

# Package ‘GWhEAT’

March 26, 2020

**Type** Package

**Title** Performed GWAS in a fast and efficient manner

**Version** 0.1.0

**Author** Lance Merrick, Samuel Prather

**Maintainer** Samuel Prather <stp4freedom@gmail.com>

**Description** This package can run GWAS with PCA and user imputed covariates.  
It's design is to reduce false positives and increases the power of a GWAS by taking into account population structure.

**License** What license is it under?

**Encoding** UTF-8

**LazyData** true

**Imports** ggplot2,  
knitr,  
gridExtra,  
data.table

**RoxygenNote** 7.0.2

## R topics documented:

fix_Dep . . . . .	2
GWASapply . . . . .	2
GWASapply_rapid . . . . .	3
GWAStest . . . . .	4
manhattan_plot . . . . .	5
power.fdr . . . . .	5
qq_plot . . . . .	6
<b>Index</b>	7

---

fix_Dep	<i>Detect linear dependencies of one matrix on another</i>
---------	--

---

### Description

Detect linear dependencies of one matrix on another

### Usage

```
fix_Dep(X1, X2, tol = .Machine$double.eps^0.5, rank.def = 0, strict = FALSE)
```

### Arguments

X1	A matrix.
X2	A matrix, the columns of which may be partially linearly dependent on the columns of X1.
tol	The tolerance to use when assessing linear dependence.
rank.def	If the degree of rank deficiency in X2, given X1, is known, then it can be supplied here, and tol is then ignored. Unused unless positive and not greater than the number of columns in X2.
strict	if TRUE then only columns individually dependent on X1 are detected, if FALSE then enough columns to make the reduced X2 full rank and independent of X1 are detected.

### Value

A vector of the columns of X2 which are linearly dependent on columns of X1 (or which need to be deleted to achieve independence and full rank if strict==FALSE). NULL if the two matrices are independent.

---

GWASapply	<i>GWAS with PCA</i>
-----------	----------------------

---

### Description

GWAS with PCA

### Usage

```
GWASapply(
  pheno = NULL,
  geno = NULL,
  Cov = NULL,
  GM = NULL,
  PCA.M = 3,
  QTN.position = NULL,
  cutoff = NULL,
  plots = FALSE,
```

```

    messages = FALSE,
    print = FALSE,
    trait = "unknown",
    rapid = FALSE
  )

```

### Arguments

pheno	file with numeric phenotypic values
geno	data.frame with genotype calls coded as 0,1,2.
Cov	numeric data.frame with covariates values
GM	genetic map of data with chr and position of each SNP
PCA.M	number of principal components to use default is 3
QTN.position	posistion of QTN if applicable
cutoff	If cutoff is default, uses Bonferroni;0.05/number of SNPs
plots	if TRUE, function plots PCA graphs, Manhattan Plot and QQ plot
messages	if TRUE, returns messages for the GWAS function
print	if TRUE, results are saved in a CSV
trait	character value for trait name
rapid	this option skips all side options and returns just GWAS results

### Value

Manhattan plot, QQ plot plus p-values, type-1 error and power for every SNP and results in a CSV

---

GWASapply_rapid	<i>Rapid GWAS with PCA</i>
-----------------	----------------------------

---

### Description

Rapid GWAS with PCA

### Usage

```

GWASapply_rapid(
  pheno = NULL,
  geno = NULL,
  Cov = NULL,
  GM = NULL,
  PCA.M = 3,
  cutoff = NULL
)

```

**Arguments**

pheno	file with numeric phenotypic values
geno	data.frame with genotype calls coded as 0,1,2.
Cov	numeric data.frame with covariates values
GM	genetic map of data with chr and position of each SNP
PCA.M	number of principal components to use default is 3
cutoff	If cutoff is default, uses Bonferroni; 0.05/number of SNPs

**Value**

GWAS P-values

---

GWAS <sub>test</sub>	<i>GWAS with PCA</i>
----------------------	----------------------

---

**Description**

GWAS with PCA

**Usage**

```
GWAStest(
  phenotypes = NULL,
  genotypes = NULL,
  Cov = NULL,
  GM = NULL,
  PCA.M = 3,
  QTN.position = NULL,
  cutoff = NULL
)
```

**Arguments**

phenotypes	file with numeric phenotypic values
genotypes	data.frame with genotype calls coded as 0,1,2.
Cov	numeric data.frame with covariates values
GM	genetic map of data with chr and position of each SNP
PCA.M	number of principal components to use default is 3
QTN.position	posistion of QTN if applicable
cutoff	If cutoff is default, uses Bonferroni; else uses -log(value) of 0.05/number of SNPs

**Value**

Manhattan plot, QQ plot plus p-values, type-1 error and power for every SNP

---

manhattan_plot	<i>Function to create a manhattan plot</i>
----------------	--

---

**Description**

Function to create a manhattan plot

**Usage**

```
manhattan_plot(
  GM,
  pvals,
  cutoff = NULL,
  QTN_index = c(),
  FP = NULL,
  TP = NULL,
  trait = "unknown",
  messages = FALSE
)
```

**Arguments**

GM	genetic map of data with chr and position of each SNP
pvals	pvals from gwas results for each SNP
cutoff	If cutoff is default, uses Bonferroni; 0.05/number of SNPs
QTN_index	posistion of QTN if applicable
FP	Positions of SNPS that are False positives
TP	Positions of SNPS that are True positives
trait	character value for trait name
messages	if TRUE, returns messages for the GWAS function

**Value**

Manhattan plot

---

power.fdr	<i>Function to caculate power, FDR and type-1 error, False Positives, and True Positives</i>
-----------	--

---

**Description**

Function to caculate power, FDR and type-1 error, False Positives, and True Positives

**Usage**

```
power.fdr(P.value, QTN.position = NULL, cutoff = NULL)
```

**Arguments**

QTN.position	position of QTN if known
cutoff	If cutoff is default, uses Bonferroni;0.05/number of SNPs
P	list of SNPs order by ascending p-value

**Value**

list of power, FDR, type-1 error, False Positives, and True Positives

---

qq_plot	<i>Function to create a QQ plot</i>
---------	-------------------------------------

---

**Description**

Function to create a QQ plot

**Usage**

```
qq_plot(GM, pvals, QTN_index = c(), trait = "unknown")
```

**Arguments**

GM	genetic map of data with chr and position of each SNP
pvals	pvals from gwas results for each SNP
QTN_index	posistion of QTN if applicable
trait	character value for trait name

**Value**

QQ plot

# Index

`fix_Dep`, [2](#)

`GWASapply`, [2](#)

`GWASapply_rapid`, [3](#)

`GWAStest`, [4](#)

`manhattan_plot`, [5](#)

`power.fdr`, [5](#)

`qq_plot`, [6](#)