Package 'gwid'

December 14, 2023

Type Package

Title Genome-Wide Identity-by-Descent

Version 0.1.0

Maintainer Soroush Mahmoudiandehkordi <soroushmdg@gmail.com>

Description Methods and tools for the analysis of Genome Wide
Identity-by-Descent ('gwid') mapping data, focusing on testing whether there
is a higher occurrence of Identity-By-Descent (IBD) segments around potential causal variants
in cases compared to controls, which is crucial for identifying rare
variants. To enhance its analytical power, 'gwid' incorporates a Sliding
Window Approach, allowing for the detection and analysis of signals from
multiple Single Nucleotide Polymorphisms (SNPs).

License MIT + file LICENSE

Encoding UTF-8

Imports data.table, gdsfmt, SNPRelate, Matrix, ggplot2, plotly, utils,
stats, RcppRoll, methods, piggyback

RoxygenNote 7.2.3

Suggests knitr, magrittr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://github.com/soroushmdg/gwid

BugReports https://github.com/soroushmdg/gwid/issues

NeedsCompilation no

Author Soroush Mahmoudiandehkordi [aut, cre],
Mehdi Maadooliat [aut],
Steven Schrodi [aut]

Repository CRAN

Date/Publication 2023-12-14 16:40:08 UTC

R topics documented:

build_gwas *Open a SNP GDS file and extract information.*

Description

Open a SNP GDS file and extract information.

Usage

```
build_gwas(gds_data = "name.gds", caco = "name.Rda", gwas_generator = TRUE)
```

Arguments

gds_data	File name
caco	An object of class caco. Output of case_control function.
gwas_generator	logical; if TRUE an object of class result_snps will be saved inside output list.

Value

a list of seven objects; including smp.id, snp.id, snp.pos, smp.indx, smp.snp (a matrix with samples in rows and snp in columns), caco, snps(column sum of smp.snp for each case control)

build_gwid *Open a ibd file and extract information.*

Description

Open a ibd file and extract information.

Usage

```
build_gwid(
  ibd_data = "name.ibd",
  gwas = "object of class gwas",
  gwid_generator = TRUE
)
```

Arguments

ibd_data	a file name for output of Refined IBD
gwas	object of class gwas
gwid_generator	logical; if TRUE an object of class result_snps will be saved inside output list.

Value

the output will be a object(list) of class gwid contains profile object, IBD object and result_snps object.

build_phase *Read .vcf structured text format files and reduce the size of file.*

Description

Read .vcf structured text format files and reduce the size of file.

Usage

```
build_phase(phased_vcf = "name.vcf", caco)
```

Arguments

phased_vcf	A file name for a variant call format (vcf) file.
caco	An object of class caco. Output of `case_control` function.

Value

the output will be a a list of class phase contains two sparse matrix for each haplotype.

case_control *Reload saved case-control list file*

Description

Reload saved case-control list file

Usage

```
case_control(case_control_rda, ...)
```

Arguments

case_control_rda	A character string giving the name of the case-control file to load. The file is a list of character vectors including subject names in each case-control groups or csv file including subject name for a disease.
...	name of a column (disease name) of csv file.

Value

The output will be a list of character vectors include subject names and groups. The class of returned object is caco.

extract *Extract information from SNP GDS file.*

Description

Extract information from SNP GDS file.

Usage

```
extract(obj, ...)
```

Arguments

obj	an object of class gwas
...	other arguments

Value

extract object instants

extract.gwas *Extract information from SNP GDS file.*

Description

Extract information from SNP GDS file.

Usage

```
## S3 method for class 'gwas'
extract(obj, type = c("snps", "snp2", "nas"), snp_start, snp_end, ...)
```

Arguments

obj	object of class gwas.
type	indicate type of aggregation on sample-snp data and must be one of snps, snp2, or nas
snp_start	select starting position of snp, which we want to aggregate.
snp_end	select ending position of snp, which we want to aggregate.
...	other arguments

Value

the output will be a result_snps (data.table) object including 3 columns including, snp_pos, case_control, and value

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`extract.gwid` *Extract information from ibd data.*

Description

Extract information from ibd data.

Usage

```
## S3 method for class 'gwid'
extract(obj = "object of class gwid", snp_start, snp_end, ...)
```

Arguments

obj	object of class gwid(output of function build_gwid)
snp_start	select starting position of snp, which we want to aggregate.
snp_end	select ending position of snp, which we want to aggregate.
...	other objects

Value

the output will be a result_snps (data.table) object including 3 columns including, "snp_pos", "case_control", and "value"

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
```

```
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`extract_window` *extract component of an object*

Description

extract component of an object

Usage

```
extract_window(obj, ...)
```

Arguments

`obj`	obj
`...`	other variables

Value

the output will be a result_snps (data.table) object including 3 columns including, "snp_pos", "case_control", and "value"

`extract_window.gwid` *Extract information from ibd data in a moving window*

Description

Extract information from ibd data in a moving window

Usage

```
## S3 method for class 'gwid'
extract_window(obj, w = 10, snp_start, snp_end, ...)
```

Arguments

obj	object of class gwid(output of function build_gwid)
w	window size
snp_start	select starting position of snp, which we want to aggregate.
snp_end	select ending position of snp, which we want to aggregate.
...	other variables

Value

the output will be a result_snps (data.table) object including 3 columns including, "snp_pos", "case_control", and "value"

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
```

```
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`fisher_test` *Fisher test*

Description

Fisher test

Usage

```
fisher_test(obj, ...)
```

Arguments

`obj`	an object
`...`	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

`fisher_test.gwas` *Fisher's Exact Test for gwas count data*

Description

Fisher's Exact Test for gwas count data

Usage

```
## S3 method for class 'gwas'
fisher_test(
  obj,
  reference,
  snp_start,
  snp_end,
  alternative = c("two.sided", "greater", "less"),
  ...
)
```

Arguments

obj	object of class gwas
reference	reference group of subjects in which we want to perform fisher test test
snp_start	select starting position of snps.
snp_end	select ending position of snp.
alternative	indicates the alternative hypothesis and must be one of "two.sided", "greater" or "less". You can specify just the initial letter. Only used in the 2 by 2 case
...	optional arguments to fisher.test

Value

the output will be a test_snps (data.table) object including 3 columns: “snp_pos”, “case_control”, and “value” which is a p-values.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
```

```
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`fisher_test.gwid` *Fisher's Exact Test for gwid count data*

Description

Fisher's Exact Test for gwid count data

Usage

```
## S3 method for class 'gwid'
fisher_test(
  obj,
  caco,
  snp_start,
  snp_end,
  reference,
  alternative = c("two.sided", "greater", "less"),
  ...
)
```

Arguments

`obj`	An object of class gwid. Output of `build_gwid` function
`caco`	An object of class caco. Output of `case_control` function.
`snp_start`	select starting position of snps.
`snp_end`	select ending position of snp.
`reference`	reference group of subjects in which we want to perform fisher test
`alternative`	indicates the alternative hypothesis and must be one of "two.sided", "greater" or "less". You can specify just the initial letter. Only used in the 2 by 2 case
`...`	optional arguments to fisher.test

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`fisher_test.result_snps`

fisher exact test for result_snps count data

Description

fisher exact test for result_snps count data

Usage

```
## S3 method for class 'result_snps'
fisher_test(
  obj,
  caco,
  reference,
  alternative = c("two.sided", "greater", "less"),
  ...
)
```

Arguments

obj	An object of class result_snps
caco	An object of class caco. Output of case_control function.
reference	reference group of subjects in which we want to perform fisher test.
alternative	indicates the alternative hypothesis and must be one of "two.sided", "greater" or "less". You can specify just the initial letter. Only used in the 2 by 2 case
...	optional arguments to fisher.test

Value

the output will be a test_snps (data.table) object including 3 columns: “snp_pos”, “case_control”, and “value” which is a p-values.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
```

```
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
model_permutation <- permutation_test(ibd_data,snp_data_gds,
snp_start = 119026294,snp_end = 120613594,nperm=20,reference = "cases")
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`gtest` *perform gtest*

Description

perform gtest

Usage

```
gtest(haplotype_structure, ...)
```

Arguments

`haplotype_structure`	object of a class
`...`	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: “snp_pos”, “case_control”, and “value” which is a p-values.

gtest.haplotype_structure *Perform G-test on haplotype structures extracted from* haplotype_structure *function*

Description

Perform G-test on haplotype structures extracted from haplotype_structure function

Usage

```
## S3 method for class 'haplotype_structure'
gtest(haplotype_structure, reference, ...)
```

Arguments

haplotype_structure	An object of class haplotype_structure. Output of haplotype_structure function.
reference	reference group of subjects in which we want to perform G-test
...	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

haplotype_frequency *haplotype frequency*

Description

haplotype frequency

Usage

```
haplotype_frequency(haplotype_structure, ...)
```

Arguments

haplotype_structure	object of class haplotype structure
...	other variables

Value

An object of class haplotype_frequency contains of two objects. first one is object of haplotype_structure_frequency (data.table) and second one is object of class result_snps(data.table)

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`haplotype_frequency.haplotype_structure` *haplotype frequency in sliding windows*

Description

haplotype frequency in sliding windows

Usage

```
## S3 method for class 'haplotype_structure'
haplotype_frequency(haplotype_structure, ...)
```

Arguments

`haplotype_structure`	An object of class haplotype_structure. Output of `haplotype_structure` function.
`...`	other variables

Value

An object of class haplotype_frequency contains of two objects. first one is object of haplotype_structure_frequency (data.table) and second one is object of class result_snps(data.table)

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
```

```
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`haplotype_structure` *haplotype structures in a window*

Description

haplotype structures in a window

Usage

```
haplotype_structure(obj, ...)
```

Arguments

`obj`	object
`...`	other variables

Value

The output will be an object of class haplotype_structure (data.table) that has information about subjects haplotype structures in a a window.

haplotype_structure.gwas *extract haplotype structures of individuals in a window*

Description

extract haplotype structures of individuals in a window

Usage

```
## S3 method for class 'gwas'
haplotype_structure(obj, phase, w = 10, snp_start, snp_end, ...)
```

Arguments

obj	object of class gwas
phase	An object of class phase. Output of build_phase function
w	window size
snp_start	select starting position of snps.
snp_end	select ending position of snps.
...	other variables

Value

The output will be an object of class haplotype_structure (data.table) that has information about subjects haplotype structures in a a window.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
```

```
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`haplotype_structure.gwid` *extract haplotype structures of pairwise ibd samples in a window*

Description

extract haplotype structures of pairwise ibd samples in a window

Usage

```
## S3 method for class 'gwid'
haplotype_structure(obj, phase, w = 10, snp_start, snp_end, ...)
```

Arguments

`obj`	An object of class gwid. Output of `build_gwid` function.
`phase`	An object of class phase. Output of `build_phase` function.
`w`	window size
`snp_start`	select starting position of snps.
`snp_end`	select ending position of snps.
`...`	other variables

Value

The output will be an object of class haplotype_structure (data.table) that has information about subjects haplotype structures in a a window.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

mcnemar_test *mcnemar test*

Description

mcnemar test

Usage

```
mcnemar_test(roh, ...)
```

Arguments

roh	roh as class result_snp
...	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

mcnemar_test.result_snps *mcnemar test*

Description

mcnemar test

Usage

```
## S3 method for class 'result_snps'
mcnemar_test(
  roh = "object of class result_snps (output of function roh with fun=sum)",
  reference,
  w = 10,
  ...
)
```

Arguments

roh	An object of class result_snps (output of function roh with fun=sum)
reference	reference group of subjects in which we want to perform fisher test.
w	window size
...	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

`mcnemar_test_permut` *mcnemar permutation*

Description

mcnemar permutation

Usage

```
mcnemar_test_permut(mcnemar, ...)
```

Arguments

`mcnemar`	macnemar test output
`...`	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

`mcnemar_test_permut.result_snps` *mcnemar permutation test*

Description

mcnemar permutation test

Usage

```
## S3 method for class 'result_snps'
mcnemar_test_permut(
 mcnemar = "object of class result_snps (output of function mcnemar_test with fun=sum)",
  roh_mat = "output of roh function when roh_mat = TRUE",
  gwas = "object of class gwas",
  nperm = 1000,
  reference = "cases",
  w,
  ...
)
```

Arguments

mcnemar	macnemar test output
roh_mat	roh matrix
gwas	gwas
nperm	number of permutation
reference	reference group
w	window
...	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

---

permutation_test *permutation test*

Description

permutation test

Usage

```
permutation_test(obj, ...)
```

Arguments

obj	object
...	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

`permutation_test.gwas` *Permutation test for gwas object*

Description

Permutation test for gwas object

Usage

```
## S3 method for class 'gwas'
permutation_test(
  obj,
  snp_start,
  snp_end,
  nperm = 1000,
  reference = "cases",
  ...
)
```

Arguments

`obj`	object of class gwas
`snp_start`	elect starting position of snps.
`snp_end`	select ending position of snp.
`nperm`	Number of permutations.
`reference`	reference group of subjects in which we want to perform fisher test
`...`	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: “snp_pos”, “case_control”, and “value” which is a p-values.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
```

```
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
model_permutation <- permutation_test(ibd_data,snp_data_gds,
snp_start = 119026294,snp_end = 120613594,nperm=20,reference = "cases")
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`permutation_test.gwid` *permutation test for gwid count data*

Description

permutation test for gwid count data

Usage

```
## S3 method for class 'gwid'
permutation_test(
  obj,
  gwas,
  snp_start,
  snp_end,
  nperm = 100,
  reference = "cases",
```

```
  ...
)
```

Arguments

obj	An object of class gwid. Output of build_gwid function
gwas	object of class gwas
snp_start	select starting position of snps.
snp_end	select ending position of snp.
nperm	Number of permutations.
reference	reference group
...	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
```

```
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
model_permutation <- permutation_test(ibd_data,snp_data_gds,
snp_start = 119026294,snp_end = 120613594,nperm=20,reference = "cases")
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`permutation_test.haplotype_structure` *Permutation test for 'haplotype_structure' object*

Description

Permutation test for 'haplotype_structure' object

Usage

```
## S3 method for class 'haplotype_structure'
permutation_test(obj, nperm, reference, ...)
```

Arguments

`obj`	object of class 'haplotype_structure'
`nperm`	Number of permutations.
`reference`	reference group of subjects in which we want to perform 'gtest'
`...`	other variables

Value

the output will be a test_snps (data.table) object including 3 columns: "snp_pos", "case_control", and "value" which is a p-values.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
```

```
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
model_permutation <- permutation_test(ibd_data,snp_data_gds,
snp_start = 119026294,snp_end = 120613594,nperm=20,reference = "cases")
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`plot.gwas` *Line plot of gwas objects*

Description

Line plot of gwas objects

Usage

```
## S3 method for class 'gwas'
plot(x, y = NA, title = "number of snps", ...)
```

Arguments

x	object of class gwas.
y	default value is NA, if specified it should be a vector of names of subject groups i.e. y = c("case","control")
title	title of the plot.
...	optional argument of plot

Value

an interactive line plot of gwas objects for each case control subjects.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
```

```
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`plot.gwid` *Line plot of gwid objects*

Description

Line plot of gwid objects

Usage

```
## S3 method for class 'gwid'
plot(
  x,
  y = NA,
  title = "number of IBD in each snp",
  plot_type = c("result_snps", "profile"),
  reference,
  ...
)
```

Arguments

`x`	An object of class gwid. Output of `build_gwid` function.
`y`	default value is NA, if specified it should be a vector of names of subject groups i.e. y = c("case","control")
`title`	title of the plot.
`plot_type`	either "result_snps" or "profile".
`reference`	reference group of subjects in which we want to have profile plot.
`...`	if plot_type is "result_snps" it is optional argument of `plot`. if plot_type is "profile" we can subset plot based on snp_start and snp_end locations.

Value

if plot_type is "result_snps" an interactive line plot of result_snps for each case control subjects. if plot_type is "profile" an interactive profile plot of identity by descent subjects in subset of locations.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

---

plot.haplotype_frequency

Line plot of haplotype_frequency object

Description

Line plot of haplotype_frequency object

Usage

```
## S3 method for class 'haplotype_frequency'
plot(
  x,
  y = NA,
  plot_type = c("haplotype_structure_frequency", "result_snps"),
  type = c("version1", "version2"),
  ly = TRUE,
  nwin,
  title,
  line_size = 0.6,
  ...
)
```

Arguments

x	an object of class haplotype_frequency
y	default value is 'NA', if specified it should be a vector of names of subject groups i.e. 'y = c("case","control")'
plot_type	either "result_snps" or ""haplotype_structure_frequency""
type	either "version1" or "version2" when plot_type is ""haplotype_structure_frequency""
ly	if TRUE, we have a plotly object and if it is false plot is going to be a ggplot object.
nwin	window number
title	title of the plot.
line_size	geom_line size
...	optional argument of plot

Value

an interactive line plot of haplotype_frequency objects for each case control subjects.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
```

```
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`plot.haplotype_structure_frequency` *Two type of line plots for haplotype_structure_frequency objects .*

Description

Two type of line plots for haplotype_structure_frequency objects .

Usage

```
## S3 method for class 'haplotype_structure_frequency'
plot(
  x,
  y = NA,
  type = c("version1", "version2"),
  nwin,
  ly = TRUE,
  line_size = 0.6,
  ...
)
```

Arguments

x	an object of class haplotype_structure_frequency
y	default value is NA, if specified it should be a vector of names of subject groups i.e. y = c("case","control")
type	either “version1” or “version2”
nwin	window number
ly	if ‘TRUE‘, we have a ‘plotly‘ object and if it is ‘FALSE‘ plot is going to be a ‘ggplot‘ object.
line_size	geom_line size
...	other variables

Value

an interactive line plot of haplotype_structure_frequency objects for each case control subjects.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
```

```
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`plot.result_snps` *Line plot of result_snps objects*

Description

Line plot of result_snps objects

Usage

```
## S3 method for class 'result_snps'
plot(x, y = NA, title, snp_start, snp_end, ly = TRUE, line_size = 0.6, ...)
```

Arguments

`x`	An object of class result_snps.
`y`	default value is NA, if specified it should be a vector of names of subject groups i.e. y = c("case","control")
`title`	title of the plot.
`snp_start`	select starting position of snps.
`snp_end`	select ending position of snps.
`ly`	if TRUE, we have a plotly object and if it is false plot is going to be a ggplot object.
`line_size`	geom_line size
`...`	other variables

Value

an interactive line plot of result_snps for each case control subjects.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`plot.test_snps` *Line plot of test_snps objects*

Description

Line plot of test_snps objects

Usage

```
## S3 method for class 'test_snps'
plot(x, y = NA, title, snp_start, snp_end, ly = TRUE, line_size = 0.6, ...)
```

Arguments

x	an object of class test_snps.
y	default value is NA, if specified it should be a vector of names of subject groups i.e. y = c("case","control")
title	title of the plot.
snp_start	select starting position of snps.
snp_end	select ending position of snps.
ly	if 'TRUE', we have a 'plotly' object and if it is 'FALSE' plot is going to be a 'ggplot' object.
line_size	geom_line size
...	other variables

Value

an interactive line plot of test_snps objects for each case control subjects.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
```

```
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

`print` *print*

Description

print

Usage

```
print(x, ...)
```

Arguments

`x`	an object
`...`	other objects

Value

print an object

`print.gwas` *print gwas instants*

Description

print gwas instants

Usage

```
## S3 method for class 'gwas'
print(x, ...)
```

Arguments

x	object gwas
...	other objects

Value

print number of subjects and number of SNPs of a GWAS object

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
print(snp_data_gds)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

roh *runs of homozygosity*

Description

runs of homozygosity

Usage

```
roh(phase, ...)
```

Arguments

phase	object of phase
...	other variables

Value

runs of homozygosity data table or matrix

roh.phase *runs of homozygosity*

Description

runs of homozygosity

Usage

```
## S3 method for class 'phase'
roh(
  phase,
  gwas,
  w = 10,
  fun = c("sum", "mean"),
  snp_start,
  snp_end,
  roh_mat = FALSE,
  ...
)
```

Arguments

phase	An object of class phase. Output of build_phase function
gwas	object of class gwas
w	window size
fun	an aggregate function. either "sum" or "mean"
snp_start	select starting position of snps.
snp_end	select ending position of snps.
roh_mat	return roh as matrix
...	other variables

Value

the output will be a result_snps (data.table) object including 3 columns including, "snp_pos", "case_control", and "value"

---

subset *subset an object*

Description

subset an object

Usage

```
subset(obj, ...)
```

Arguments

obj	object
...	other variables

Value

the output will be a object(list) of class gwid contains profile object and result_snps object.

subset.gwid *subset gwid object based on snp position*

Description

subset gwid object based on snp position

Usage

```
## S3 method for class 'gwid'
subset(obj, snp_start, snp_end, ...)
```

Arguments

obj	object of class gwid(output of function build_gwid)
snp_start	select starting position of snp, which we want to aggregate.
snp_end	select ending position of snp, which we want to aggregate.
...	other variables

Value

the output will be a object(list) of class gwid contains profile object and result_snps object.

Examples

```
piggyback::pb_download(repo = "soroushmdg/gwid",tag = "v0.0.1",dest = tempdir())
ibd_data_file <- paste0(tempdir(),"//chr3.ibd")
genome_data_file <- paste0(tempdir(),"//chr3.gds")
phase_data_file <- paste0(tempdir(),"//chr3.vcf")
case_control_data_file <- paste0(tempdir(),"//case-cont-RA.withmap.Rda")
# case-control data
case_control <- gwid::case_control(case_control_rda = case_control_data_file)
names(case_control) #cases and controls group
summary(case_control) # in here, we only consider cases,cont1,cont2,cont3 groups in the study
case_control$cases[1:3] # first three subject names of cases group
# read SNP data (use SNPRelate to convert it to gds) and count number of minor alleles
snp_data_gds <- gwid::build_gwas(gds_data = genome_data_file,
caco = case_control,gwas_generator = TRUE)
class(snp_data_gds)
names(snp_data_gds)
head(snp_data_gds$snps) # it has information about counts of minor alleles in each location.
# read haplotype data (output of beagle)
haplotype_data <- gwid::build_phase(phased_vcf = phase_data_file,caco = case_control)
class(haplotype_data)
names(haplotype_data)
dim(haplotype_data$Hap.1) #22302 SNP and 1911 subjects
# read IBD data (output of Refined-IBD)
```

```
ibd_data <- gwid::build_gwid(ibd_data = ibd_data_file,gwas = snp_data_gds)
class(ibd_data)
ibd_data$ibd # refined IBD output
ibd_data$res # count number of IBD for each SNP location
# plot count of IBD in chromosome 3
plot(ibd_data,y = c("cases","cont1"),ly = FALSE)
# Further investigate location between 117M and 122M
# significant number of IBD's in group cases, compare to cont1, cont2 and cont3.
plot(ibd_data,y = c("cases","cont1"),snp_start = 119026294,snp_end = 120613594,ly = FALSE)
model_fisher <- gwid::fisher_test(ibd_data,case_control,reference = "cases",
snp_start = 119026294,snp_end = 120613594)
class(model_fisher)
plot(model_fisher, y = c("cases","cont1"),ly = FALSE)
hap_str <- gwid::haplotype_structure(ibd_data,phase = haplotype_data,w = 10,
snp_start = 119026294,snp_end = 120613594)
haplo_freq <- gwid::haplotype_frequency(hap_str)
plot(haplo_freq,y = c("cases", "cont1"),plot_type = "haplotype_structure_frequency",
nwin = 1, type = "version1",ly = FALSE)
```

Index