Package 'gJLS'

January 6, 2017

Type Package

Title Generalized Joint Location Scale (gJLS) Test

Version 0.1.0

Author David Soave <david.soave@mail.utoronto.ca>

Maintainer David Soave <david.soave@mail.utoronto.ca>

Description Joint Location Scale (JLS) test to simultaneously test for mean and variance differences between genotype groups.

License GPL-2

LazyData TRUE

RoxygenNote 5.0.1

Imports quantreg, nlme

URL http://github.com/dsoave/gJLS

BugReports http://github.com/dsoave/gJLS/issues

R topics documented:

`gJLS_test` *Generalized Joint Location Scale (gJLS) Test*

Description

This function performs the Joint Location Scale (JLS) test (Soave and Sun 2017, Soave et al. 2015) to simultaneously test for mean and variance differences between groups, allowing for correlated errors and group uncertainty. The gJLS test uses Fisher's combined p-value method to combine evidence from the individual, generalized locaiton (gL) and scale (gS) tests.

Usage

```
gJLS_test(model.loc, model.scale, data, correlation = NULL)
```

Arguments

`model.loc`	a two-sided linear formula object describing the location test model, with the response (y) on the left and a ~ operator separating the covariates of interest on the right, separated by + operators.
`model.scale`	a two-sided linear formula object describing the scale test model, with the response (y) on the left and a ~ operator separating the covariates of interest on the right, separated by + operators.
`data`	a data frame containing the variables named in model and correlation arguments. This is required.
`correlation`	an optional corStruct object describing the within-group correlation structure. The correlation structure must be called directly from the nlme pacakge using "nlme::" (see examples below). See the documentation of corClasses for a description of the available corStruct classes. If a grouping variable is to be used, it must be specified in the form argument to the corStruct constructor. Defaults to NULL, corresponding to uncorrelated errors.

Details

No missing data are allowed - function will return an "error". Absolute residuals, are estimated using least absolute deviation (LAD) regression. Outcome (phenotype) must be quantitative and covariate (genotype) may be discrete (categorical) or continuous.

Value

a table consisting of test statistics, degrees of freedom and p-vaules for each of the generalized location (gL), scale (gS) and, joint location-scale (gJLS) tests.

numDF the gS test statistic numerator degrees of freedom

denDF the gS test statistic denominator degrees of freedom

gS_p the gS test p-value

Author(s)

David Soave

References

Soave, D., Corvol, H., Panjwani, N., Gong, J., Li, W., Boelle, P.Y., Durie, P.R., Paterson, A.D., Rommens, J.M., Strug, L.J., and Sun, L. (2015). A Joint Location-Scale Test Improves Power to Detect Associated SNPs, Gene Sets, and Pathways. American journal of human genetics 97, 125-138.

Soave, D. and Sun, L. (2017). A Generalized Levene's Scale Test for Variance Heterogeneity in the Presence of Sample Correlation and Group Uncertainty. Biometrics (Accepted).

See Also

`gL_test`, `gS_test`

Examples

```
##############################################################################
## Example simulating data from model [i] (Soave et al. 2015 AJHG)
##############################################################################

n<-2000  ## total sample size
pA<-0.3  ## MAF
pE1<-0.3  ## frequency of exposure E1

## Genotypes (XG)
genocount<-rmultinom(1,size=n,prob=c(pA*pA, 2*pA*(1-pA), (1-pA)*(1-pA)))
XG<-c(rep(0, genocount[1]), rep(1, genocount[2]), rep(2,genocount[3]))
XG<-sample(XG,size=length(XG),replace=FALSE)

## Exposures (E1)
E1<-rbinom(n,1,prob=pE1)

## Phenotype (y)
y<-0.01*XG+0.3*E1+0.5*XG*E1+rnorm(n,0,1)

# Additive model (will work with dosages)
JLS_test(y,XG,XG)
# Genotypic model (will not work with dosages --> factor() will create many groups)
JLS_test(y,factor(XG),factor(XG))

X2=round(cbind(XG==1,XG==2)) #convert to 2 columns

# Genotypic model --> same result as results above using JLS_test(y,factor(X),factor(X))
# This is how genotype probabilities will be analyzed (using a 2 column design matrix)
JLS_test(y,X2,X2)
```

`gL_test` *Generalized Location (gL) Test*

Description

This function performs a generalized location (gL) test (Soave and Sun 2017) using the generalized least squares function, gls(), from the nlme package (Pinheiro and Bates 2000) to allow for correlated errors.

Usage

```
gL_test(model, data, correlation = NULL)
```

Arguments

`model`	a two-sided linear formula object describing the model, with the response (y) on the left and a ~ operator separating the covariates of interest on the right, separated by + operators.
`data`	a data frame containing the variables named in model and correlation arguments. This is required.

`correlation` an optional corStruct object describing the within-group correlation structure. The correlation structure must be called directly from the nlme pacakge using "nlme::" (see examples below). See the documentation of corClasses for a description of the available corStruct classes. If a grouping variable is to be used, it must be specified in the form argument to the corStruct constructor. Defaults to NULL, corresponding to uncorrelated errors.

Details

No missing data are allowed - function will return an "error". Outcome (phenotype) must be quantitative and covariate (genotype) may be discrete (categorical) or continuous.

Value

gL_F the gL test statistic

numDF the gL test statistic numerator degrees of freedom

denDF the gL test statistic denominator degrees of freedom

gL_p the gL test p-value

Author(s)

David Soave

References

Soave, D., Corvol, H., Panjwani, N., Gong, J., Li, W., Boelle, P.Y., Durie, P.R., Paterson, A.D., Rommens, J.M., Strug, L.J., and Sun, L. (2015). A Joint Location-Scale Test Improves Power to Detect Associated SNPs, Gene Sets, and Pathways. American journal of human genetics 97, 125-138.

Soave, D. and Sun, L. (2017). A Generalized Levene's Scale Test for Variance Heterogeneity in the Presence of Sample Correlation and Group Uncertainty. Biometrics (Accepted).

See Also

gS_test, gJLS_test

Examples

```
################################################################################
## Example simulating data from model [i] (Soave et al. 2015 AJHG)
################################################################################


n<-2000  ## total sample size
pA<-0.3  ## MAF
pE1<-0.3  ## frequency of exposure E1

## Genotypes (XG)
genocount<-rmultinom(1,size=n,prob=c(pA*pA, 2*pA*(1-pA), (1-pA)*(1-pA)))
XG<-c(rep(0, genocount[1]), rep(1, genocount[2]), rep(2,genocount[3]))
XG<-sample(XG,size=length(XG),replace=FALSE)

## Exposures (E1)
E1<-rbinom(n,1,prob=pE1)
```

```
## Phenotype (y)
y<-0.01*XG+0.3*E1+0.5*XG*E1+rnorm(n,0,1)

# Additive model (will work with dosages)
JLS_test(y,XG,XG)
# Genotypic model (will not work with dosages --> factor() will create many groups)
JLS_test(y,factor(XG),factor(XG))

X2=round(cbind(XG==1,XG==2)) #convert to 2 columns

# Genotypic model --> same result as results above using JLS_test(y,factor(X),factor(X))
# This is how genotype probabilities will be analyzed (using a 2 column design matrix)
JLS_test(y,X2,X2)
```

`gS_test` *Generalized Scale (gS) Test*

Description

This function performs the generalized scale (gS) test (Soave and Sun 2017). This extension of Levene's (1960) test allows for correlated errors and group uncertainty.

Usage

```
gS_test(model, data, correlation = NULL)
```

Arguments

`model`	a two-sided linear formula object describing the model, with the response (y) on the left and a ~ operator separating the covariates of interest on the right, separated by + operators.
`data`	a data frame containing the variables named in model and correlation arguments. This is required.
`correlation`	an optional corStruct object describing the within-group correlation structure. The correlation structure must be called directly from the nlme pacakge using "nlme::" (see examples below). See the documentation of corClasses for a description of the available corStruct classes. If a grouping variable is to be used, it must be specified in the form argument to the corStruct constructor. Defaults to NULL, corresponding to uncorrelated errors.

Details

No missing data are allowed - function will return an "error". Absolute residuals, are estimated using least absolute deviation (LAD) regression. Outcome (phenotype) must be quantitative and covariate (genotype) may be discrete (categorical) or continuous.

Value

gS_F the gS test statistic

numDF the gS test statistic numerator degrees of freedom

denDF the gS test statistic denominator degrees of freedom

gS_p the gS test p-value

Author(s)

David Soave

References

Soave, D., Corvol, H., Panjwani, N., Gong, J., Li, W., Boelle, P.Y., Durie, P.R., Paterson, A.D., Rommens, J.M., Strug, L.J., and Sun, L. (2015). A Joint Location-Scale Test Improves Power to Detect Associated SNPs, Gene Sets, and Pathways. American journal of human genetics 97, 125-138.

Soave, D. and Sun, L. (2017). A Generalized Levene's Scale Test for Variance Heterogeneity in the Presence of Sample Correlation and Group Uncertainty. Biometrics (Accepted).

See Also

gL_test, gJLS_test

Examples

```
##############################################################################
## Example simulating data from Web Table 5 (Soave and Sun 2017 Biometrics)
##############################################################################

#### Simulation parameters
n=100 # number of sibling pairs
pA=0.2 # minor allele frequency

r1=0.5  # within-pair correlation
a=0.7 # group uncertainty
s0=1;s1=1.5;s2=2  # within-group variances
m0=0;m1=0.3;m2=0.6  # group means


#### identical by decent (IBD) sharing probabilities
GIBD1=c(pA^4,(1-pA)^4,4*pA^2*(1-pA)^2,2*pA^3*(1-pA),2*pA^3*(1-pA),2*pA*(1-pA)^3,2*pA*(1-pA)^3,
       pA^2*(1-pA)^2,pA^2*(1-pA)^2)
GIBD2=c(pA^3,(1-pA)^3,pA*(1-pA),pA^2*(1-pA),pA^2*(1-pA),pA*(1-pA)^2,pA*(1-pA)^2,0,0)
GIBD3=c(pA^2,(1-pA)^2,2*pA*(1-pA),0,0,0,0,0,0)


### drawing the number of alleles shared IBD, D = 0, 1 or 2, from a multinomial distribution
### with parameters (0.25, 0.5, 0.25), independently for each sib-pair
IBD=rmultinom(1,size=n,prob=c(.25,.5,.25))

### Given the IBD status D, we then simulated paired genotypes (G1;G2), following
### the known conditional distribution of {(G1;G2)|D}
dfG=cbind(data.frame(rmultinom(1,size=IBD[1],prob=GIBD1)),data.frame(rmultinom(1,size=IBD[2],
       prob=GIBD2)), data.frame(rmultinom(1,size=IBD[3],prob=GIBD3)))
G1G2=apply(dfG,1,sum)
XG=c(rep(c(2,0,1,1,0,2,1,2,0),c(G1G2)),rep(c(2,0,1,0,1,1,2,0,0),c(G1G2)))
dfr=data.frame(XG,cbind(rep(1:n,2)))
names(dfr)[1:2]=c("XG","FID")
dfr$sub=1:(2*n)

dfr=dfr[order(dfr$XG),]
dfr=dfr[order(dfr$FID),]
```

```
### Generate paired outcome data from a bivariate normal distribution, BVN(0,1,r1),
### where r1 is the the within sib-pair correlation.
blockC=matrix(c(1,r1,r1,1), 2)
yy=cbind(MASS::mvrnorm(n,rep(0,dim(blockC)[1]),blockC))
yy=data.frame(c(yy[,1],yy[,2]))
yy$FID=rep(1:n,2)
yy1=yy[order(yy$FID),]
dfr$y=c(yy1[,1])
#dfr$y=dfr$y*sqrt(s0*(dfr$XG==0)+s1*(dfr$XG==1)+s2*(dfr$XG==2))

### induced scale differences and mean differences bewteen the TRUE genotype groups
dfr$y=dfr$y*sqrt(s0*(dfr$XG==0)+s1*(dfr$XG==1)+s2*(dfr$XG==2))+(m0*(dfr$XG==0)+
       m1*(dfr$XG==1)+m2*(dfr$XG==2))

### Convert the genotypes to pair of indicator variables for G=1 and G=2 minor alleles.
dfr$X1=with(dfr,(XG==1))+0
dfr$X2=with(dfr,(XG==2))+0


##############################################################################
## Analysis of true genotypes
##############################################################################

### Generalized scale, location and joint location-scale tests, using a compound
### symmetric correlation structure to account for within sib-pair correlation.
gS_test(model=y~X1+X2,data=dfr,correlation=nlme::corCompSymm(form=~1|FID))
gL_test(model=y~X1+X2,data=dfr,correlation=nlme::corCompSymm(form=~1|FID))
gJLS_test(model.loc=y~X1+X2,model.scale=y~X1+X2,data=dfr,correlation=nlme::corCompSymm(form=~1|FID))


### Generalized scale, location and joint location-scale tests, ignoring correlation
### structure (incorrect analysis)
gS_test(model=y~X1+X2,data=dfr)
gL_test(model=y~X1+X2,data=dfr)
gJLS_test(model.loc=y~X1+X2,model.scale=y~X1+X2,data=dfr)



### Convert the simulated true genotypes (XG) to probabilistic data (p) using a Dirichlet distribution
### with scale parameters a for the correct genotype category and (1-a)/2 for the other two
#install and load MCMCpack packageto use the rdirichlet function
#install.packages('MCMCpack')
#library(MCMCpack)

dfr=dfr[order(dfr$XG),]
XG<-dfr$XG
p=rbind(rdirichlet(length(XG[XG==0]),c(a,(1-a)/2,(1-a)/2)), rdirichlet(length(XG[XG==1]),
       c((1-a)/2,a,(1-a)/2)),rdirichlet(length(XG[XG==2]),c((1-a)/2,(1-a)/2,a))  )
dfr$p1=p[,2]
dfr$p2=p[,3]
dfr$dosage=dfr$p1+2*dfr$p2

### Best guess genotypes based on probabilistic data
B_G=function(x){return(which(x==max(x)))}
dfr$bgX=apply(p,1, B_G)-1
```

```
##############################################################################
## Analysis of genotype probabilities (reflecting group uncertainity)
##############################################################################

### Generalized scale, location and joint location-scale tests, using a compound
### symmetric correlation structure to account for within sib-pair correlation.
gS_test(model=y~p1+p2,data=dfr,correlation=nlme::corCompSymm(form=~1|FID))
gL_test(model=y~p1+p2,data=dfr,correlation=nlme::corCompSymm(form=~1|FID))
gJLS_test(model.loc=y~p1+p2,model.scale=y~p1+p2,data=dfr,correlation=nlme::corCompSymm(form=~1|FID))
```

Index