

Package ‘HAPPI.GWAS’

May 1, 2020

Title HAPPI.GWAS

Version 1.0.0

Author Yen On Chan [aut, cre]

Maintainer Yen On Chan <chanye@mail.missouri.edu>

Description HAPPI.GWAS is a package for GWAS.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 3.6.0)

biocViews

Imports methods, magrittr, dplyr, tidyr, ggplot2, tibble, stringr,
gplots, ape, BiocManager, car, data.table, DataCombine,
EMMREML, foreach, doParallel, lme4, scatterplot3d, genetics,
LDheatmap, gridExtra, yaml, bigmemory, biganalytics, Biobase,
BiocGenerics, snpStats, multtest, zlibbioc, maditr, MASS,
plotly

RoxygenNote 6.1.1

R topics documented:

extract_haplotype	2
farming_with_GAPIT	2
generate_BLUE	3
generate_BLUP	3
search_genes	4
Index	5

extract_haplotype	<i>Extract haplotype using Haploview</i>
-------------------	------------------------------------------

Description

The goal of search_genes is to extract haplotype blocks and append the results to combined_gwas_result.

Usage

```
extract_haplotype(combined_gwas_result = NULL,
  output_path = file.path("~"), Haploview_file_path = NULL,
  Haploview_file_name = NULL, Haploview_file_extension = NULL,
  Haploview_file_named_sequentially_from = NULL,
  Haploview_file_named_sequentially_to = NULL)
```

Arguments

output_path An output path.

Value

combined_gwas_result or NULL if something missing.

farming_with_GAPIT	<i>Farming with GAPIT</i>
--------------------	---------------------------

Description

The goal of farming_with_GAPIT is to run GWAS with GAPIT.

Usage

```
farming_with_GAPIT(dat = NULL, by_column = 1, start_column = 2,
  output_path, p_value_threshold = NA, p_value_fdr_threshold = NA,
  ld_number = 0, KI = NULL, CV = NULL, G = NULL, GD = NULL,
  GM = NULL, file.Ext.G = NULL, file.Ext.GD = NULL,
  file.Ext.GM = NULL, file.G = NULL, file.GD = NULL,
  file.GM = NULL, file.path = NULL, file.from = 0, file.to = 0,
  model = NULL, SNP.MAF = 0, PCA.total = 0,
  Model.selection = FALSE, SNP.test = FALSE, file.output = FALSE)
```

Arguments

dat An input dataset.
 by_column The accession column.
 start_column The start column index for traits.
 output_path An output path.

Value

combined_gwas_result or NULL if something missing.

generate_BLUE	<i>generate BLUE function</i>
---------------	-------------------------------

Description

The goal of generate_BLUE is to run Best Linear Unbiased Estimations.

Usage

```
generate_BLUE(dat = NULL, by_column = c(1, 2), start_column = 3)
```

Arguments

dat	An input dataset.
by_column	The accession column.
start_column	The start column index for traits.

Value

blue or NULL if something missing.

generate_BLUP	<i>generate BLUP function</i>
---------------	-------------------------------

Description

The goal of generate_BLUP is to run Best Linear Unbiased Predictions.

Usage

```
generate_BLUP(dat = NULL, by_column = c(1, 2), start_column = 3)
```

Arguments

dat	An input dataset.
by_column	The accession column.
start_column	The start column index for traits.

Value

blue or NULL if something missing.

search_genes	<i>Search genes from gff files</i>
--------------	------------------------------------

Description

The goal of search_genes is to append genes to combined_gwas_result.

Usage

```
search_genes(combined_gwas_result = NULL, output_path = file.path("~"),
             GFF_file_path = NULL, GFF_file_name = NULL,
             GFF_file_extension = NULL, GFF_file_named_sequentially_from = NULL,
             GFF_file_named_sequentially_to = NULL)
```

Arguments

output_path An output path.

Value

combined_gwas_result or NULL if something missing.

Index

- *Topic **BLUE**,
 - generate_BLUE, [3](#)
- *Topic **BLUP**,
 - generate_BLUP, [3](#)
- *Topic **GAPIT**,
 - farming_with_GAPIT, [2](#)
- *Topic **GWAS**
 - farming_with_GAPIT, [2](#)
- *Topic **Genes**
 - search_genes, [4](#)
- *Topic **Haplotype_blocks**
 - extract_haplotype, [2](#)
- *Topic **Pre-GWAS**
 - generate_BLUE, [3](#)
 - generate_BLUP, [3](#)
- [extract_haplotype, 2](#)
- [farming_with_GAPIT, 2](#)
- [generate_BLUE, 3](#)
- [generate_BLUP, 3](#)
- [search_genes, 4](#)