Package 'GxEprs'

November 21, 2023

Title Genotype-by-Environment Interaction in Polygenic Score Models

Version 1.2

Description A novel PRS model is introduced to enhance the prediction accuracy by utilising GxE effects. This package performs Genome Wide Association Studies (GWAS) and Genome Wide Environment Interaction Studies (GWEIS) using a discovery dataset. The package has the ability to obtain polygenic risk scores (PRSs) for a target sample. Finally it predicts the risk values of each individual in the target sample. Users have the choice of using existing models (Li et al., 2015) <doi:10.1093/annonc/mdu565>, (Pandis et al., 2013) <doi:10.1093/ejo/cjt054>, (Peyrot et al., 2018) <doi:10.1016/j.biopsych.2017.09.009> and (Song et al., 2022) <doi:10.1038/s41467-022-32407-9>, as well as newly proposed models for genomic risk prediction (refer to the URL for more details).

URL https://github.com/DoviniJ/GxEprs

License GPL (>=3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Depends R (>= 2.10)

LazyData true

R topics documented:

`Bcov_discovery` *Covariate data file of the discovery dataset when the outcome is binary. This contains covariate information of the individuals in the discovery dataset following confounders.*

Description

Covariate data file of the discovery dataset when the outcome is binary. This contains covariate information of the individuals in the discovery dataset following confounders.

Usage

```
Bcov_discovery
```

Format

A dataframe with 800 rows and 18 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Standardized covariate

Column 4 Square of the standardized covariate

Column 5 Confounder 1

Column 6 Confounder 2

Column 7 Confounder 3

Column 8 Confounder 4

Column 9 Confounder 5

Column 10 Confounder 6

Column 11 Confounder 7

Column 12 Confounder 8

Column 13 Confounder 9

Column 14 Confounder 10

Column 15 Confounder 11

Column 16 Confounder 12

Column 17 Confounder 13

Column 18 Confounder 14

`Bcov_target`	*Covariate data file of the target dataset when the outcome is binary. This contains covariate information of the individuals in the target dataset following confounders.*

Description

Covariate data file of the target dataset when the outcome is binary. This contains covariate information of the individuals in the target dataset following confounders.

Usage

```
Bcov_target
```

Format

A dataframe with 200 rows and 18 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Standardized covariate

Column 4 Square of the standardized covariate

Column 5 Confounder 1

Column 6 Confounder 2

Column 7 Confounder 3

Column 8 Confounder 4

Column 9 Confounder 5

Column 10 Confounder 6

Column 11 Confounder 7

Column 12 Confounder 8

Column 13 Confounder 9

Column 14 Confounder 10

Column 15 Confounder 11

Column 16 Confounder 12

Column 17 Confounder 13

Column 18 Confounder 14

`Bphe_discovery`	*Phenotype data file of the discovery dataset when the outcome is binary. This contains phenotype information of the individuals in the discovery dataset.*

Description

Phenotype data file of the discovery dataset when the outcome is binary. This contains phenotype information of the individuals in the discovery dataset.

Usage

```
Bphe_discovery
```

Format

A dataframe with 800 rows and 3 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Phenotype (1=controls, 2=cases)

`Bphe_target`	*Phenotype data file of the target dataset when the outcome is binary. This contains phenotype information of the individuals in the target dataset.*

Description

Phenotype data file of the target dataset when the outcome is binary. This contains phenotype information of the individuals in the target dataset.

Usage

```
Bphe_target
```

Format

A dataframe with 200 rows and 3 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Phenotype (0=controls, 1=cases)

`DummyData.bim` *PLINK .bim file*

Description

PLINK .bim file

Usage

```
DummyData.bim
```

Format

This follows PLINK general format

Column 1 Chromosome ID

Column 2 SNP ID

Column 3 Position of centimorgans

Column 4 Base-pair coordinate

Column 5 Minor Allele

Column 6 Reference Allele

`DummyData.fam` *PLINK .fam file*

Description

PLINK .fam file

Usage

```
DummyData.fam
```

Format

This follows PLINK general format

Column 1 Family ID

Column 2 Individual ID

Column 3 Father's ID

Column 4 Mother's ID

Column 5 Sex

Column 6 Phenotype value

DummyData.map *PLINK .map file*

Description

PLINK .map file

Usage

```
DummyData.map
```

Format

This follows PLINK general format

DummyData.ped *PLINK .ped file*

Description

PLINK .ped file

Usage

```
DummyData.ped
```

Format

This follows PLINK general format

GWAS_binary *GWAS_binary function This function performs GWAS using plink2 and outputs the GWAS summary statistics with additive SNP effects. Users may save the output in a user-specified file (see example).*

Description

GWAS_binary function This function performs GWAS using plink2 and outputs the GWAS summary statistics with additive SNP effects. Users may save the output in a user-specified file (see example).

Usage

```
GWAS_binary(plink_path, b_file, Bphe_discovery, Bcov_discovery, thread = 20)
```

Arguments

plink_path	Path to the PLINK executable application
b_file	Prefix of the binary files, where all .fam, .bed and .bim files have a common prefix
Bphe_discovery	Name (with file extension) of the phenotype file containing family ID, individual ID and phenotype of the discovery dataset as columns, without heading
Bcov_discovery	Name (with file extension) of the covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the discovery dataset as columns, without heading
thread	Number of threads used

Value

This function will perform GWAS and output

B_out.trd.sum
: GWAS summary statistics with additive SNP effects

Examples

```
## Not run:
x <- GWAS_binary(plink_path, DummyData, Bphe_discovery, Bcov_discovery,
thread = 20)
sink("B_out.trd.sum") #to create a file in the working directory
write.table(x[c("ID", "A1", "OR")], sep = " ",
row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
head(x) #to obtain the head of GWAS summary statistics of additive SNP effects
x$CHROM #to extract the chromosome number
x$POS #to extract the base pair position
x$ID #to extract the SNP ID
x$REF #to extract the reference allele
x$ALT #to extract the alternate allele
x$A1 #to extract the minor allele
x$OBS_CT #to extract the number of allele observations
x$OR #to extract the odds ratios of the SNP effects
x$LOG_OR_SE #to extract the standard errors of log odds
x$Z_STAT #to extract the test statistics
x$P #to extract the p values

## End(Not run)
```

GWAS_quantitative *GWAS_quantitative function This function performs GWAS using plink2 and outputs the GWAS summary statistics with additive SNP effects. Users may save the output in a user-specified file (see example).*

Description

GWAS_quantitative function This function performs GWAS using plink2 and outputs the GWAS summary statistics with additive SNP effects. Users may save the output in a user-specified file (see example).

Usage

```
GWAS_quantitative(
  plink_path,
  b_file,
  Qphe_discovery,
  Qcov_discovery,
  thread = 20
)
```

Arguments

plink_path	Path to the PLINK executable application
b_file	Prefix of the binary files, where all .fam, .bed and .bim files have a common prefix
Qphe_discovery	Name (with file extension) of the phenotype file containing family ID, individual ID and phenotype of the discovery dataset as columns, without heading
Qcov_discovery	Name (with file extension) of the covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the discovery dataset as columns, without heading
thread	Number of threads used

Value

This function will perform GWAS and output

Q_out.trd.sum	GWAS summary statistics with additive SNP effects

Examples

```
## Not run:
x <- GWAS_quantitative(plink_path, DummyData, Qphe_discovery, Qcov_discovery,
thread = 20)
sink("Q_out.trd.sum") #to create a file in the working directory
write.table(x[c("ID", "A1", "BETA")], sep = " ",
row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
head(x) #to obtain the head of GWAS summary statistics of additive SNP effects
x$CHROM #to extract the chromosome number
x$POS #to extract the base pair position
x$ID #to extract the SNP ID
x$REF #to extract the reference allele
x$ALT #to extract the alternate allele
x$A1 #to extract the minor allele
x$OBS_CT #to extract the number of allele observations
x$BETA #to extract the SNP effects
```

```
x$SE #to extract the standard errors of the SNP effects
x$T_STAT #to extract the test statistics
x$P #to extract the p values

## End(Not run)
```

`GWEIS_binary` *GWEIS_binary function This function performs GWEIS using plink2 and outputs the GWEIS summary statistics with additive SNP effects and interaction SNP effects. Users may save the outputs in separate user-specified files (see examples).*

Description

GWEIS_binary function This function performs GWEIS using plink2 and outputs the GWEIS summary statistics with additive SNP effects and interaction SNP effects. Users may save the outputs in separate user-specified files (see examples).

Usage

```
GWEIS_binary(plink_path, b_file, Bphe_discovery, Bcov_discovery, thread = 20)
```

Arguments

`plink_path`	Path to the PLINK executable application
`b_file`	Prefix of the binary files, where all .fam, .bed and .bim files have a common prefix
`Bphe_discovery`	Phenotype file containing family ID, individual ID and phenotype of the discovery dataset as columns, without heading
`Bcov_discovery`	Covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the discovery dataset as columns, without heading
`thread`	Number of threads used

Value

This function will perform GWEIS and output

`B_out.sum`	GWEIS summary statistics with additive and interaction SNP effects

Examples

```
## Not run:
x <- GWEIS_binary(plink_path, DummyData, Bphe_discovery, Bcov_discovery,
thread = 20)
sink("B_out.add.sum") #to create a file in the working directory
write.table(x[c("ID", "A1", "ADD_OR")], sep = " ",
row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
sink("B_out.gxe.sum") #to create a file in the working directory
```

```
write.table(x[c("ID", "A1", "INTERACTION_OR")], sep = " ",
row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
head(x) #to extract the head of all columns in GWEIS summary
#statistics of additive and interaction SNP effects
x$CHROM #to extract the chromosome number
x$POS #to extract the base pair position
x$ID #to extract the SNP ID
x$REF #to extract the reference allele
x$ALT #to extract the alternate allele
x$A1 #to extract the minor allele
x$OBS_CT #to extract the number of allele observations
x$ADD_OR #to extract the odds ratios of additive SNP effects
x$ADD_LOG_OR_SE #to extract the standard errors of log
#odds of additive SNP effects
x$ADD_Z_STAT #to extract the test statistics of additive
#SNP effects
x$ADD_P #to extract the p values of additive SNP effects
x$INTERACTION_OR #to extract the odds ratios of
#interaction SNP effects
x$INTERACTION_LOG_OR_SE #to extract the standard errors of
#log odds of interaction SNP effects
x$INTERACTION_Z_STAT #to extract the test statistics of
#interaction SNP effects
x$INTERACTION_P #to extract the p values of interaction
#SNP effects

## End(Not run)
```

`GWEIS_quantitative` *GWEIS_quantitative function This function performs GWEIS using plink2 and outputs the GWEIS summary statistics with additive SNP effects and interaction SNP effects separately. It is recommended to save the outputs in separate user-specified files (see examples).*

Description

GWEIS_quantitative function This function performs GWEIS using plink2 and outputs the GWEIS summary statistics with additive SNP effects and interaction SNP effects separately. It is recommended to save the outputs in separate user-specified files (see examples).

Usage

```
GWEIS_quantitative(
  plink_path,
  b_file,
  Qphe_discovery,
  Qcov_discovery,
  thread = 20
)
```

Arguments

plink_path	Path to the PLINK executable application
b_file	Prefix of the binary files, where all .fam, .bed and .bim files have a common prefix
Qphe_discovery	Phenotype file containing family ID, individual ID and phenotype of the discovery dataset as columns, without heading
Qcov_discovery	Covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the discovery dataset as columns, without heading
thread	Number of threads used

Value

This function will perform GWEIS and output

Q_out.sum	GWEIS summary statistics with additive and interaction SNP effects

Examples

```
## Not run:
x <- GWEIS_quantitative (plink_path, DummyData, Qphe_discovery, Qcov_discovery,
thread = 20)
sink("Q_out.add.sum") #to create a file in the working directory
write.table(x[c("ID", "A1", "ADD_BETA")], sep = " ",
row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
sink("Q_out.gxe.sum") #to create a file in the working directory
write.table(x[c("ID", "A1", "INTERACTION_BETA")], sep = " ",
row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
head(x) #to extract the head of all columns in GWEIS summary
#statistics of additive and interaction SNP effects
x$CHROM #to extract the chromosome number
x$POS #to extract the base pair position
x$ID #to extract the SNP ID
x$REF #to extract the reference allele
x$ALT #to extract the alternate allele
x$A1 #to extract the minor allele
x$OBS_CT #to extract the number of allele observations
x$ADD_BETA #to extract the additive SNP effects
x$ADD_SE #to extract the standard errors of the
#additive SNP effects
x$ADD_T_STAT #to extract the test statistics of additive
#SNP effects
x$ADD_P #to extract the p values of additive SNP effects
x$INTERACTION_BETA #to extract the interaction SNP effects
x$INTERACTION_SE #to extract the standard errors of the
#interaction SNP effects
x$INTERACTION_T_STAT #to extract the test statistics of
#interaction SNP effects
x$INTERACTION_P #to extract the p values of interaction
#SNP effects
```

```
## End(Not run)
```

`PRS_binary` *PRS_binary function This function uses plink2 and outputs Polygenic Risk Scores (PRSs) of all the individuals, using pre-generated GWAS and/or GWEIS summary statistics. Note that the input used in this function can be generated by using GWAS_binary and/or GWEIS_binary functions. Users may save the output in a user-specified file (see examples).*

Description

PRS_binary function This function uses plink2 and outputs Polygenic Risk Scores (PRSs) of all the individuals, using pre-generated GWAS and/or GWEIS summary statistics. Note that the input used in this function can be generated by using GWAS_binary and/or GWEIS_binary functions. Users may save the output in a user-specified file (see examples).

Usage

```
PRS_binary(plink_path, b_file, summary_input)
```

Arguments

`plink_path`	Path to the PLINK executable application
`b_file`	Prefix of the binary files, where all .fam, .bed and .bim files have a common prefix
`summary_input`	Pre-generated GWAS and/or GWEIS summary statistics

Value

This function will output

`prs.sscore`	PRSs for each individual

Examples

```
## Not run:
a <- GWAS_binary(plink_path, DummyData, Bphe_discovery, Bcov_discovery)
trd <- a[c("ID", "A1", "OR")]
b <- GWEIS_binary(plink_path, DummyData, Bphe_discovery, Bcov_discovery)
add <- b[c("ID", "A1", "ADD_OR")]
gxe <- b[c("ID", "A1", "INTERACTION_OR")]
x <- PRS_binary(plink_path, DummyData, summary_input = trd)
sink("B_trd.sscore") #to create a file in the working directory
write.table(x, sep = " ", row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
head(x) #to read the head of all columns in the output
x$FID #to extract the family ID's of full dataset
x$IID #to extract the individual ID's of full dataset
x$PRS #to extract the polygenic risk scores of full dataset
y <- PRS_binary(plink_path, DummyData, summary_input = add)
sink("B_add.sscore") #to create a file in the working directory
```

```
write.table(y, sep = " ", row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
z <- PRS_binary(plink_path, DummyData, summary_input = gxe)
sink("B_gxe.sscore") #to create a file in the working directory
write.table(z, sep = " ", row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output

## End(Not run)
```

`PRS_quantitative` *PRS_quantitative function This function uses plink2 and outputs Polygenic Risk Scores (PRSs) of all the individuals, using pre-generated GWAS and/or GWEIS summary statistics. Note that the input used in this function can be generated by using GWAS_quantitative and/or GWEIS_quantitative functions. Users may save the output in a user-specified file (see examples).*

Description

PRS_quantitative function This function uses plink2 and outputs Polygenic Risk Scores (PRSs) of all the individuals, using pre-generated GWAS and/or GWEIS summary statistics. Note that the input used in this function can be generated by using GWAS_quantitative and/or GWEIS_quantitative functions. Users may save the output in a user-specified file (see examples).

Usage

```
PRS_quantitative(plink_path, b_file, summary_input)
```

Arguments

`plink_path`	Path to the PLINK executable application
`b_file`	Prefix of the binary files, where all .fam, .bed and .bim files have a common prefix
`summary_input`	Pre-generated GWAS and/or GWEIS summary statistics

Value

This function will output

`prs.sscore`	PRSs for each individual

Examples

```
## Not run:
a <- GWAS_quantitative(plink_path, DummyData, Qphe_discovery, Qcov_discovery)
trd <- a[c("ID", "A1", "BETA")]
b <- GWEIS_quantitative(plink_path, DummyData, Qphe_discovery, Qcov_discovery)
add <- b[c("ID", "A1", "ADD_BETA")]
gxe <- b[c("ID", "A1", "INTERACTION_BETA")]
x <- PRS_quantitative(plink_path, DummyData, summary_input = trd)
sink("Q_trd.sscore") #to create a file in the working directory
write.table(x, sep = " ", row.names = FALSE, quote = FALSE) #to write the output
```

```
sink() #to save the output
head(x) #to read the head of all columns in the output
x$FID #to extract the family ID's of full dataset
x$IID #to extract the individual ID's of full dataset
x$PRS #to extract the polygenic risk scores of full dataset
y <- PRS_quantitative(plink_path, DummyData, summary_input = add)
sink("Q_add.sscore") #to create a file in the working directory
write.table(y, sep = " ", row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output
z <- PRS_quantitative(plink_path, DummyData, summary_input = gxe)
sink("Q_gxe.sscore") #to create a file in the working directory
write.table(z, sep = " ", row.names = FALSE, quote = FALSE) #to write the output
sink() #to save the output

## End(Not run)
```

`Qcov_discovery` *Covariate data file of the discovery dataset when the outcome is quantitative. This contains covariate information of the individuals in the discovery dataset following confounders.*

Description

Covariate data file of the discovery dataset when the outcome is quantitative. This contains covariate information of the individuals in the discovery dataset following confounders.

Usage

```
Qcov_discovery
```

Format

A dataframe with 800 rows and 18 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Standardized covariate

Column 4 Square of the standardized covariate

Column 5 Confounder 1

Column 6 Confounder 2

Column 7 Confounder 3

Column 8 Confounder 4

Column 9 Confounder 5

Column 10 Confounder 6

Column 11 Confounder 7

Column 12 Confounder 8

Column 13 Confounder 9

Column 14 Confounder 10

Column 15 Confounder 11

Column 16 Confounder 12

Column 17 Confounder 13

Column 18 Confounder 14

`Qcov_target` *Covariate data file of the target dataset when the outcome is quantitative. This contains covariate information of the individuals in the target dataset following confounders.*

Description

Covariate data file of the target dataset when the outcome is quantitative. This contains covariate information of the individuals in the target dataset following confounders.

Usage

```
Qcov_target
```

Format

A dataframe with 200 rows and 18 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Standardized covariate

Column 4 Square of the standardized covariate

Column 5 Confounder 1

Column 6 Confounder 2

Column 7 Confounder 3

Column 8 Confounder 4

Column 9 Confounder 5

Column 10 Confounder 6

Column 11 Confounder 7

Column 12 Confounder 8

Column 13 Confounder 9

Column 14 Confounder 10

Column 15 Confounder 11

Column 16 Confounder 12

Column 17 Confounder 13

Column 18 Confounder 14

`Qphe_discovery`	*Phenotype data file of the discovery dataset when the outcome is quantitative. This contains phenotype information of the individuals in the discovery dataset.*

Description

Phenotype data file of the discovery dataset when the outcome is quantitative. This contains phenotype information of the individuals in the discovery dataset.

Usage

```
Qphe_discovery
```

Format

A dataframe with 800 rows and 3 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Phenotype

`Qphe_target`	*Phenotype data file of the target dataset when the outcome is quantitative. This contains phenotype information of the individuals in the target dataset.*

Description

Phenotype data file of the target dataset when the outcome is quantitative. This contains phenotype information of the individuals in the target dataset.

Usage

```
Qphe_target
```

Format

A dataframe with 200 rows and 3 columns

Column 1 Family ID

Column 2 Individual ID

Column 3 Phenotype

summary_permuted_binary

summary_permuted_binary function This function outputs the p value of permuted model in the target dataset, using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_quantitative function. It is recommended to run this function, if you choose to fit 'PRS_gxe x E' interaction component (i.e. novel proposed model, Model 5) when generating risk scores. If the 'PRS_gxe x E' term is significant in Model 5, and insignificant in Model 5 (permuted p value), consider that the 'PRS_gxe x E' interaction component is actually insignificant (always give priority to the p value obtained from the permuted model).*

Description

summary_permuted_binary function This function outputs the p value of permuted model in the target dataset, using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_quantitative function. It is recommended to run this function, if you choose to fit 'PRS_gxe x E' interaction component (i.e. novel proposed model, Model 5) when generating risk scores. If the 'PRS_gxe x E' term is significant in Model 5, and insignificant in Model 5* (permuted p value), consider that the 'PRS_gxe x E' interaction component is actually insignificant (always give priority to the p value obtained from the permuted model).

Usage

```
summary_permuted_binary(
  Bphe_target,
  Bcov_target,
  iterations = 1000,
  add_score,
  gxe_score
)
```

Arguments

Bphe_target	Phenotype file containing family ID, individual ID and phenotype of the target dataset as columns, without heading
Bcov_target	Covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the target dataset as columns, without heading
iterations	Number of iterations used in permutation
add_score	PRSs generated using additive SNP effects of GWEIS summary statistics
gxe_score	PRSs generated using interaction SNP effects of GWEIS summary statistics

Value

This function will output

B_permuted_p	the p value of the permuted model

Examples

```
## Not run:
a <- GWEIS_binary(plink_path, DummyData, Bphe_discovery, Bcov_discovery)
add <- a[c("ID", "A1", "ADD_OR")]
gxe <- a[c("ID", "A1", "INTERACTION_OR")]
p <- PRS_binary(plink_path, DummyData, summary_input = add)
q <- PRS_binary(plink_path, DummyData, summary_input = gxe)
x <- summary_permuted_binary(Bphe_target, Bcov_target, iterations = 1000,
add_score = p, gxe_score = q)
x

## End(Not run)
```

`summary_permuted_quantitative`

summary_permuted_quantitative function This function outputs the p value of permuted model in the target dataset, using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_quantitative functions. It is recommended to run this function, if you choose to fit 'PRS_gxe x E' interaction component (i.e. novel proposed model, Model 4) when generating risk scores. If the 'PRS_gxe x E' term is significant in Model 4, and insignificant in Model 4 (permuted p value), consider that the 'PRS_gxe x E' interaction component is actually insignificant (always give priority to the p value obtained from the permuted model).*

Description

summary_permuted_quantitative function This function outputs the p value of permuted model in the target dataset, using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_quantitative functions. It is recommended to run this function, if you choose to fit 'PRS_gxe x E' interaction component (i.e. novel proposed model, Model 4) when generating risk scores. If the 'PRS_gxe x E' term is significant in Model 4, and insignificant in Model 4* (permuted p value), consider that the 'PRS_gxe x E' interaction component is actually insignificant (always give priority to the p value obtained from the permuted model).

Usage

```
summary_permuted_quantitative(
  Qphe_target,
  Qcov_target,
  iterations = 1000,
  add_score,
  gxe_score
)
```

Arguments

`Qphe_target`	Phenotype file containing family ID, individual ID and phenotype of the target dataset as columns, without heading

Qcov_target	Covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the target dataset as columns, without heading
iterations	Number of iterations used in permutation
add_score	PRSs generated using additive SNP effects of GWEIS summary statistics
gxe_score	PRSs generated using interaction SNP effects of GWEIS summary statistics

Value

This function will output

Q_permuted_p the p value of the permuted model

Examples

```
## Not run:
a <- GWEIS_quantitative(plink_path, DummyData, Qphe_discovery, Qcov_discovery)
add <- a[c("ID", "A1", "ADD_BETA")]
gxe <- a[c("ID", "A1", "INTERACTION_BETA")]
p <- PRS_quantitative(plink_path, DummyData, summary_input = add)
q <- PRS_quantitative(plink_path, DummyData, summary_input = gxe)
x <- summary_permuted_quantitative(Qphe_target, Qcov_target, iterations = 1000,
add_score = p, gxe_score = q)
x

## End(Not run)
```

---

summary_regular_binary

summary_regular_binary function This function outputs the summary of regular model and final risk score values of each individual in the target dataset using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_binary function.

Description

summary_regular_binary function This function outputs the summary of regular model and final risk score values of each individual in the target dataset using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_binary function.

Usage

```
summary_regular_binary(
  Bphe_target,
  Bcov_target,
  add_score = NULL,
  gxe_score = NULL,
  Model
)
```

Arguments

`Bphe_target`	Phenotype file containing family ID, individual ID and phenotype of the target dataset as columns, without heading
`Bcov_target`	Covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the target dataset as columns, without heading
`add_score`	PRSs generated using additive SNP effects of GWAS/GWEIS summary statistics
`gxe_score`	PRSs generated using interaction SNP effects of GWEIS summary statistics
`Model`	Specify the model number (0: y = PRS_trd + E + confounders, 1: y = PRS_trd + E + PRS_trd x E + confounders, 2: y = PRS_add + E + PRS_add x E + confounders, 3: y = PRS_add + E + PRS_gxe x E + confounders, 4: y = PRS_add + E + PRS_gxe + PRS_gxe x E + confounders, 5: y = PRS_add + E + E^2 + PRS_gxe + PRS_gxe x E + confounders, where y is the outcome variable, E is the covariate of interest, PRS_trd and PRS_add are the polygenic risk scores computed using additive SNP effects of GWAS and GWEIS summary statistics respectively, and PRS_gxe is the polygenic risk scores computed using GxE interaction SNP effects of GWEIS summary statistics.)

Value

This function will output

`Bsummary`	the summary of the fitted model
`Individual_risk_values`	the estimated risk values of individuals in the target sample

Examples

```
## Not run:
a <- GWAS_binary(plink_path, DummyData, Bphe_discovery, Bcov_discovery)
trd <- a[c("ID", "A1", "OR")]
b <- GWEIS_binary(plink_path, DummyData, Bphe_discovery, Bcov_discovery)
add <- b[c("ID", "A1", "ADD_OR")]
gxe <- b[c("ID", "A1", "INTERACTION_OR")]
p <- PRS_binary(plink_path, DummyData, summary_input = trd)
q <- PRS_binary(plink_path, DummyData, summary_input = add)
r <- PRS_binary(plink_path, DummyData, summary_input = gxe)
summary_regular_binary(Bphe_target, Bcov_target,
                            add_score = p,
                            Model = 0)
summary_regular_binary(Bphe_target, Bcov_target,
                            add_score = p,
                            Model = 1)
summary_regular_binary(Bphe_target, Bcov_target,
                            add_score = q,
                            Model = 2)
summary_regular_binary(Bphe_target, Bcov_target,
                            add_score = q,
                            gxe_score = r,
                            Model = 3)
summary_regular_binary(Bphe_target, Bcov_target,
                            add_score = q,
```

```
                              gxe_score = r,
                              Model = 4)
x <- summary_regular_binary(Bphe_target, Bcov_target,
                              add_score = q,
                              gxe_score = r,
                              Model = 5)
sink("Bsummary.txt") #to create a file in the working directory
print(x$summary) #to write the output
sink() #to save the output
sink("Individual_risk_values.txt") #to create a file in the working directory
write.table(x$risk.values, sep = " ", row.names = FALSE, col.names = FALSE,
quote = FALSE) #to write the output
sink() #to save the output
x$summary #to obtain the model summary output
x$risk.values #to obtain the predicted risk values of target individuals

## End(Not run)
```

`summary_regular_quantitative`

summary_regular_quantitative function This function outputs the summary of regular model and final risk score values of each individual in the target dataset using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_quantitative function.

Description

summary_regular_quantitative function This function outputs the summary of regular model and final risk score values of each individual in the target dataset using pre-generated Polygenic Risk Scores (PRSs) of all the individuals. Note that the input used in this function can be generated by using PRS_quantitative function.

Usage

```
summary_regular_quantitative(
  Qphe_target,
  Qcov_target,
  add_score = NULL,
  gxe_score = NULL,
  Model
)
```

Arguments

`Qphe_target`	Phenotype file containing family ID, individual ID and phenotype of the target dataset as columns, without heading
`Qcov_target`	Covariate file containing family ID, individual ID, standardized covariate, square of standardized covariate, and/or confounders of the target dataset as columns, without heading
`add_score`	PRSs generated using additive SNP effects of GWAS/GWEIS summary statistics

`gxe_score` PRSs generated using interaction SNP effects of GWEIS summary statistics

`Model` Specify the model number (0: y = PRS_trd + E + confounders, 1: y = PRS_trd + E + PRS_trd x E + confounders, 2: y = PRS_add + E + PRS_add x E + confounders, 3: y = PRS_add + E + PRS_gxe x E + confounders, 4: y = PRS_add + E + PRS_gxe + PRS_gxe x E + confounders, where y is the outcome variable, E is the covariate of interest, PRS_trd and PRS_add are the polygenic risk scores computed using additive SNP effects of GWAS and GWEIS summary statistics respectively, and PRS_gxe is the polygenic risk scores computed using GxE interaction SNP effects of GWEIS summary statistics.)

Value

This function will output

`Qsummary.txt` the summary of the fitted model

`Individual_risk_values.txt`
the estimated risk values of individuals in the target sample

Examples

```
## Not run:
a <- GWAS_quantitative(plink_path, DummyData, Qphe_discovery, Qcov_discovery)
trd <- a[c("ID", "A1", "BETA")]
b <- GWEIS_quantitative(plink_path, DummyData, Qphe_discovery, Qcov_discovery)
add <- b[c("ID", "A1", "ADD_BETA")]
gxe <- b[c("ID", "A1", "INTERACTION_BETA")]
p <- PRS_quantitative(plink_path, DummyData, summary_input = trd)
q <- PRS_quantitative(plink_path, DummyData, summary_input = add)
r <- PRS_quantitative(plink_path, DummyData, summary_input = gxe)
summary_regular_quantitative(Qphe_target, Qcov_target,
                             add_score = p,
                             Model = 0)
summary_regular_quantitative(Qphe_target, Qcov_target,
                             add_score = p,
                             Model = 1)
summary_regular_quantitative(Qphe_target, Qcov_target,
                             add_score = q,
                             Model = 2)
summary_regular_quantitative(Qphe_target, Qcov_target,
                             add_score = q,
                             gxe_score = r,
                             Model = 3)
x <- summary_regular_quantitative(Qphe_target, Qcov_target,
                             add_score = q,
                             gxe_score = r,
                             Model = 4)
sink("Qsummary.txt") #to create a file in the working directory
print(x$summary) #to write the output
sink() #to save the output
sink("Individual_risk_values.txt") #to create a file in the working directory
write.table(x$risk.values, sep = " ", row.names = FALSE, col.names = FALSE,
quote = FALSE) #to write the output
sink() #to save the output
x$summary #to obtain the model summary output
x$risk.values #to obtain the predicted risk values of target individuals
```

```
## End(Not run)
```

Index