

# Package ‘aGE’

December 3, 2018

**Type** Package

**Title** Adaptive Set-Based Gene-Environment Interaction Test for Rare Variants

**Version** 0.0.9

**Description** Tests gene-environment interaction for rare genetic variants within the framework of adaptive sum of powered score test. The package includes two tests: adaptive gene-by-environment interaction test, and joint test for genetic main effects and gene-environment interaction. See Yang et al (2018) <doi:10.1002/sim.8037>.

**Depends** R (>= 3.4)

**License** GPL-3

**Encoding** UTF-8

**Imports** nlme, MASS, survey, stats

**NeedsCompilation** no

**Date** 2018-11-11

**RoxygenNote** 6.1.0

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**LazyData** true

**Author** Tianzhong Yang [aut, cre],  
Han Chen [aut],  
Peng Wei [aut]

**Maintainer** Tianzhong Yang <tianzhong.yang@uth.tmc.edu>

**Repository** CRAN

**Date/Publication** 2018-12-03 19:50:13 UTC

## R topics documented:

aGE . . . . .	2
aGE.joint . . . . .	3
<b>Index</b>	<b>4</b>

---

aGE *aGE interaction test*

---

### Description

aGE interaction test

### Usage

```
aGE(Y, G, cov = NULL, model = c("gaussian", "binomial"),
    pow = c(1:6), n.perm = 1000, method = "Simulation", nonparaE = F,
    DF = 10, stepwise = T)
```

### Arguments

Y	a numeric vector of phenotype values
G	a matrix for all RVs in the test gene or genomic region. The order of rows must match the order of Y. Missing is imputed as 0.
cov	a matrix with first column as the environmental variable to be tested. The order of rows must match the order of Y.
model	"binomial" for binary traits or "gaussian" for quantitative traits.
pow	Gamma set used to build a family of tests, default=c(1:6) for rare variants
n.perm	number of simulation to calculate the p-values, default=1000. Can increase to higher value depending on the significance level.
method	only have one option: "Simulation", also called Monte Carlo Method.
nonparaE	"T": use cubic splines for the environmental variable to fit the model; "F": use a linear function of the environmental variable to fit the model
DF	degree of freedom to use in the cubic splines, default=10. This option only works when nonparaE is set to "T"
stepwise	an option to speed up the simulation procedure for large n.perm number in real-data application. Up to \$n.perm=10^8\$

### Value

p-values

### Examples

```
{
  set.seed(12345)
  phenotype <- c(rep(1,50),rep(0,50))
  genotype <- data.frame(g1=sample(c(rep(1,10),rep(0,90))),g2=sample(c(rep(1,5), rep(0,95))))
  covariates <- data.frame(Envir=rnorm(100), Age=rnorm(100,60,5))
  exD <- list(Y=phenotype, G=genotype, X=covariates)
  aGE(Y=exD$Y, G=exD$G, cov=exD$X, model='binomial', nonparaE=FALSE, stepwise=FALSE)
}
```

aGE.joint

*aGE joint test***Description**

aGE joint test

**Usage**

```
aGE.joint(Y, G, cov = NULL, model = c("gaussian", "binomial"),
  pow = c(1:6), n.perm = 1000, method = c("Simulation"),
  nonparaE = F, DF = 10)
```

**Arguments**

Y	a numeric vector of phenotype values
G	a matrix or data frame for all RVs in the test gene or genomic region. The order of rows must match the order of Y. Missing is imputed as 0.
cov	a matrix or data frame with first column as the environmental variable to be tested. The order of rows must match the order of Y.
model	"binomial" for binary traits or "gaussian" for quantitative traits.
pow	Gamma set used to build a family of tests, default=c(1:6) for rare variants
n.perm	number of simulation to calculate the p-values, default=1000. Can increase to higher value depending on the significance level.
method	'Simulation': Monte Carlo Method
nonparaE	"T": use cubic splines for the environmental variable to fit the model; "F": use a linear function of the environmental variable to fit the model
DF	degree of freedom to use in the cubic splines, default=10. This option only works when nonparaE is set to "T".

**Value**

p-values

**Examples**

```
{
  set.seed(12345)
  phenotype <- c(rep(1,50),rep(0,50))
  genotype <- data.frame(g1=sample(c(rep(1,10),rep(0,90))),g2=sample(c(rep(1,5), rep(0,95))))
  covariates <- data.frame(Envir=rnorm(100), Age=rnorm(100,60,5))
  exD <- list(Y=phenotype, G=genotype, X=covariates)
  aGE.joint(Y=exD$Y, G=exD$G, cov=exD$X, model='binomial')
}
```

# Index

aGE, [2](#)  
aGE.joint, [3](#)