 - pa
```

```
gwrpv_batch(myregresults,mua,siga,mub,sigb,pa,pb)
# store results in a user-defined variable g
g <- gwrpv_batch(myregresults,mua,siga,mub,sigb,pa,pb,logdelta=-16,lognearnorm=-4,logtopsum=8)
g$pvalue
```

`highlow` *highlow()*

Description

If possible, trim the upper and lower bounds

Usage

```
highlow(downtrim, n, pa, pb)
```

Arguments

`downtrim`	lower bound
`n`	upper bound
`pa`	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
`pb`	parameter of the mixture distribution, a real number between zero and one with pa+pb=1

Value

c(lhigh, llow)) # return the new upper and lower bounds

`loop_calc_pvalue` *loop_calc_pvalue()*

Description

calls calc_pvalue()

Usage

```
loop_calc_pvalue(lowone, highone, lowtwo, hightwo, lowthree, highthree, n0a,
  n1a, n2a, n0, n1, n2, pa, pb, x, mua, mub, sumsqx, siga, sigb, vary, beta,
  skipiter, pvalue)
```

Arguments

lowone	lower bound outer loop
highone	upper bound outer loop
lowtwo	lower bound middle loop
hightwo	upper bound middle loop
lowthree	lower bound inner loop
highthree	upper bound inner loop
n0a	outer loop index
n1a	middle loop index
n2a	inner loop index
n0	the major allele homozygotes
n1	the major allele heterozygotes
n2	the minor allele zygotes
pa	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
pb	parameter of the mixture distribution, a real number between zero and one with pa+pb=1
x	a zero mean explanatory variable from the SNP data set
mua	parameter of the mixture distribution, can be any real number
mub	parameter of the mixture distribution, can be any real number
sumsqx	sum of the squares of x
siga	parameter of the mixture distribution, can be any real number
sigb	parameter of the mixture distribution, can be any real number
vary	vary <- pa*(mua^2+siga^2)+pb*(mub^2+sigb^2)-(pa*mua+pb*mub)^2
beta	the beta from the regression being tested
skipiter	flag to determine if we can skip some calculations
pvalue	the input pvalue prior to calculating new improved pvalue

Value

pvalue

---

regresults *regresults: sample data*

Description

A sample dataset of input regression results based on machine-level accurate cumulative normal values. Rather than just typing in a few digits of the 2.5 the norminverse function in RATS was used to create sample-case betas which are exact

Format

csv format file with 4 variables (beta, n0, n1, n2) and 120 rows

Index