

Package: locuscomparer (via r-universe)

June 2, 2026

Type Package

Title locuscomparer: Visualization of GWAS-QTL Colocalization Events

Version 1.0.0

Description locuscomparer is the standalone version of the LocusCompare webserver (www.locuscompare.ml:3838). locuscomparer performs visualization of colocalization of summary statistics from pairs of association datasets (e.g. GWAS and eQTL).

License GPL-3

Depends R (>= 2.10)

Imports cowplot (>= 0.6.1), DBI (>= 1.0.0), ggplot2 (>= 2.2.1), ggrepel (>= 0.6.5), RMySQL (>= 0.10.17)

Suggests testthat

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Config/pak/sysreqs libmysqlclient-dev

Repository <https://mrcieu.r-universe.dev>

Date/Publication 2025-08-06 08:01:56 UTC

RemoteUrl <https://github.com/remlapmot/locuscomparer>

RemoteRef suggestions

RemoteSha 208d7fae39096ec599e940faee9c9fc31f450085

Contents

add_label	2
assign_color	2
config	3
get_lead_snp	4
get_position	4

locuscompare	5
make_combined_plot	6
make_locuszoom	7
make_scatterplot	8
read_metal	10
retrieve_LD	10

Index	12
--------------	-----------

add_label	<i>Add a column of SNP labels to input data.frame</i>
-----------	---

Description

Add a column of SNP labels to input data.frame

Usage

```
add_label(merged, snp)
```

Arguments

merged	(data.frame) Input data.frame, which is a result by merging two association studies. See the example under 'get_lead_snp()' for generation of such data.frame.
snp	(character vector) A vector of SNP rsIDs. If only labeling one SNP, this can also be a single string.

Examples

```
# Select the lead SNP
in_fn_1 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn_1, marker_col = 'rsid', pval_col = 'pval')
in_fn_2 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d2 = read_metal(in_fn_2, marker_col = 'rsid', pval_col = 'pval')
merged = merge(d1, d2, by = "rsid", suffixes = c("1", "2"), all = FALSE)
merged = add_label(merged, 'rs9349379')
```

assign_color	<i>Assign color to each SNP according to LD.</i>
--------------	--

Description

Assign color to each SNP according to LD.

Usage

```
assign_color(rsid, snp, ld)
```

Arguments

rsid	(character vector) A vector of rsIDs on which to assign color.
snp	(string) rsID for lead SNP. This SNP will be colored purple. Other SNPs will be assigned color based on their LD with the lead SNP.
ld	(data.frame) The output from 'retrieve_LD()'.

Examples

```
# Select the lead SNP
in_fn_1 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn_1, marker_col = 'rsid', pval_col = 'pval')
in_fn_2 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d2 = read_metal(in_fn_2, marker_col = 'rsid', pval_col = 'pval')
merged = merge(d1, d2, by = "rsid", suffixes = c("1", "2"), all = FALSE)
ld <- retrieve_LD('6', 'rs9349379', 'AFR')
color = assign_color(rsid = merged$rsid, snp = 'rs9349379', ld)
```

config

Example dataset

Description

Example dataset

Usage

config

Format

'who' A data frame with 1 row and 3 columns:

a "URL"

b "locuscomparer"

c "12345678"

get_lead_snp	<i>Get the lead SNP from the list of SNPs in input data.frame The lead SNP is defined as the SNP with the lowest sum of p-values from the two studies.</i>
--------------	--

Description

Get the lead SNP from the list of SNPs in input data.frame The lead SNP is defined as the SNP with the lowest sum of p-values from the two studies.

Usage

```
get_lead_snp(merged, snp = NULL)
```

Arguments

merged	(data.frame) Input data.frame, which is a result by merging two association studies.
snp	(string, optional) SNP rsID. If NULL, the function will select the lead SNP based on the sum of p-values from the two studies. If an rsID is supplied, the function will simply return the rsID.

Examples

```
# Select the lead SNP
in_fn_1 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn_1, marker_col = 'rsid', pval_col = 'pval')
in_fn_2 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d2 = read_metal(in_fn_2, marker_col = 'rsid', pval_col = 'pval')
merged = merge(d1, d2, by = "rsid", suffixes = c("1", "2"), all = FALSE)
get_lead_snp(merged)
```

get_position	<i>Append two columns, chromosome (chr) and position (pos), to the input data.frame.</i>
--------------	--

Description

Append two columns, chromosome (chr) and position (pos), to the input data.frame.

Usage

```
get_position(x, genome = c("hg19", "hg38"))
```

Arguments

`x` (data.frame) Input data.frame.
`genome` (string, optional) Genome assembly, either 'hg19' or 'hg38'. Default: 'hg19'.

Examples

```
in_fn = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn, marker_col = 'rsid', pval_col = 'pval')
get_position(d1)
```

locuscompare *Make a locuscompare plot.*

Description

Make a locuscompare plot.

Usage

```
locuscompare(
  in_fn1,
  in_fn2,
  marker_col1 = "rsid",
  pval_col1 = "pval",
  title1 = "eQTL",
  marker_col2 = "rsid",
  pval_col2 = "pval",
  title2 = "GWAS",
  snp = NULL,
  population = "EUR",
  combine = TRUE,
  legend = TRUE,
  legend_position = c("bottomright", "topright", "topleft"),
  lz_ylab_linebreak = FALSE,
  genome = c("hg19", "hg38")
)
```

Arguments

`in_fn1` (string) Path to the input file for study 1.
`in_fn2` (string) Path to the input file for study 2.
`marker_col1` (string, optional) Name of the marker column. Default: 'rsid'.
`pval_col1` (string, optional) Name of the p-value column. Default: 'pval'.
`title1` (string) The title for the x-axis.
`marker_col2` (string, optional) Name of the marker column. Default: 'rsid'.

pval_col2	(string, optional) Name of the p-value column. Default: 'pval'.
title2	(string) The title for the y-axis.
snp	(string, optional) SNP rsID. If NULL, the function will select the lead SNP. Default: NULL.
population	(string, optional) One of the 5 populations from 1000 Genomes: 'AFR', 'AMR', 'EAS', 'EUR', and 'SAS'. Default: 'EUR'.
combine	(boolean, optional) Should the three plots be combined into one plot? If FALSE, a list of three plots will be returned. Default: TRUE.
legend	(boolean, optional) Should the legend be shown? Default: TRUE.
legend_position	(string, optional) Either 'bottomright', 'topright', or 'topleft'. Default: 'bottomright'.
lz_ylab_linebreak	(boolean, optional) Whether to break the line of y-axis of the locuszoom plot.
genome	(string, optional) Genome assembly, either 'hg19' or 'hg38'. Default: 'hg19'.

Examples

```
in_fn1 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
in_fn2 = system.file('extdata', 'eqtl.tsv', package = 'locuscomparer')
locuscompare(in_fn1 = in_fn1, in_fn2 = in_fn2)
```

make_combined_plot	<i>Generated a combined plot with two locuszoom plots and a locuscompare plot. Each locuszoom plot represent an association study.</i>
--------------------	--

Description

Generated a combined plot with two locuszoom plots and a locuscompare plot. Each locuszoom plot represent an association study.

Usage

```
make_combined_plot(
  merged,
  title1,
  title2,
  ld,
  chr,
  snp = NULL,
  combine = TRUE,
  legend = TRUE,
  legend_position = c("bottomright", "topright", "topleft"),
  lz_ylab_linebreak = FALSE
)
```

Arguments

merged	(data.frame) An input data.frame which has the following columns: rsid, pval1 (p-value for study 1), logp1 (p-value for study 2), logp1 (log p-value for study 1), logp2 (log p-value for study 2), chr, pos. See the example for 'get_lead_snp()' on how to generate this data.frame.
title1	(string) The title for the x-axis.
title2	(string) The title for the y-axis.
ld	(data.frame) The output from 'retrieve_LD()'.
chr	(string) Chromosome name. e.g. '22'. Notice that the name should not contain 'chr'.
snp	(string, optional) SNP rsID. If NULL, the function will select the lead SNP. Default: NULL.
combine	(boolean, optional) Should the three plots be combined into one plot? If FALSE, a list of three plots will be returned. Default: TRUE.
legend	(boolean, optional) Should the legend be shown? Default: TRUE.
legend_position	(string, optional) Either 'bottomright', 'topright', or 'topleft'. Default: 'bottom-right'.
lz_ylab_linebreak	(boolean, optional) Whether to break the line of y-axis of the locuszoom plot. If FALSE, the y-axis title and '-log10(p-value)'. will be on the same line. Default: FALSE.

Examples

```
## Not run:
# Select the lead SNP
in_fn_1 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn_1, marker_col = 'rsid', pval_col = 'pval')
in_fn_2 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d2 = read_metal(in_fn_2, marker_col = 'rsid', pval_col = 'pval')
merged = merge(d1, d2, by = "rsid", suffixes = c("1", "2"), all = FALSE)
merged = add_label(merged, 'rs9349379')
ld <- retrieve_LD('6', 'rs9349379', 'AFR')
make_combined_plot(merged, 'GWAS', 'eQTL', ld, chr)

## End(Not run)
```

make_locuszoom

Make a locuszoom plot. Details see <http://locuszoom.org/>.

Description

Make a locuszoom plot. Details see <http://locuszoom.org/>.

Usage

```
make_locuszoom(metal, title, chr, color, shape, size, ylab_linebreak = FALSE)
```

Arguments

metal (data.frame) input file with two column, rsID and p-value. See 'read_metal()' for more details.

title (string) y-axis title.

chr (string) chromosome.

color (data.frame) The output from 'assign_color()'.
shape (data.frame) Specification of the shape of each SNP. See example blow on how to generate this data.frame.

size (data.frame) Specification of the size of each SNP. See example below on how to generate this data.frame.

ylab_linebreak (boolean, optional) Whether to break the line of y-axis. If FALSE, the y-axis title and '-log10(p-value)' will be on the same line. Default: FALSE.

Examples

```
## Not run:
# Select the lead SNP
in_fn_1 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn_1, marker_col = 'rsid', pval_col = 'pval')
in_fn_2 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d2 = read_metal(in_fn_2, marker_col = 'rsid', pval_col = 'pval')
merged = merge(d1, d2, by = "rsid", suffixes = c("1", "2"), all = FALSE)
merged = add_label(merged, 'rs9349379')
ld <- retrieve_LD('6', 'rs9349379', 'AFR')
color = assign_color(rsid = merged$rsid, snp = 'rs9349379', ld)
snp = 'rs9349379'
shape = ifelse(merged$rsid == snp, 23, 21)
names(shape) = merged$rsid
size = ifelse(merged$rsid == snp, 3, 2)
names(size) = merged$rsid
chr = '6'
d1 = get_position(d1)
make_locuszoom(d1, title = 'GWAS', chr, color, shape, size)

## End(Not run)
```

make_scatterplot

Make a scatter plot (called the LocusCompare plot). Each axis of the LocusCompare plot represent the -log10(p-value) from an association study. Each point thus represent a SNP. By default, the lead SNP is a purple diamond, whereas the other SNPs are colored according to their LD with the lead SNP.

Description

Make a scatter plot (called the LocusCompare plot). Each axis of the LocusCompare plot represent the $-\log_{10}(\text{p-value})$ from an association study. Each point thus represent a SNP. By default, the lead SNP is a purple diamond, whereas the other SNPs are colored according to their LD with the lead SNP.

Usage

```
make_scatterplot(
  merged,
  title1,
  title2,
  color,
  shape,
  size,
  legend = TRUE,
  legend_position = c("bottomright", "topright", "topleft")
)
```

Arguments

merged	(data.frame) An input data.frame which has the following columns: rsid, pval1 (p-value for study 1), logp1 (p-value for study 2), logp1 (log p-value for study 1), logp2 (log p-value for study 2), chr, pos. See the example for 'get_lead_snp()' on how to generate this data.frame.
title1	(string) The title for the x-axis.
title2	(string) The title for the y-axis.
color	(data.frame) The output from 'assign_color()'.
shape	(data.frame) Specification of the shape of each SNP. See example blow on how to generate this data.frame.
size	(data.frame) Specification of the size of each SNP. See example below on how to generate this data.frame.
legend	(boolean) Whether to include the legend.
legend_position	(string, optional) Either 'bottomright', 'topright', or 'topleft'. Default: 'bottom-right'.

Examples

```
# Select the lead SNP
in_fn_1 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn_1, marker_col = 'rsid', pval_col = 'pval')
in_fn_2 = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d2 = read_metal(in_fn_2, marker_col = 'rsid', pval_col = 'pval')
merged = merge(d1, d2, by = "rsid", suffixes = c("1", "2"), all = FALSE)
merged = add_label(merged, 'rs9349379')
ld <- retrieve_LD('6', 'rs9349379', 'AFR')
```

```

color = assign_color(rsid = merged$rsid, snp = 'rs9349379', ld)
snp = 'rs9349379'
shape = ifelse(merged$rsid == snp, 23, 21)
names(shape) = merged$rsid
size = ifelse(merged$rsid == snp, 3, 2)
names(size) = merged$rsid
make_scatterplot(merged, title1 = 'GWAS', title2 = 'eQTL', color, shape, size)

```

read_metal	<i>Read association summary statistics from file and append column. The file must contain 2 columns: markers, i.e SNPs, and p-value. The marker column should contain SNP rsIDs.</i>
------------	--

Description

Read association summary statistics from file and append column. The file must contain 2 columns: markers, i.e SNPs, and p-value. The marker column should contain SNP rsIDs.

Usage

```
read_metal(in_fn, marker_col = "rsid", pval_col = "pval")
```

Arguments

in_fn	(string) Path to the input file.
marker_col	(string, optional) Name of the marker column. Default: 'rsid'.
pval_col	(string, optional) Name of the p-value column. Default: 'pval'.

Examples

```

in_fn = system.file('extdata', 'gwas.tsv', package = 'locuscomparer')
d1 = read_metal(in_fn, marker_col = 'rsid', pval_col = 'pval')

```

retrieve_LD	<i>Retrive SNP pairwise LD from database. SNP pairwise ID are calculated based on 1000 Genomes Project Phase 3 version 5. For storage-efficiency, the output will only include SNPs with $r^2 > 0.2$ with the input SNP.</i>
-------------	--

Description

Retrive SNP pairwise LD from database. SNP pairwise ID are calculated based on 1000 Genomes Project Phase 3 version 5. For storage-efficiency, the output will only include SNPs with $r^2 > 0.2$ with the input SNP.

Usage

```
retrieve_LD(chr, snp, population)
```

Arguments

chr	(string) Chromosome name. e.g. '22'. Notice that the name should not contain 'chr'.
snp	(string) SNP rsID.
population	(string) One of the 5 populations from 1000 Genomes: 'AFR', 'AMR', 'EAS', 'EUR', and 'SAS'.

Examples

```
retrieve_LD('6', 'rs9349379', 'AFR')
```

Index

* datasets

config, 3

add_label, 2

assign_color, 2

config, 3

get_lead_snp, 4

get_position, 4

locuscompare, 5

make_combined_plot, 6

make_locuszoom, 7

make_scatterplot, 8

read_metal, 10

retrieve_LD, 10