

# Package: LDWeaver (via r-universe)

September 19, 2026

**Type** Package

**Title** Genomewide Epistasis Analysis on Bacteria

**Version** 1.5.2

**Maintainer** Sudaraka Mallawaarachchi <smallawaarachchi@gmail.com>

**Description** Perform genomewide epistasis analysis by evaluating the LD structure in bacteria.

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

**Encoding** UTF-8

**LazyData** true

**biocViews** Software

**Depends** R (>= 4.0.0),

**Imports** ape, Biostrings, chromoMap, data.table, dplyr, fitdistrplus, GenomicRanges, GenomeInfoDb, ggplot2, ggraph, grDevices, heatmap3, htmlwidgets, igraph, IRanges, Matrix, MatrixExtra, methods, parallel, plyr, RColorBrewer, Rcpp, RcppArmadillo, S4Vectors, stats, utils, VariantAnnotation

**Suggests** phytools, ggtree, ggnewscale, spam, spam64, testthat (>= 3.0.0)

**LinkingTo** Rcpp, RcppArmadillo

**RoxygenNote** 7.3.2

**URL** <https://github.com/Sudaraka88/LDWeaver>

**BugReports** <https://github.com/Sudaraka88/LDWeaver/issues>

**Biarch** TRUE

**Config/testthat/edition** 3

**Config/pak/sysreqs** cmake libfontconfig1-dev libfreetype6-dev libglpk-dev make libbz2-dev libcud-dev liblzma-dev libpng-dev libuv1-dev libxml2-dev libssl-dev xz-utils zlib1g-dev

**Repository** <https://sudaraka88.r-universe.dev>

**Date/Publication** 2026-08-07 03:27:38 UTC

**RemoteUrl** <https://github.com/Sudaraka88/LDWeaver>

**RemoteRef** HEAD

**RemoteSha** 05b77d6221698ceffeb93b8fa36e8215266a46d1

## Contents

|                                             |    |
|---------------------------------------------|----|
| ._legendfun2 . . . . .                      | 3  |
| .mat . . . . .                              | 3  |
| .rescale01 . . . . .                        | 4  |
| analyse_long_range_links . . . . .          | 4  |
| cleanup . . . . .                           | 6  |
| cleanup_support . . . . .                   | 7  |
| create_network . . . . .                    | 7  |
| create_network_for_gene . . . . .           | 8  |
| create_tanglegram . . . . .                 | 9  |
| estimate_Hamming_distance_weights . . . . . | 10 |
| estimate_variation_in_CDS . . . . .         | 11 |
| GBAccession-class . . . . .                 | 12 |
| GenBankFile-class . . . . .                 | 12 |
| GenBankRecord-class . . . . .               | 12 |
| generate_Links_SNPS_fasta . . . . .         | 13 |
| genomewide_LDMap . . . . .                  | 13 |
| LDWeaver . . . . .                          | 14 |
| make_gwes_plots . . . . .                   | 17 |
| parse_fasta_alignment . . . . .             | 18 |
| parse_fasta_SNP_alignment . . . . .         | 19 |
| parse_genbank_file . . . . .                | 20 |
| parse_gff_file . . . . .                    | 21 |
| perform_MI_computation . . . . .            | 22 |
| perform_snpEff_annotations . . . . .        | 23 |
| read_AnnotatedLinks . . . . .               | 25 |
| read_GFF3_Annotation . . . . .              | 25 |
| read_LongRangeLinks . . . . .               | 26 |
| read_ReferenceFasta . . . . .               | 27 |
| read_ShortRangeLinks . . . . .              | 27 |
| read_TopHits . . . . .                      | 28 |
| runARACNE . . . . .                         | 28 |
| snpdat_to_fa . . . . .                      | 29 |
| view_tree . . . . .                         | 30 |
| write_output_for_gwes_explorer . . . . .    | 32 |

**Index**

**33**

---

|              |              |
|--------------|--------------|
| ._legendfun2 | ._legendfun2 |
|--------------|--------------|

---

**Description**

.\_legendfun2

**Usage**

```
._legendfun2(  
  legend = c("Group A", "Group B"),  
  lwd = 3,  
  cex = 1.1,  
  col = c("red", "blue"),  
  ...  
)
```

**Arguments**

|        |        |
|--------|--------|
| legend | legend |
| lwd    | lwd    |
| cex    | cex    |
| col    | col    |
| ...    | ...    |

---

|      |      |
|------|------|
| .mat | .mat |
|------|------|

---

**Description**

.mat

**Usage**

```
.mat(n, r)
```

**Arguments**

|   |                      |
|---|----------------------|
| n | number of dimensions |
| r | reduction factor     |

**Value**

kernel reduction

**Examples**

```
## Not run:  
.mat(10, 5)  
  
## End(Not run)
```

---

`.rescale01`*.rescale01*

---

**Description**`.rescale01`**Usage**`.rescale01(v)`**Arguments**

|                |                               |
|----------------|-------------------------------|
| <code>v</code> | vector to rescale between 0-1 |
|----------------|-------------------------------|

**Value**

rescaled vector

**Examples**

```
## Not run:  
.rescale01(sample(10,25,10))  
  
## End(Not run)
```

---

`analyse_long_range_links`*analyse\_long\_range\_links*

---

**Description**

Function to analyse long range links. These links can be computed using LDWeaver or Spydrpick (faster). Spydrpick is recommended for larger datasets.

**Usage**

```
analyse_long_range_links(
  dset,
  lr_links_path,
  sr_links_path,
  are_lrlinks_ordered = F,
  SnpEff_Annotate = F,
  snpeff_jar_path = NULL,
  gbk_path = NULL,
  gff3_path = NULL,
  ref_fasta_path = NULL,
  validate_ref_ann_lengths = T,
  snp.dat = NULL,
  cds_var = NULL,
  max_tophits = 500,
  links_from_spydrpick = F,
  mega_dset = F
)
```

**Arguments**

|                          |                                                                                                                                                                                                                   |
|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dset                     | name of the dataset, all outputs will be saved to the folder <dset>, possible to use the same dset name used for the short-range analysis                                                                         |
| lr_links_path            | path to saved LR links, generally called <lr_links.tsv> or the SpydrPick edges (links) file                                                                                                                       |
| sr_links_path            | path to saved SR links, generally called <sr_links.tsv>                                                                                                                                                           |
| are_lrlinks_ordered      | are the links sorted in descending MI order (default = F), leave at default if unsure                                                                                                                             |
| SnpEff_Annotate          | specify whether to perform annotations using SnpEff (default = F)                                                                                                                                                 |
| snpeff_jar_path          | path to <snpeff.jar> (default = NULL). If annotations are not required, set SnpEff_Annotate = F                                                                                                                   |
| gbk_path                 | path to genbank annotations file (default = NULL). Only provide one of genbank or gff3 annotation files.                                                                                                          |
| gff3_path                | path to gff3 annotations file (default = NULL). Only provide one of genbank or gff3 annotation files.                                                                                                             |
| ref_fasta_path           | path to Reference fasta file. The file MUST be in fasta format and contain exactly one sequence! Required for gff3 annotations, not required for genbank annotations if the file contains the reference sequence. |
| validate_ref_ann_lengths | check if the gbk reference sequence length matches the with fasta alignment (default = T)                                                                                                                         |
| snp.dat                  | output from LDWeaver::parse_fasta_alignment(), not needed if SnpEff_Annotate = F                                                                                                                                  |

|                                   |                                                                                                                                                                                                                                 |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>cds_var</code>              | output from LDWeaver::estimate_variation_in_CDS(), not needed if SnpEff_Annotate = F                                                                                                                                            |
| <code>max_tophits</code>          | specify the maximum number of long range links to save as <lr_tophits.tsv>. Note: all short-range links will be annotated (and saved separately), but only the top <max_tophits> will be used for visualisation (default = 500) |
| <code>links_from_spydrpick</code> | are the links computed using spydrpick (default = F)                                                                                                                                                                            |
| <code>mega_dset</code>            | set TRUE for mega scale datasets (default = F)                                                                                                                                                                                  |

### Examples

```
## Not run:
analyse_long_range_links = function("dset", "dset/lr_links.tsv", "dset/sr_links.tsv")

## End(Not run)
```

---

cleanup

*cleanup*

---

### Description

Function to clean up the files in <dset> folder, this will organise the saved files in to a more sensible folder structure. If clean up was performed already, nothing will be changed.

### Usage

```
cleanup(dset, delete_after_moving = F)
```

### Arguments

|                                  |                                                                                                                                                                       |
|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>dset</code>                | <dset> folder that requires cleanup.                                                                                                                                  |
| <code>delete_after_moving</code> | specify whether the organised files should be deleted from the original location (default = F). If False, all organised files will be moved to a directory called OLD |

### Examples

```
## Not run:
gff3_path = "<path_to_gff3_annotation>"
gff = read_GFF3_Annotation(gff3_path)

## End(Not run)
```

---

|                 |                        |
|-----------------|------------------------|
| cleanup_support | <i>cleanup_support</i> |
|-----------------|------------------------|

---

**Description**

File operations for the cleanup function

**Usage**

```
cleanup_support(files, fldr)
```

**Arguments**

|       |                      |
|-------|----------------------|
| files | files to copy <from> |
| fldr  | folder to copy <to>  |

**Value**

parsed annotation file in list format

**Examples**

```
## Not run:  
gff3_path = "<path_to_gff3_annotation">  
gff = read_GFF3_Annotation(gff3_path)  
  
## End(Not run)
```

---

|                |                       |
|----------------|-----------------------|
| create_network | <i>create_network</i> |
|----------------|-----------------------|

---

**Description**

Function to generate a network view of top GWES hits

**Usage**

```
create_network(  
  tophits,  
  netplot_path = NULL,  
  plot_title = NULL,  
  separator = ":",  
  max_plot_nodes = NULL,  
  plot_w = 6000,  
  plot_h = 4000,  
  min_links_to_include = 2  
)
```

**Arguments**

|                      |                                                                                                                                                                        |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tophits              | data frame with top short or long range GWES links, returned from LDWeaver::perform_snpEff_annotation                                                                  |
| netplot_path         | folder to save the network (default = NULL, return ggplot object)                                                                                                      |
| plot_title           | title for the network plot (default = NULL)                                                                                                                            |
| separator            | break pattern in annotation (default = ":", this is the default used by SnpEff)                                                                                        |
| max_plot_nodes       | specify the maximum number of nodes to visualise in the plot (default = NULL, use all). Large values will result in a cluttered output, consider increasing plot size. |
| plot_w               | specify plot width in pixels (default = 6000)                                                                                                                          |
| plot_h               | specify plot height in pixels (default = 4000)                                                                                                                         |
| min_links_to_include | specify the minimum number of links required to retain the node in the plot (default = 2)                                                                              |

**Value**

Network plots (ggplot object)

**Examples**

```
## Not run:
create_network(tophits, netplot_path)

## End(Not run)
```

---

```
create_network_for_gene
      create_network_for_gene
```

---

**Description**

Function to generate a network view of GWES hits for a given gene

**Usage**

```
create_network_for_gene(
  gene_name,
  sr_annotated_path = NULL,
  lr_annotated_path = NULL,
  drop_syXsy = T,
  drop_indirect = T,
  level = 1,
  separator = ":",
  min_links_to_include = 3
)
```



**Arguments**

|                   |                                                                                                                                              |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| tophits           | data frame with top short range GWES links, returned from LDWeaver::perform_snpEff_annotations()                                             |
| gbk               | output from parsing the genbank file using LDWeaver::parse_genbank_file() (default = NULL), only provide either GFF3 or GBK annotation       |
| gff               | output from parsing the gff3 file using LDWeaver::parse_gff_file() (default = NULL), only provide either GFF3 or GBK annotation              |
| tanglegram_folder | folder to save tanglegram(s)                                                                                                                 |
| break_segments    | specify the number of genome segments to prepare - one tanglegram per segment (default = 5)                                                  |
| links_type        | specify the links type long-range "LR" or short-range "SR" (default = "SR"). A tanglegram may not be suitable to visualise long range links! |

**Value**

none

**Examples**

```
## Not run:
gbk = LDWeaver::parse_genbank_file("path_to_genbank_file", length_check = F)
create_tanglegram(tophits, gbk, "save_folder")

## End(Not run)
```

---

```
estimate_Hamming_distance_weights
```

```
estimate_Hamming_distance_weights
```

---

**Description**

Function to estimate the similarity between sequences based on the Hamming distance

**Usage**

```
estimate_Hamming_distance_weights(snp.dat, threshold = 0.1, mega_dset = F)
```

**Arguments**

|           |                                                                                       |
|-----------|---------------------------------------------------------------------------------------|
| snp.dat   | output from parsing the multi fasta alignment using LDWeaver::parse_fasta_alignment() |
| threshold | Hamming distance similarity threshold (default = 0.1, i.e. 10%)                       |
| mega_dset | set to TRUE for mega scale datasets (default = F)                                     |

**Value**

Vector with Hamming distance weights for each sequence that can be used to reweight Mutual Information estimates

**Examples**

```
## Not run:
hdw <- estimate_Hamming_distance_weights(snp.dat, threshold=0.1)

## End(Not run)
```

---

```
estimate_variation_in_CDS
      estimate_variation_in_CDS
```

---

**Description**

Function to estimate the variation within each coding region, the output from this function can be used to segment the genome into diversity-based clusters.

**Usage**

```
estimate_variation_in_CDS(
  snp.dat,
  ncores,
  gbk = NULL,
  gff = NULL,
  num_clusts_CDS = 3,
  clust_plt_path = NULL,
  mega_dset = FALSE
)
```

**Arguments**

|                |                                                                                       |
|----------------|---------------------------------------------------------------------------------------|
| snp.dat        | output from parsing the multi fasta alignment using LDWeaver::parse_fasta_alignment() |
| ncores         | specify the number of cores to use                                                    |
| gbk            | output from parsing the genbank file using LDWeaver::parse_genbank_file()             |
| gff            | output from parsing the gff3 file using LDWeaver::parse_gff_file()                    |
| num_clusts_CDS | partition to genome into num_clusts_CDS regions using k-means (default = 3)           |
| clust_plt_path | specify path to save CDS variation plot                                               |
| mega_dset      | set to TRUE for mega scale datasets (default = F)                                     |

**Value**

R list with CDS variation and allele distribution details

Examples

```
## Not run:
cds_var <- estimate_variation_in_CDS(gbk, snp.dat, ncores = 10)

## End(Not run)
```

---

|                   |                             |
|-------------------|-----------------------------|
| GBAccession-class | <i>GBAccession ID class</i> |
|-------------------|-----------------------------|

---

Description

A class representing the (versioned) GenBank accession

Usage

```
GBAccession(id)
```

Arguments

id                    A versioned GenBank Accession id

Value

a GBAccession object.

---

|                   |                     |
|-------------------|---------------------|
| GenBankFile-class | <i>GenBank File</i> |
|-------------------|---------------------|

---

Description

A resource class for use within the rtracklayer framework

---

|                     |                              |
|---------------------|------------------------------|
| GenBankRecord-class | <i>GenBank data objects.</i> |
|---------------------|------------------------------|

---

Description

These objects represent GenBank annotations

---

```
generate_Links_SNPS_fasta
    generate_Links_SNPS_fasta
```

---

### Description

Create a link SNP only fasta file. This file is required to generate (detailed) tree plots

### Usage

```
generate_Links_SNPS_fasta(
  snp.dat,
  aln_path,
  pos_path,
  lr_tophits_path = NULL,
  lr_annotated_links_path = NULL,
  sr_tophits_path = NULL,
  sr_annotated_links_path = NULL
)
```

### Arguments

|                         |                                                                                                                                            |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| snp.dat                 | output from parsing the multi fasta alignment using LDWeaver::parse_fasta_alignment(), generally saved to Additional_Outputs/snp_ACGTN.rds |
| aln_path                | path to save output fasta file                                                                                                             |
| pos_path                | path to write the SNP positions                                                                                                            |
| lr_tophits_path         | path to the lr_tophits file (typically Tophits/lr_tophits.tsv).                                                                            |
| lr_annotated_links_path | path to the lr_annotated_links file (typically Annotated_links/lr_links_annotated.tsv).                                                    |
| sr_tophits_path         | path to the sr_tophits file (typically Tophits/sr_tophits.tsv).                                                                            |
| sr_annotated_links_path | path to the sr_annotated_links file (typically Annotated_links/sr_links_annotated.tsv).                                                    |

---

```
genomewide_LDMap    genomewide_LDMap
```

---

### Description

Function to generate the bird's-eye view of the genomewide LD structure (measured as weighted-MI)

**Usage**

```

genomewide_LDMap(
  lr_links_path,
  sr_links_path,
  plot_save_path,
  reducer = NULL,
  plot_title = NULL,
  links_from_spydrpick = F,
  from = NULL,
  to = NULL
)

```

**Arguments**

|                                   |                                                                                                                                                                                                                 |
|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>lr_links_path</code>        | path to saved <lr_links.tsv> or spydrpick edges (links) file                                                                                                                                                    |
| <code>sr_links_path</code>        | path to saved SR links, generally called <sr_links.tsv>                                                                                                                                                         |
| <code>plot_save_path</code>       | path to save the LD Map                                                                                                                                                                                         |
| <code>reducer</code>              | positive value used to reduce the LDMap size (default = NULL, will use length(positions)/1e3). The default paramter should work well, reducing the parameter will increase resolution at the cost of resources. |
| <code>plot_title</code>           | title for the LD Map (default = NULL)                                                                                                                                                                           |
| <code>links_from_spydrpick</code> | are the links computed using spydrpick (default = F)                                                                                                                                                            |
| <code>from</code>                 | draw the plot within the region starting <from> (default = NULL)                                                                                                                                                |
| <code>to</code>                   | draw the plot within the region ending in <to> (default = NULL)                                                                                                                                                 |

**Examples**

```

## Not run:
genomewide_LDMap(lr_links_path, sr_links_path, plot_save_path)

## End(Not run)

```

---

LDWeaver

*LDWeaver*


---

**Description**

Function to run the LDWeaver pipeline

**Usage**

```
LDWeaver(
  dset,
  aln_path,
  aln_has_all_bases = T,
  pos = NULL,
  gbk_path = NULL,
  gff3_path = NULL,
  ref_fasta_path = NULL,
  validate_ref_ann_lengths = T,
  snp_filt_method = "default",
  gap_freq = 0.15,
  maf_freq = 0.01,
  hdw_threshold = 0.1,
  perform_SR_analysis_only = F,
  SnpEff_Annotate = T,
  sr_dist = 20000,
  lr_retain_links = 1e+06,
  max_tophits = 250,
  num_clusts_CDS = 3,
  srp_cutoff = 3,
  tanglegram_break_segments = 5,
  write_gwesExplorer = T,
  multicore = T,
  max_blk_sz = 10000,
  ncores = NULL,
  save_additional_outputs = F,
  mega_dset = F
)
```

**Arguments**

|                   |                                                                                                                                                                                                                                       |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dset              | name of the dataset, all outputs will be saved to the folder <dset>                                                                                                                                                                   |
| aln_path          | path to the multi fasta alignment - can be a SNP only or full alignment                                                                                                                                                               |
| aln_has_all_bases | specify whether the alignment has all bases in the reference genome (default = T). For a SNP only alignment, set to F and provide SNP positions in pos (see example below)                                                            |
| pos               | numeric vector of positions for each base in the alignment (default = NULL). Only required for SNP alignments (e.g. Output from snp-sites or FastaR <a href="https://github.com/Sudaraka88/Fa">https://github.com/Sudaraka88/Fa</a> ) |
| gbk_path          | path to genbank annotations file (default = NULL). Only provide one of genbank or gff3                                                                                                                                                |
| gff3_path         | path to gff3 annotations file (default = NULL). Only provide one of genbank or gff3                                                                                                                                                   |
| ref_fasta_path    | path to Reference fasta file. This must be in fasta format and contain exactly one sequence! Only required for gff3 annotations                                                                                                       |

|                           |                                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| validate_ref_ann_lengths  | check if the reference sequence length matches with fasta alignment. (default = T for full alignments, = F for snp-only alignments.). Set to F for alignments that only contain a portion of the genome                                                                                                                                                                |
| snp_filt_method           | specify SNP filtering method: 'relaxed' or 'default' (default = 'default'). Unlike default, relaxed considers ambiguous/gap characters (N) as minor alleles when applying the maf_freq filter. Eg: Under default filter values, a site with allele frequencies A:0.85, C:0.0095, N:0.1405 will be respectively, dropped and allowed by 'default' and 'relaxed'         |
| gap_freq                  | sites with a gap frequency >gap_freq will be dropped (default = 0.15)                                                                                                                                                                                                                                                                                                  |
| maf_freq                  | sites with a minor allele frequency <maf_freq will be dropped (default = 0.01)                                                                                                                                                                                                                                                                                         |
| hdw_threshold             | Hamming distance similarity threshold (default = 0.1, i.e. 10%) - lower values will force stricter population structure control at the cost of masking real signal                                                                                                                                                                                                     |
| perform_SR_analysis_only  | specify whether to only perform LDWeaver (short-range link) analysis. (default = FALSE) will analyse all links (i.e., LDWeaver + SpydrPick)                                                                                                                                                                                                                            |
| SnEff_Annotate            | specify whether to perform annotations using SnpEff (default = TRUE)                                                                                                                                                                                                                                                                                                   |
| sr_dist                   | links shorter than <sr_dist> apart are considered 'short range' (default = 20000), range 1000 - 25000 bp                                                                                                                                                                                                                                                               |
| lr_retain_links           | specify the maximum number of long-range MI links to retain (default = 1000000) - in each block, only a top subset of links will be saved such approximately this many links will be retained                                                                                                                                                                          |
| max_tophits               | specify the maximum number of short range links to save as <sr_tophits.tsv>. Note: all short-range links will be annotated (and saved separately), but only the top <max_tophits> will be used for visualisation (default = 250), range 50 - 1000                                                                                                                      |
| num_clusts_CDS            | partition to genome into num_clusts_CDS regions using k-means (default = 3) range 1 - 10 - accounts for within genome heterogeneity in the short-range analysis                                                                                                                                                                                                        |
| srp_cutoff                | specify the short-range -log10(p) cut-off value to discard short-range links before returning the data.frame (default = 3, i.e., $p = 10^{-3}$ ). This setting has no impact on modelling since all links are used. Larger values will reduce memory usage, plotting time and ARACNE run time. If all links are required, set to 0 (i.e. $p = 10^0 = 1$ ), range 0 - 5 |
| tanglegram_break_segments | specify the number of genome tanglegram segments to prepare (default = 5 segments). Set NULL to skip tanglegram. range 1 - 10.                                                                                                                                                                                                                                         |
| write_gwesExplorer        | specify whether to write output for GWESExplorer (default = T)                                                                                                                                                                                                                                                                                                         |
| multicore                 | specify whether to use parallel processing (default = T)                                                                                                                                                                                                                                                                                                               |
| max_blk_sz                | specify maximum block size for MI computation (default = 10000), larger sizes require more RAM, range 1000 - 100000                                                                                                                                                                                                                                                    |

|                         |                                                                                                                                                                              |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ncores                  | specify the number of cores to use for parallel processing. Auto detect (detect = NULL)                                                                                      |
| save_additional_outputs | specify whether to save outputs such as extracted SNPs and Hamming distance weights. Recommended for very large datasets to save time on re-computation (default = F)        |
| mega_dset               | specify whether the datasets is megascale. This mode requires spam and spam64 packages. This is >5 times slower, set to TRUE only if the normal analysis fails (default = F) |

### Value

All generated outputs will be saved to folder <dset>.

### Examples

```
## Not run:
# These examples use data included with the package.

# Example 1 - using a full alignment that includes non SNP sites
dset <- "full_dset"
gbk_path <- system.file("extdata", "sample.gbk", package = "LDWeaver")
aln_path <- system.file("extdata", "sample.aln.gz", package = "LDWeaver")
LDWeaver::LDWeaver(dset = dset, aln_path = aln_path, gbk_path = gbk_path,
  validate_ref_ann_lengths = F)

# Example 2- using a SNP only alignment
dset <- "snp_dset"
gbk_path <- system.file("extdata", "sample.gbk", package = "LDWeaver")
aln_path <- system.file("extdata", "snp_sample.fa.gz", package = "LDWeaver")
pos <- as.numeric(readLines(system.file("extdata", "snp_sample.fa.pos",
  package = "LDWeaver"))))
LDWeaver::LDWeaver(dset = dset, aln_path = aln_path, aln_has_all_bases = F,
  pos = pos, gbk_path = gbk_path)

dset <- "full_dset_spam"
gbk_path <- system.file("extdata", "sample.gbk", package = "LDWeaver")
aln_path <- system.file("extdata", "sample.aln.gz", package = "LDWeaver")
LDWeaver::LDWeaver(dset = dset, aln_path = aln_path, gbk_path = gbk_path,
  validate_ref_ann_lengths = F, mega_dset = T)

## End(Not run)
```

---

make\_gwes\_plots

make\_gwes\_plots

---

### Description

Function to generate long and short range GWES plots

**Usage**

```
make_gwes_plots(
  lr_links = NULL,
  sr_links = NULL,
  plt_folder = NULL,
  are_srlinks_ordered = F
)
```

**Arguments**

|                                  |                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>lr_links</code>            | data.frame containing <code>lr_links</code> or the path to saved <code>lr_links</code> TSV file. Usually output from <code>perform_MI_computation()</code> , (default = <code>NULL</code> ). Instead of using this function to generate the <code>lr-gwes</code> plot, use the dedicated <code>LDWeaver::analyse_long_range_links()</code> function for long range link analysis (and plotting). |
| <code>sr_links</code>            | data.frame containing <code>sr_links</code> or the path to saved <code>sr_links</code> TSV file. Usually output from <code>perform_MI_computation()</code> , (default = <code>NULL</code> )                                                                                                                                                                                                      |
| <code>plt_folder</code>          | specify the folder to save generated plots (default = <code>NULL</code> , will be saved to a folder called <code>PLOTS</code> in <code>getwd()</code> )                                                                                                                                                                                                                                          |
| <code>are_srlinks_ordered</code> | if <code>False</code> , links will be ordered before plotting, which will plot higher <code>srp</code> links on top of lower ones (default = <code>F</code> ). Set <code>True</code> only if the links are already in the preferred order. Note: This function will reverse the link order to match the <code>ggplot</code> 's top to bottom plotting order.                                     |

**Examples**

```
## Not run:
sr_links_red = perform_MI_computation(snp.dat = snp.dat, hdw = hdw, cds_var = cds_var, ncores = 10,
lr_save_path = "lr.tsv", plt_folder = "plots")

## End(Not run)
```

---

`parse_fasta_alignment` *extractSNP\_from\_MSA*

---

**Description**

Function to extract SNPs from a multi-fasta alignment.

**Usage**

```
parse_fasta_alignment(
  aln_path,
  gap_freq = 0.15,
  maf_freq = 0.01,
```

```

    method = "default",
    mega_dset = F
  )

```

### Arguments

|           |                                                                                |
|-----------|--------------------------------------------------------------------------------|
| aln_path  | path to multi fasta alignment                                                  |
| gap_freq  | sites with a gap frequency >gap_greq will be dropped (default = 0.15)          |
| maf_freq  | sites with a minor allele frequency <maf_freq will be dropped (default = 0.01) |
| method    | specify the filtering method 'relaxed' or 'default' (default = 'default')      |
| mega_dset | set to TRUE for mega scale datasets (default = F)                              |

### Value

R list with SNPs in sparse format and additional parameters

### Examples

```

## Not run:
aln_path <- system.file("extdata", "sample.aln", package = "LDWeaver")
snp.dat <- parseFastaAlignment(aln_path, gap_freq = 0.15, maf_freq = 0.01, method = "default")

## End(Not run)

```

---

```

parse_fasta_SNP_alignment
      extractSNPs_from_SNP_MSA

```

---

### Description

Function to extract SNPs from a SNP only multi-fasta alignment. This will be the output from running snp-sites.

### Usage

```

parse_fasta_SNP_alignment(
  aln_path,
  pos,
  gap_freq = 0.15,
  maf_freq = 0.01,
  method = "default",
  mega_dset = F
)

```

**Arguments**

|           |                                                                                |
|-----------|--------------------------------------------------------------------------------|
| aln_path  | path to multi fasta SNP alignment                                              |
| pos       | numeric vector containing positions in the original alignment                  |
| gap_freq  | sites with a gap frequency >gap_freq will be dropped (default = 0.15)          |
| maf_freq  | sites with a minor allele frequency <maf_freq will be dropped (default = 0.01) |
| method    | specify the filtering method 'relaxed' or 'default' (default = 'default')      |
| mega_dset | set to TRUE for mega scale datasets (default = F)                              |

**Value**

R list with SNPs in sparse format and additional parameters

**Examples**

```
## Not run:
aln_path <- system.file("extdata", "sample.aln", package = "LDWeaver")
snp.dat <- parseFastaAlignment(aln_path, gap_freq = 0.15, maf_freq = 0.01, method = "default")

## End(Not run)
```

---

|                    |                           |
|--------------------|---------------------------|
| parse_genbank_file | <i>parse_genbank_file</i> |
|--------------------|---------------------------|

---

**Description**

Function to parse the genbank file for the fasta alignment. Parser built using genbankr source available here: <https://github.com/gmbecker/genbankr>.

**Usage**

```
parse_genbank_file(gbk_path, g = NULL, length_check = T)
```

**Arguments**

|              |                                                                                                                               |
|--------------|-------------------------------------------------------------------------------------------------------------------------------|
| gbk_path     | path to genbank file                                                                                                          |
| g            | sequence length, available from the LDWeaver::parse_fasta_alignment() output (default = NULL), required if <length_check = T> |
| length_check | specify whether to check if fasta and gbk sequence lengths are equal (default = T)                                            |

**Value**

GenBankRecord object

**Examples**

```
## Not run:
aln_path <- system.file("extdata", "sample.aln", package = "LDWeaver")
snp.dat <- parseFastaAlignment(aln_path, gap_freq = 0.15, maf_freq = 0.01, method = "default")
gbk_path <- system.file("extdata", "sample.gbk", package = "LDWeaver")
gbk <- parse_genbank_file(gbk_path, snp.dat$g)

## End(Not run)
```

---

|                |                       |
|----------------|-----------------------|
| parse_gff_file | <i>parse_gff_file</i> |
|----------------|-----------------------|

---

**Description**

Function to parse the gff3 file for the fasta alignment.

**Usage**

```
parse_gff_file(gff3_path, ref_fasta_path, perform_length_check = T)
```

**Arguments**

|                      |                                                                                                         |
|----------------------|---------------------------------------------------------------------------------------------------------|
| gff3_path            | path to gff3_annotation file. This function relies on <code>ape::read.gff()</code> to read in the file  |
| ref_fasta_path       | path to Reference fasta file. The file <b>MUST</b> be in fasta format and contain exactly one sequence! |
| perform_length_check | specify whether to check if the gff3 annotations agree with the reference length (default = T)          |

**Value**

gff annotation and reference sequence as a list object

**Examples**

```
## Not run:
gff3_path <- "<path_to_gff3_file>"
ref_fasta_path <- "<path_to_reference_fasta>"
ref_gff <- parse_gff_file(gff3_path, ref_fasta_path)

## End(Not run)
```

---

```
perform_MI_computation
```

```
perform_MI_computation
```

---

## Description

Function to compute pairwise MI (long range links saved to text file), model the background LD levels within each cluster (plots saved) and compute short-range p-values for each short-range links. If requested, this function can also run the ARACNE algorithm

## Usage

```
perform_MI_computation(
  snp.dat,
  hdw,
  cds_var,
  ncores,
  lr_save_path = NULL,
  sr_save_path = NULL,
  plt_folder = NULL,
  sr_dist = 20000,
  lr_retain_links = 1e+06,
  max_blk_sz = 10000,
  srp_cutoff = 3,
  runARACNE = TRUE,
  perform_SR_analysis_only = FALSE,
  order_links = T,
  mega_dset = F
)
```

## Arguments

|                           |                                                                                                                                                           |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>snp.dat</code>      | output from parsing the multi fasta alignment using <code>LDWeaver::parse_fasta_alignment()</code>                                                        |
| <code>hdw</code>          | vector of Hamming distance weights, output from <code>LDWeaver::estimate_Hamming_distance_weights()</code>                                                |
| <code>cds_var</code>      | output from <code>LDWeaver::estimate_variation_in_CDS()</code>                                                                                            |
| <code>ncores</code>       | specify the number of cores to use                                                                                                                        |
| <code>lr_save_path</code> | specify the location to save long range MI links as a tsv file (default = NULL, will be auto set)                                                         |
| <code>sr_save_path</code> | specify the location to save short range MI links as a tsv file (default = NULL, will be auto set), links below <code>srp_cutoff</code> will not be saved |
| <code>plt_folder</code>   | specify the folder to save generated plots (default = NULL, will be saved to a folder called PLOTS in <code>getwd()</code> )                              |
| <code>sr_dist</code>      | specify the short-range basepair separation (default = 20000)                                                                                             |

|                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| lr_retain_links          | specify the maximum number of long-range MI links to retain (default = 1000000)<br>- in each block, only a top subset of links will be saved                                                                                                                                                                                                                                                                                                       |
| max_blk_sz               | specify maximum block size for MI computation (default = 10000), larger sizes require more RAM                                                                                                                                                                                                                                                                                                                                                     |
| srp_cutoff               | specify the short-range $-\log_{10}(p)$ cut-off value to discard short-range links before returning the data.frame. This setting has no impact on the modelling since all links are used. However, setting a threshold $> 2$ will generally reduce the memory usage, plotting time (default = 3, i.e. corresponding to $p = 0.001$ ), and run time for ARACNE. If all links are required to be returned, set to 0 (i.e. corresponding to $p = 1$ ) |
| runARACNE                | specify whether to run ARACNE on short-range links (default = TRUE), if set to FAULT, all links will be marked as ARACNE=1                                                                                                                                                                                                                                                                                                                         |
| perform_SR_analysis_only | skip the long range link analysis (default = FALSE)                                                                                                                                                                                                                                                                                                                                                                                                |
| order_links              | return and save links after sorting in short-range p-value order, most to least important (default = T)                                                                                                                                                                                                                                                                                                                                            |
| mega_dset                | set TRUE for mega scale datasets (default = F)                                                                                                                                                                                                                                                                                                                                                                                                     |

**Value**

R data frame with short range GWES links (plots and long range links will be saved)

**Examples**

```
## Not run:
sr_links_red = perform_MI_computation(snp.dat = snp.dat, hdw = hdw, cds_var = cds_var, ncores = 10,
lr_save_path = "lr.tsv", plt_folder = "plots")

## End(Not run)
```

---

```
perform_snpEff_annotations
      perform_snpEff_annotations
```

---

**Description**

Function to generate long and short range GWES plots

**Usage**

```
perform_snpEff_annotations(
  dset_name,
  annotation_folder,
  snpeff_jar,
```

```

    snp.dat,
    cds_var,
    links_df,
    gbk = NULL,
    gbk_path = NULL,
    gff = NULL,
    tophits_path = NULL,
    max_tophits = 250,
    links_type = "SR"
  )

```

### Arguments

|                   |                                                                                                                                        |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| dset_name         | name of the dataset                                                                                                                    |
| annotation_folder | folder to save annotations                                                                                                             |
| snpeff_jar        | path to <snpeff.jar>                                                                                                                   |
| snp.dat           | output from parsing the multi fasta alignment using LDWeaver::parse_fasta_alignment()                                                  |
| cds_var           | output from LDWeaver::estimate_variation_in_CDS()                                                                                      |
| links_df          | data.frame containing links (reduced links can be used to save time) output from perform_MI_computation()                              |
| gbk               | output from parsing the genbank file using LDWeaver::parse_genbank_file() (default = NULL), only provide either GFF3 or GBK annotation |
| gbk_path          | path to genbank file - plan to omit (default = NULL)                                                                                   |
| gff               | output from parsing the gff3 file using LDWeaver::parse_gff_file() (default = NULL), only provide either GFF3 or GBK annotation        |
| tophits_path      | specify the path to save tophit links (default = NULL). If NULL, will be saved to annotations_folder/tophits.tsv                       |
| max_tophits       | specify the number of top hits to return (all links will be annotated and saved to the annotations_folder)                             |
| links_type        | specify the links type long-range "LR" or short-range "SR" (default = "SR")                                                            |

### Value

R data frame with the top short range GWES links with snpEff annotated embedded

### Examples

```

## Not run:
sr_links_red = perform_snpEff_annotations(dset_name, annotation_folder, snpeff_jar, gbk,
gbk_path, cds_var, sr_links, snp.dat, max_tophits = 250)

## End(Not run)

```

---

|                     |                            |
|---------------------|----------------------------|
| read_AnnotatedLinks | <i>read_AnnotatedLinks</i> |
|---------------------|----------------------------|

---

**Description**

read\_AnnotatedLinks

**Usage**

```
read_AnnotatedLinks(annotated_links_path)
```

**Arguments**

annotated\_links\_path  
path to saved <sr> or <lr>\_annotated\_links.tsv file

**Value**

Annotated links as a data.frame

**Examples**

```
## Not run:  
sr_annotated_links_path = "<path_to_annotated_sr_links.tsv_file>"  
sr_links_ann = read_AnnotatedLinks(sr_annotated_links_path)  
  
## End(Not run)
```

---

|                      |                             |
|----------------------|-----------------------------|
| read_GFF3_Annotation | <i>read_GFF3_Annotation</i> |
|----------------------|-----------------------------|

---

**Description**

read\_GFF3\_Annotation

**Usage**

```
read_GFF3_Annotation(gff3_path)
```

**Arguments**

gff3\_path            path to gff3\_annotation file. This function relies on ape::read.gff() to read in the file, refer to ?ape::read.gff for additional help

**Value**

parsed annotation file in list format

**Examples**

```
## Not run:
gff3_path = "<path_to_gff3_annotation>"
gff = read_GFF3_Annotation(gff3_path)

## End(Not run)
```

---

|                     |                            |
|---------------------|----------------------------|
| read_LongRangeLinks | <i>read_LongRangeLinks</i> |
|---------------------|----------------------------|

---

**Description**

read\_LongRangeLinks

**Usage**

```
read_LongRangeLinks(lr_links_path, links_from_spydrpick = F, sr_dist = 20000)
```

**Arguments**

|                      |                                                                                       |
|----------------------|---------------------------------------------------------------------------------------|
| lr_links_path        | path to saved lr_links.tsv or spydrpick file                                          |
| links_from_spydrpick | are the links computed using spydrpick (default = F)                                  |
| sr_dist              | links less than <sr_dist> apart will be dropped from the data.frame (default = 20000) |

**Value**

Long range links as a data.frame

**Examples**

```
## Not run:
lr_links_path = "<path_to_lrlinks.tsv_file>"
lr_links = read_LongRangeLinks(lr_links_path)

## End(Not run)
```

---

|                     |                            |
|---------------------|----------------------------|
| read_ReferenceFasta | <i>read_ReferenceFasta</i> |
|---------------------|----------------------------|

---

**Description**

read\_ReferenceFasta

**Usage**

read\_ReferenceFasta(ref\_fasta\_path)

**Arguments**

ref\_fasta\_path path to Reference fasta file. The file MUST be in fasta format and contain exactly one sequence!

**Value**

parsed reference sequence with metadata in list format

**Examples**

```
## Not run:  
ref_fa = "<path_to_ref_fasta"  
reference = read_ReferenceFasta(ref_fa)  
  
## End(Not run)
```

---

|                      |                             |
|----------------------|-----------------------------|
| read_ShortRangeLinks | <i>read_ShortRangeLinks</i> |
|----------------------|-----------------------------|

---

**Description**

read\_ShortRangeLinks

**Usage**

read\_ShortRangeLinks(sr\_links\_path)

**Arguments**

sr\_links\_path path to saved sr\_links.tsv file

**Value**

Long range links as a data.frame

Examples

```
## Not run:
sr_links_path = "<path_to_sr_links.tsv_file>"
sr_links = read_ShortRangeLinks(sr_links_path)

## End(Not run)
```

---

|              |                     |
|--------------|---------------------|
| read_TopHits | <i>read_TopHits</i> |
|--------------|---------------------|

---

Description

read\_TopHits

Usage

```
read_TopHits(top_hits_path)
```

Arguments

top\_hits\_path    path to saved top\_hits file

Value

Top hits as a data.frame

Examples

```
## Not run:
top_hits_path = "<path_to_tophits_file>"
tophits = read_TopHits(top_hits_path)

## End(Not run)
```

---

|           |                  |
|-----------|------------------|
| runARACNE | <i>runARACNE</i> |
|-----------|------------------|

---

Description

runARACNE

Usage

```
runARACNE(links_to_check, links_full)
```

**Arguments**

links\_to\_check   data.frame comprising subset of links to run ARACNE on  
 links\_full       data.frame with all links

**Value**

Long range links as a data.frame

**Examples**

```
## Not run:
sr_links_path = "<path_to_sr_links.tsv_file>"
sr_links = readShortRangeLinks(sr_links_path)
sr_links_red = sr_links[which(sr_links$srp > 3), ]
sr_links_red$ARACNE = runARACNE(sr_links_red, sr_links)

## End(Not run)
```

---

|              |                     |
|--------------|---------------------|
| snpdat_to_fa | <i>snpdat_to_fa</i> |
|--------------|---------------------|

---

**Description**

Create a SNP only fasta or tsv file from snp.dat. The user has the option to include only chosen positions.

**Usage**

```
snpdat_to_fa(snp.dat, aln_path, pos_path = NULL, pos = NULL, format = "fasta")
```

**Arguments**

|          |                                                                                                                                                                                                   |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| snp.dat  | output from parsing the multi fasta alignment using LDWeaver::parse_fasta_alignment(), generally saved to Additional_Outputs/snp_ACGTN.rds                                                        |
| aln_path | path to save output fasta file or tsv file                                                                                                                                                        |
| pos_path | path to write the SNP positions, only required for format = "fasta" (default = NULL)                                                                                                              |
| pos      | set of loci to be included in the saved alignment (default = NULL - save everything). <pos> should be a numeric vector comprising a subset of sites in snp.dat\$POS and cannot contain duplicates |
| format   | Output can be presented as a standard fasta file, or an optional tsv (default = "fasta")                                                                                                          |

view\_tree

*view\_tree*

## Description

Function to visualise allele distributions and metadata aligned below the tree. The tree file must be readable with `ape::read.tree()` and `<tip.labels>` must match the sequence names in the provided fasta file(s). Besides the tree file, this function requires minimum three inputs: (1) a links file or data.frame, (2) a SNP fasta alignment and (3) a position `<text>` file with each row containing the genomic positions of the fasta alignment. The links file is straightforward: it can be either one of `<Tophits>` or the `<Annotated_links>` files, both short and long range links can be provided. A subset of manually chosen links can also be provided as a data.frame. The fasta/position files can be generated using `LDWeaver::generate_Links_SNPS_fasta()`. If metadata is available, they can be visualised along with the allele data in a separate panel.

## Usage

```
view_tree(
  tree_path,
  perform_midpoint_rooting = T,
  metadata_df = NULL,
  fasta_path,
  pos_file_path,
  links_df = NULL,
  lr_tophits_path = NULL,
  lr_annotated_links_path = NULL,
  sr_tophits_path = NULL,
  sr_annotated_links_path = NULL,
  ntop_links = 10,
  from = NULL,
  to = NULL,
  offset_metadata = NULL,
  offset_alleles = NULL,
  width_metadata = NULL,
  width_alleles = NULL,
  plot_save_path = NULL,
  plot_height = 20,
  plot_width = 15
)
```

## Arguments

|                                       |                                                                                                                                                                           |
|---------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>tree_path</code>                | Path to tree file (must be readable using <code>ape::read.tree()</code> ). <code>tip.labels</code> must exactly match the sequence names in the fasta and metadata files. |
| <code>perform_midpoint_rooting</code> | Specify wheter the tree should be midpoint rooted (default = T)                                                                                                           |

|                         |                                                                                                                                                                                                                                                                                                                         |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| metadata_df             | data.frame with metadata. Must contain a column labelled 'id' that exactly matches the sequence names in the fasta file and tip.labels of the tree.                                                                                                                                                                     |
| fasta_path              | fasta file comprising the sites associated with links                                                                                                                                                                                                                                                                   |
| pos_file_path           | position file containing fasta sites                                                                                                                                                                                                                                                                                    |
| links_df                | A data.frame comprising a subset of tophits links, if provided, links are not read from the files. NOTE: Each links must (at a minimum) contain the fields: <pos1> & <pos2>. If alleles from the <ntop_links> are to be visualised, ensure that the links_df is sorted such that the most important link is at the top. |
| lr_tophits_path         | Path to the lr_tophits file (typically Tophits/lr_tophits.tsv).                                                                                                                                                                                                                                                         |
| lr_annotated_links_path | Path to the lr_annotated_links file (typically Annotated_links/lr_links_annotated.tsv).                                                                                                                                                                                                                                 |
| sr_tophits_path         | Path to the sr_tophits file (typically Tophits/sr_tophits.tsv).                                                                                                                                                                                                                                                         |
| sr_annotated_links_path | Path to the sr_annotated_links file (typically Annotated_links/sr_links_annotated.tsv).                                                                                                                                                                                                                                 |
| ntop_links              | Specify the number of links to plot from tophits file(s). ntop_links will be selected from each file if both LR and SR links should be visualised (default = 10).                                                                                                                                                       |
| from                    | draw the SNPs that are within the region starting <from> and <to> below (default = NULL). SNPs that are also in links with SNPs from the chosen region will be plotted.                                                                                                                                                 |
| to                      | If <from> is provided, <to> muts also be provided and <ntop_links> will be ignored (default = NULL).                                                                                                                                                                                                                    |
| offset_metadata         | specify the separation between tree and metadata layer. These parameters should be modified to tune the plot dimensions as required. (default = NULL)                                                                                                                                                                   |
| offset_alleles          | specify the speciation between tree and alleles layer (default = NULL)                                                                                                                                                                                                                                                  |
| width_metadata          | specify the width of the metadata layer as a proportion of the tree size (default = NULL)                                                                                                                                                                                                                               |
| width_alleles           | specify the width of the alleles layer as a proportion of the tree size (default = NULL)                                                                                                                                                                                                                                |
| plot_save_path          | path to save the tree plot (default = NULL, show plot instead of saving)                                                                                                                                                                                                                                                |
| plot_height             | if saving, specify the plot height in inches (default = 20)                                                                                                                                                                                                                                                             |
| plot_width              | if saving, specify the plot height in inches (default = 15). Modify these numbers to tune the plot dimensions as required.                                                                                                                                                                                              |

## Examples

```
## Not run:
genomewide_LDMap(lr_links_path, sr_links_path, plot_save_path)

## End(Not run)
```

---

```
write_output_for_gwes_explorer  
    write_output_for_gwes_explorer
```

---

## Description

Function to generate long and short range GWES plots

## Usage

```
write_output_for_gwes_explorer(  
  snp.dat,  
  tophits,  
  gwes_explorer_folder,  
  links_type = "SR"  
)
```

## Arguments

|                      |                                                                                                  |
|----------------------|--------------------------------------------------------------------------------------------------|
| snp.dat              | output from parsing the multi fasta alignment using LDWeaver::parse_fasta_alignment()            |
| tophits              | data frame with top short range GWES links, returned from LDWeaver::perform_snpEff_annotations() |
| gwes_explorer_folder | folder to save the inputs for GWESExplorer                                                       |
| links_type           | specify the links type long-range "LR" or short-range "SR" (default = "SR")                      |

## Value

none

## Examples

```
## Not run:  
write_output_for_gwes_explorer(snp.dat, tophits, gwes_explorer_folder)  
  
## End(Not run)
```

# Index

[.\\_legendfun2](#), 3  
[.mat](#), 3  
[.rescale01](#), 4  
  
[analyse\\_long\\_range\\_links](#), 4  
  
[cleanup](#), 6  
[cleanup\\_support](#), 7  
[create\\_network](#), 7  
[create\\_network\\_for\\_gene](#), 8  
[create\\_tanglegram](#), 9  
  
[estimate\\_Hamming\\_distance\\_weights](#), 10  
[estimate\\_variation\\_in\\_CDS](#), 11  
  
[GBAccession](#) ([GBAccession-class](#)), 12  
[GBAccession-class](#), 12  
[GBFile-class](#) ([GenBankFile-class](#)), 12  
[GBKFile-class](#) ([GenBankFile-class](#)), 12  
[GenBankFile-class](#), 12  
[GenBankRecord-class](#), 12  
[generate\\_Links\\_SNPS\\_fasta](#), 13  
[genomewide\\_LDMap](#), 13  
  
[LDWeaver](#), 14  
  
[make\\_gwes\\_plots](#), 17  
  
[parse\\_fasta\\_alignment](#), 18  
[parse\\_fasta\\_SNP\\_alignment](#), 19  
[parse\\_genbank\\_file](#), 20  
[parse\\_gff\\_file](#), 21  
[perform\\_MI\\_computation](#), 22  
[perform\\_snpEff\\_annotations](#), 23  
  
[read\\_AnnotatedLinks](#), 25  
[read\\_GFF3\\_Annotation](#), 25  
[read\\_LongRangeLinks](#), 26  
[read\\_ReferenceFasta](#), 27  
[read\\_ShortRangeLinks](#), 27  
[read\\_TopHits](#), 28  
  
[runARACNE](#), 28  
  
[snpdatt\\_to\\_fa](#), 29  
  
[view\\_tree](#), 30  
  
[write\\_output\\_for\\_gwes\\_explorer](#), 32