

# Package: PAST (via r-universe)

June 27, 2024

**Type** Package

**Title** Pathway Association Study Tool (PAST)

**Version** 1.21.0

**Description** PAST takes GWAS output and assigns SNPs to genes, uses those genes to find pathways associated with the genes, and plots pathways based on significance. Implements methods for reading GWAS input data, finding genes associated with SNPs, calculating enrichment score and significance of pathways, and plotting pathways.

**License** GPL (>=3) + file LICENSE

**Encoding** UTF-8

**Depends** R (>= 4.0)

**Imports** stats, utils, dplyr, rlang, iterators, parallel, foreach, doParallel, qvalue, rtracklayer, ggplot2, GenomicRanges, S4Vectors

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**RoxygenNote** 7.1.0

**URL** <https://github.com/IGBB/past>

**BugReports** <https://github.com/IGBB/past/issues>

**biocViews** Pathways, GeneSetEnrichment

**Repository** <https://bioc.r-universe.dev>

**RemoteUrl** <https://github.com/bioc/PAST>

**RemoteRef** HEAD

**RemoteSha** b99ea775a8198a89336d669e4f01719a950d9ea2

## Contents

assign_chunk . . . . .	2
assign_SNPs_to_genes . . . . .	2
determine_linkage . . . . .	3
find_pathway_significance . . . . .	4
find_representative_SNP . . . . .	5
find_representative_SNP_gene_pairing . . . . .	5
load_GWAS_data . . . . .	6
load_LD . . . . .	7
plot_pathways . . . . .	7
<b>Index</b>	<b>9</b>

---

assign_chunk	<i>Assign SNPs in a chunk to genes</i>
--------------	--

---

### Description

Assign SNPs in a chunk to genes

### Usage

```
assign_chunk(gff, chunk, window)
```

### Arguments

gff	The GFF data for the chromosome being parsed
chunk	The dataframe containing SNP data
window	The search window around the SNPs

### Value

tagSNPs labeled with gene names

---

assign_SNPs_to_genes	<i>Assign SNPs to genes</i>
----------------------	-----------------------------

---

### Description

Assign SNPs to genes

**Usage**

```
assign_SNPs_to_genes(  
  gwas_data,  
  LD,  
  gff_file,  
  filter_type,  
  window,  
  r_squared_cutoff,  
  num_cores  
)
```

**Arguments**

gwas_data	Merged association and effects data from merge_data()
LD	Linkage disequilibrium data from parse_LD()
gff_file	The path to a GFF file
window	The search window for genes around the SNP
r_squared_cutoff	The R <sup>2</sup> value used to determine SNP significance
num_cores	The number of cores to use in parallelizing PAST

**Value**

A dataframe of genes from the SNP data

**Examples**

```
example("load_GWAS_data")  
example("load_LD")  
demo_genes_file = system.file("extdata", "genes.gff",  
  package = "PAST", mustWork = TRUE)  
filter_type = c("gene")  
genes <- assign_SNPs_to_genes(gwas_data, LD, demo_genes_file, filter_type, 1000, 0.8, 2)
```

---

determine_linkage	<i>Determine Linkage</i>
-------------------	--------------------------

---

**Description**

Determine Linkage

**Usage**

```
determine_linkage(chunk, r_squared_cutoff)
```

**Arguments**

chunk                    A chunk of data to be processed  
 r\_squared\_cutoff        The R<sup>2</sup> value to check against

**Value**

Either the first unlinked SNP or a set of linked SNPs

---

*find\_pathway\_significance*  
*Find Pathway Significance*

---

**Description**

Find Pathway Significance

**Usage**

```
find_pathway_significance(  
  genes,  
  pathways_file,  
  gene_number_cutoff = 5,  
  mode,  
  sample_size = 1000,  
  num_cores  
)
```

**Arguments**

genes                    Genes from `assign_SNPs_to_genes()`  
 pathways\_file        A file containing the pathway IDs, their names, and the genes in the pathway  
 gene\_number\_cutoff    A cut-off for the minimum number of genes in a pathway  
 mode                    increasing/decreasing  
 sample\_size            How many times to sample the effects data during random sampling  
 num\_cores              The number of cores to use in parallelizing PAST

**Value**

Rugplots data

**Examples**

```
example("assign_SNPs_to_genes")
demo_pathways_file = system.file("extdata", "pathways.txt.xz",
  package = "PAST", mustWork = TRUE)
rugplots_data <- find_pathway_significance(genes, demo_pathways_file, 5,
  "increasing", 1000, 2)
```

---

find\_representative\_SNP

*Find representative SNP for a chunk of SNPs*

---

**Description**

Find representative SNP for a chunk of SNPs

**Usage**

```
find_representative_SNP(chunk, r_squared_cutoff)
```

**Arguments**

chunk                    A chunk of data to parse  
r\_squared\_cutoff        The R<sup>2</sup> value to check against when counting SNPs

**Value**

A single SNP representing the whole chunk

---

find\_representative\_SNP\_gene\_pairing

*Find the SNP-gene assignment that represents SNPs assigned to a gene*

---

**Description**

Find the SNP-gene assignment that represents SNPs assigned to a gene

**Usage**

```
find_representative_SNP_gene_pairing(chunk)
```

**Arguments**

chunk                    A chunk of gene assignments

**Value**

A single SNP-gene assignment representing all SNPS assigned to the same gene to a gene

---

load_GWAS_data	<i>Load GWAS data</i>
----------------	-----------------------

---

**Description**

Load GWAS data

**Usage**

```
load_GWAS_data(
  association_file,
  effects_file,
  association_columns = c("Trait", "Marker", "Locus", "Site", "p", "marker_R2"),
  effects_columns = c("Trait", "Marker", "Locus", "Site", "Effect")
)
```

**Arguments**

```
association_file
                The association file

effects_file    The effects file

association_columns
                The names of the columns in your association data for Trait, Marker, Chromo-
                some, Site, F, p, and marker_Rsquared

effects_columns
                The names of the columns in your effects data for Trait, Marker, Chromosome,
                Site, and effect
```

**Value**

The association data and the effects data merged into a dataframe with one row for each SNP

**Examples**

```
demo_association_file = system.file("extdata", "association.txt.xz",
  package = "PAST", mustWork = TRUE)
demo_effects_file = system.file("extdata", "effects.txt.xz",
  package = "PAST", mustWork = TRUE)
gwas_data <- load_GWAS_data(demo_association_file, demo_effects_file)
```

---

load_LD	<i>Load Linkage Disequilibrium</i>
---------	------------------------------------

---

**Description**

Load Linkage Disequilibrium

**Usage**

```
load_LD(  
  LD_file,  
  LD_columns = c("Locus1", "Position1", "Site1", "Position2", "Site2", "Dist_bp",  
                "R.2")  
)
```

**Arguments**

LD_file	The file containing linkage disequilibrium data
LD_columns	The names of the columns in your linkage disequilibrium data for the chromosome of the first SNP, the position of the first SNP, the site of the first SNP, the chromosome of the second SNP, the position of the second SNP, the site of the second SNP, the distance between the two SNPs, and the R.2

**Value**

The linkage disequilibrium data in a list containing dataframes for each chromosome.

**Examples**

```
demo_LD_file = system.file("extdata", "LD.txt.xz",  
  package = "PAST", mustWork = TRUE)  
LD <- load_LD(demo_LD_file)
```

---

plot_pathways	<i>Plot Rugplots for Selected Pathways</i>
---------------	--

---

**Description**

Plot Rugplots for Selected Pathways

**Usage**

```
plot_pathways(  
  rugplots_data,  
  filter_type,  
  filter_parameter,  
  mode,  
  output_directory  
)
```

**Arguments**

rugplots_data	The data to be plotted (returned from find_pathway_significance())
filter_type	The parameter to be used for filtering
filter_parameter	The cut-off value of the filtering parameter
mode	The mode used to create the data (increasing/decreasing)
output_directory	An existing directory to save results in

**Value**

Does not return a value

**Examples**

```
example("find_pathway_significance")  
plot_pathways(rugplots_data, "pvalue", "0.03", "decreasing", tempdir())
```



# Index

`assign_chunk`, [2](#)  
`assign_SNPs_to_genes`, [2](#)  
  
`determine_linkage`, [3](#)  
  
`find_pathway_significance`, [4](#)  
`find_representative_SNP`, [5](#)  
`find_representative_SNP_gene_pairing`,  
[5](#)  
  
`load_GWAS_data`, [6](#)  
`load_LD`, [7](#)  
  
`plot_pathways`, [7](#)