

# Package ‘SMUT’

July 21, 2025

**Type** Package

**Title** Multi-SNP Mediation Intersection-Union Test

**Version** 1.1

**Author** Wujuan Zhong

**Maintainer** Wujuan Zhong <zhongwujuan@gmail.com>

**Description** Testing the mediation effect of multiple SNPs on an outcome through a mediator.

**LazyData** true

**License** GPL (>= 2)

**Depends** R (>= 2.10)

**Imports** Rcpp (>= 0.12.14), SKAT, MASS

**LinkingTo** Rcpp, RcppEigen

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2019-09-24 04:40:08 UTC

## Contents

eigenMapMatMult . . . . .	2
Generalized_Testing_coefficient_of_mediator . . . . .	2
Genotype_data . . . . .	3
GSMUT . . . . .	4
SMUT . . . . .	6
Testing_coefficient_of_mediator . . . . .	8

<b>Index</b>	<b>10</b>
--------------	-----------

---

eigenMapMatMult      *Matrix multiplication using RcppEigen*

---

**Description**

Matrix multiplication using RcppEigen.

**Usage**

```
eigenMapMatMult(A,B)
```

**Arguments**

A, B                  numeric (double) complex matrices or vectors.

**Value**

The matrix product. The value is the same as A %\*% B

**Examples**

```
library(SMUT)
A=matrix(1:9,3,3)
A=A+0
B=as.matrix(c(5.0, 2.0, 0.0))
eigenMapMatMult(A,B)
# the result is the same as A %*% B

# Thanks for using our R package SMUT
```

---

Generalized\_Testing\_coefficient\_of\_mediator

*Testing coefficient of mediator in the outcome model for an outcome following an exponential family distribution or a survival outcome*

---

**Description**

Testing coefficient of mediator, namely theta, in the outcome model. The outcome model is the following.

$$\text{outcome} \sim \text{intercept} + G \cdot \text{gamma} + \text{mediator} \cdot \text{theta} + \text{error}$$
**Usage**

```
Generalized_Testing_coefficient_of_mediator(G,mediator,outcome,
covariates=NULL,outcome_type,
approx=TRUE,verbose=FALSE)
```

**Arguments**

G	n by p matrix (n rows and p columns). Each row is one individual; each column is one SNP.
mediator	a vector length of n. It is the mediator variable.
outcome	a vector length of n. It is the outcome variable.
covariates	n by r matrix (n rows and r columns). Each row is one individual; each column is one covariate.
outcome_type	Type of the outcome variable. "continuous" for a continuous outcome; "binary" for a binary outcome; "count" for a count outcome; "survival" for a survival outcome.
approx	a boolean value. This is an indicator whether the approximation of computing derivatives is applied to save computing time. Default is TRUE.
verbose	a boolean value. If TRUE a lot of computing details is printed. Default is FALSE.

**Value**

p_value	P value for testing the coefficient of mediator in the outcome model.
theta_hat	The point estimate of theta (coefficient of mediator) in the outcome model.

**Author(s)**

Wujuan Zhong

---

Genotype\_data

*Example genotype data for SMUT*

---

**Description**

Example genotype data for SMUT. It is a matrix with 100 rows and 200 columns. Each row is an individual; each column is a SNP.

**Format**

It is a matrix with 100 rows and 200 columns. Each row is an individual; each column is a SNP.

GSMUT

*Generalized Multi-SNP Mediation Intersection-Union Test***Description**

Testing the mediation effect of multiple SNPs on an outcome following an exponential family distribution or a survival outcome through a continuous mediator.

**Usage**

```
GSMUT(G,mediator,outcome,covariates=NULL,outcome_type,
approx=TRUE,verbose=FALSE)
```

**Arguments**

G	n by p matrix (n rows and p columns). Each row is one individual; each column is one SNP.
mediator	a vector length of n. It is the mediator variable.
outcome	a vector length of n. It is the outcome variable.
covariates	n by r matrix (n rows and r columns). Each row is one individual; each column is one covariate.
outcome_type	Type of the outcome variable. "continuous" for a continuous outcome; "binary" for a binary outcome; "count" for a count outcome; "survival" for a survival outcome.
approx	a boolean value. This is an indicator whether the approximation of computing derivatives is applied to save computing time. Default is TRUE.
verbose	a boolean value. If TRUE a lot of computing details is printed. Default is FALSE.

**Value**

p_value_IUT	The p value for testing the mediation effect ( $\theta \cdot \beta$ ) based on intersection-union test.
p_value_theta	The p value for testing $\theta$ in the outcome model. The outcome model is the following. $\text{outcome} \sim \text{intercept} + \text{covariates} \cdot \text{iota} + G \cdot \text{gamma} + \text{mediator} \cdot \text{theta}$
theta_hat	The point estimate of $\theta$ (coefficient of mediator) in the outcome model.
p_value_beta	The p value for testing $\beta$ in the mediator model. The mediator model is the following. $\text{mediator} \sim \text{intercept} + \text{covariates} \cdot \text{iota} + G \cdot \text{beta} + \text{error}$

**Author(s)**

Wujuan Zhong

**Examples**

```

library(SMUT)
# load the Genotype data included in this R package
data("Genotype_data")

##### for a binary outcome #####
set.seed(1)

# generate two covariates
covariate_1=rnorm(nrow(Genotype_data),0,1)
covariate_2=sample(c(0,1),size=nrow(Genotype_data),replace = TRUE)
covariates=cbind(covariate_1,covariate_2)

# generate a mediator
beta=rnorm(ncol(Genotype_data),0,0.5)
tau_M=c(-0.3,0.2)
e1 = rnorm(nrow(Genotype_data), 0, 1)
mediator = 1 + eigenMapMatMult(Genotype_data,beta) +
            eigenMapMatMult(covariates, tau_M) + e1

#### generate a binary outcome ####
theta=1
gamma=rnorm(ncol(Genotype_data),0,0.5)
tau=c(-0.2,0.2)
eta=1 + eigenMapMatMult(Genotype_data, gamma) +
        eigenMapMatMult(covariates, tau) + theta * mediator
pi=1/(1+exp( -(eta) ))
outcome=rbinom(length(pi),size=1,prob=pi)
result=GSMUT(G=Genotype_data,mediator=mediator,outcome=outcome,
covariates=covariates,outcome_type="binary")
print(result)
# p_value_IUT is the p value for the mediation effect.

## Not run:

##### generate a count outcome #####
theta=1
gamma=rnorm(ncol(Genotype_data),0,0.5)
tau=c(-0.2,0.2)
eta=1 + eigenMapMatMult(Genotype_data, gamma) +
        eigenMapMatMult(covariates, tau) + theta * mediator
mu_param=exp(eta) # the mean parameter
phi_param=10 # the shape parameter
outcome=rnbinom(length(mu_param),size=phi_param,mu=mu_param)
result=GSMUT(G=Genotype_data,mediator=mediator,outcome=outcome,
covariates=covariates,outcome_type="count")
print(result)
# p_value_IUT is the p value for the mediation effect.

##### generate a survival outcome #####
theta=2

```

```

gamma=rnorm(ncol(Genotype_data),0,0.5)
tau=c(-0.2,0.2)
eta=1 + eigenMapMatMult(Genotype_data, gamma) +
      eigenMapMatMult(covariates, tau) + theta * mediator
v=runif(nrow(Genotype_data))
lambda=0.01; rho=1; rateC=0.001
Tlat=(- log(v) / (lambda * exp( eta )))^(1 / rho)
# censoring times
C= rexp(nrow(Genotype_data), rate=rateC)
# follow-up times and event indicators
time= pmin(Tlat, C)
status= as.numeric(Tlat <= C)
outcome=cbind(time,status)
colnames(outcome)=c("time","status")
result=GSMUT(G=Genotype_data,mediator=mediator,outcome=outcome,
covariates=covariates,outcome_type="survival")
print(result)
# p_value_IUT is the p value for the mediation effect.

## End(Not run)

```

---

SMUT

---

*Multi-SNP Mediation Intersection-Union Test*


---

## Description

Testing the mediation effect of multiple SNPs on an outcome through a mediator.

## Usage

```
SMUT(G, mediator, outcome,
outcome_type="continuous", method="score", approxi=TRUE, debug=FALSE)
```

## Arguments

G	n by p matrix (n rows and p columns). Each row is one individual; each column is one SNP.
mediator	a vector length of n. It is the mediator variable.
outcome	a vector length of n. It is the outcome variable.
outcome_type	Type of the outcome variable. For now, this package only deals with continuous outcome. Default is "continuous".
method	The method of testing coefficient of mediator in the outcome model. The score test is used. Default is "score".
approxi	a boolean value. This is an indicator whether the approximation of the score statistic is applied to save computing time. Default is TRUE.

debug a boolean value. If TRUE a lot of computing details is printed; otherwise the function is completely silent. Default is FALSE.

### Value

p\_value\_IUT The p value for testing the mediation effect ( $\theta \cdot \beta$ ) based on intersection-union test.

p\_value\_theta The p value for testing  $\theta$  in the outcome model. The outcome model is the following.  

$$\text{outcome} \sim \text{intercept} + G \cdot \gamma + \text{mediator} \cdot \theta + \text{error}.$$

p\_value\_beta The p value for testing  $\beta$  in the mediator model. The mediator model is the following.  

$$\text{mediator} \sim \text{intercept} + G \cdot \beta + \text{error}$$

### Author(s)

Wujuan Zhong

### References

Zhong, W., Spracklen, C. N., Mohlke, K. L., Zheng, X., Fine, J., & Li, Y. (2019). Multi-SNP mediation intersection-union test. *Bioinformatics*.

### Examples

```
library(SMUT)
# load the Genotype data included in this R package
data("Genotype_data")

# generate one mediator and one outcome

# first example, the mediation effect is significant
set.seed(1)
beta=rnorm(ncol(Genotype_data),1,2)
e1 = rnorm(nrow(Genotype_data), 0, 1)
mediator = 1 + eigenMapMatMult(Genotype_data,beta) + e1

theta=0.8
gamma=rnorm(ncol(Genotype_data),0.5,2)
e2 = rnorm(nrow(Genotype_data), 0, 1)
outcome = 2 + eigenMapMatMult(Genotype_data,gamma) + theta*mediator + e2

p_value=SMUT(G=Genotype_data,mediator=mediator,outcome=outcome)
print(p_value)

# p_value_IUT is the p value for the mediation effect.
# we have significant(at alpha level 0.05) mediation effects (p_value_IUT = 0.001655787).

# second example, the mediation effect is non-significant
set.seed(1)
beta=rnorm(ncol(Genotype_data),1,2)
```

```

e1 = rnorm(nrow(Genotype_data), 0, 1)
mediator = 1 + eigenMapMatMult(Genotype_data,beta) + e1

theta=0
gamma=rnorm(ncol(Genotype_data),0.5,2)
e2 = rnorm(nrow(Genotype_data), 0, 1)
outcome = 2 + eigenMapMatMult(Genotype_data,gamma) + theta*mediator + e2

p_value=SMUT(G=Genotype_data,mediator=mediator,outcome=outcome)
print(p_value)

# p_value_IUT is the p value for the mediation effect.
# we have non-significant(at alpha level 0.05) mediation effects (p_value_IUT = 0.3281677).

# third example, the mediation effect is non-significant
set.seed(1)
beta=rep(0,ncol(Genotype_data))
e1 = rnorm(nrow(Genotype_data), 0, 1)
mediator = 1 + eigenMapMatMult(Genotype_data,beta) + e1

theta=0.8
gamma=rnorm(ncol(Genotype_data),0.5,2)
e2 = rnorm(nrow(Genotype_data), 0, 1)
outcome = 2 + eigenMapMatMult(Genotype_data,gamma) + theta*mediator + e2

p_value=SMUT(G=Genotype_data,mediator=mediator,outcome=outcome)
print(p_value)

# p_value_IUT is the p value for the mediation effect.
# we have non-significant(at alpha level 0.05) mediation effects (p_value_IUT = 0.5596977).

# Thanks for using our R package SMUT

```

---

Testing\_coefficient\_of\_mediator

*Testing coefficient of mediator in the outcome model*

---

## Description

Testing coefficient of mediator, namely theta, in the outcome model. The outcome model is the following.

outcome ~ intercept + G\*gamma + mediator\*theta + error

## Usage

```

Testing_coefficient_of_mediator(G, mediator, outcome,
outcome_type="continuous", method="score", approxi=TRUE, debug=FALSE)

```

**Arguments**

G	n by p matrix (n rows and p columns). Each row is one individual; each column is one SNP.
mediator	a vector length of n. It is the mediator variable.
outcome	a vector length of n. It is the outcome variable.
outcome_type	Type of the outcome variable. For now, this package only deals with continuous outcome. Default is "continuous".
method	The method of testing coefficient of mediator in the outcome model. The score test is used. Default is "score".
approx	a boolean value. This is an indicator whether the approximation of the score statistic is applied to save computing time. Default is TRUE.
debug	a boolean value. If TRUE a lot of computing details is printed; otherwise the function is completely silent. Default is FALSE.

**Value**

P value for testing the coefficient of mediator in the outcome model.

**Author(s)**

Wujuan Zhong

**Examples**

```
library(SMUT)
# load the Genotype data included in this R package
data("Genotype_data")

# generate one mediator and one outcome

set.seed(1)
beta=rnorm(ncol(Genotype_data),1,2)
e1 = rnorm(nrow(Genotype_data), 0, 1)
mediator = 1 + eigenMapMatMult(Genotype_data,beta) + e1

theta=0.8
gamma=rnorm(ncol(Genotype_data),0.5,2)
e2 = rnorm(nrow(Genotype_data), 0, 1)
outcome = 2 + eigenMapMatMult(Genotype_data,gamma) + theta*mediator + e2

p_value=Testing_coefficient_of_mediator(G=Genotype_data,mediator=mediator,outcome=outcome)
print(p_value)

# Thanks for using our R package SMUT
```

# Index

eigenMapMatMult, [2](#)

Generalized\_Testing\_coefficient\_of\_mediator,  
[2](#)

Genotype\_data, [3](#)

GSMUT, [4](#)

SMUT, [6](#)

Testing\_coefficient\_of\_mediator, [8](#)