

Package ‘GPlinksR’

August 1, 2026

Type Package

Title Building Gene-Peak Network for ATAC-RNA Integration

Version 0.99.2

Description GPlinksR constructs gene-peak regulatory networks for ATAC-RNA integration by combining enhancer-based, promoter-based, and proximity (closest-gene) mappings. The package accepts direct peak and gene vectors as well as container-based inputs through a wrapper for common Bioconductor object classes. Enhancer-gene links are obtained from the PEREGRINE enhancer-gene datasets provided by AnnoQ, while promoter and gene coordinates are retrieved from EnsDb.Hsapiens.v86.

License MIT + file LICENSE

URL <https://github.com/Corawang123/GPlinksR>

BugReports <https://github.com/Corawang123/GPlinksR/issues>

Encoding UTF-8

Depends R (>= 4.6.0)

Imports BiocFileCache, data.table, GenomicRanges, GenomeInfoDb, IRanges, methods, MultiAssayExperiment, S4Vectors, EnsDb.Hsapiens.v86, ensemblDb, biomaRt, dplyr, SingleCellExperiment, SummarizedExperiment

biocViews GeneExpression, Network, Sequencing, Transcriptomics

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

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

VignetteBuilder knitr

Config/testthat/edition 3

git_url <https://git.bioconductor.org/packages/GPlinksR>

git_branch devel

git_last_commit 6a7555b

git_last_commit_date 2026-07-14

Repository Bioconductor 3.24

Date/Publication 2026-07-31

Author Xinran Wang [aut, cre] (ORCID: <<https://orcid.org/0009-0000-1805-3280>>),
Kelly Street [ctb],
Huaiyu Mi [ctb],
Bryan Queme [ctb]

Maintainer Xinran Wang <xwang210@usc.edu>

Contents

GPLinksR-package	2
build_gp_links	2
build_gp_links_wrapper	4
get_peregrine_file	5
gp_example_inputs	6
gp_example_links	7
Index	8

GPLinksR-package	<i>GPLinksR: Gene-Peak Regulatory Network Construction</i>
------------------	--

Description

GPLinksR constructs gene-peak regulatory networks for ATAC-RNA integration. It combines enhancer-gene links, promoter overlaps, and closest-gene mappings, and accepts either direct feature vectors or common Bioconductor container objects.

Main functions

- build_gp_links() constructs links from peak coordinates and gene symbols.
- build_gp_links_wrapper() extracts inputs from Bioconductor containers.
- get_peregrine_file() downloads and caches PEREGRINE enhancer-gene data.

build_gp_links	<i>Build Gene-Peak Network (Enhancer, Promoter, Closest)</i>
----------------	--

Description

Build Gene-Peak Network (Enhancer, Promoter, Closest)

Usage

```
build_gp_links(  
  pk,  
  gn,  
  ver = 19,  
  enh_file = NULL,  
  enh_coords_file = NULL,  
  gene_map = NULL,  
  mart = NULL  
)
```

Arguments

pk	Either: (1) A character vector of genomic coordinates ("chr1:1000-1200"), or (2) A data.frame containing peak coordinates with columns "chr","start","end" or a "PeakRegion" column ("chr1:1000-1200" style).
gn	Character vector of gene symbols (HGNC official symbols, e.g. "TP53", "FMNL2"). Each input peak is linked to the nearest transcript TSS among these genes.
ver	Integer (17, 18, or 19) - PANTHER enhancer-gene link version.
enh_file	Optional local path to enhancer-gene link file (.tsv).
enh_coords_file	Optional local path to the corresponding PEREGRINE enhancer-coordinate file. The file must contain chromosome, start, end, and enhancer identifier columns without a header. If NULL, the hg38 PEREGRINE coordinate file is downloaded or retrieved from cache.
gene_map	Optional data.frame with columns hgnc_symbol and hgnc_id. When supplied, this local mapping is used instead of querying BioMart.
mart	Optional biomaRt::Mart object used for gene identifier mapping. If NULL, the default Ensembl human genes mart is used. Supply an object created by biomaRt::useEnsembl() to select a mirror or custom host.

Value

A data.frame with columns: Peak, Gene, Src. Closest-gene mappings assign each peak center to its single nearest transcript TSS across the genes in gn. When a peak-gene pair has multiple evidence sources, each source is returned as a separate row.

Examples

```
data("gp_example_inputs", package = "GPlinksR")
pk <- gp_example_inputs$pk[seq_len(3)]
gn <- gp_example_inputs$gn[seq_len(4)]

enh_file <- system.file(
  "extdata", "gp_example_enhancer_links.tsv",
  package = "GPlinksR", mustWork = TRUE
)
enh_coords_file <- system.file(
  "extdata", "gp_example_enhancer_coordinates.tsv",
  package = "GPlinksR", mustWork = TRUE
)
gene_map <- data.frame(
  hgnc_symbol = c("TTLL10", "SAMD11"),
  hgnc_id = c("HGNC:26693", "HGNC:28706")
)

gp <- build_gp_links(
  pk, gn,
  enh_file = enh_file,
  enh_coords_file = enh_coords_file,
  gene_map = gene_map
)
gp

if (interactive()) {
```

```

mirror_mart <- biomaRt::useEnsembl(
  biomaRt = "genes",
  dataset = "hsapiens_gene_ensembl",
  mirror = "useast"
)
gp <- build_gp_links(pk, gn, mart = mirror_mart)
head(gp)
}

```

build_gp_links_wrapper

Build Gene-Peak Links from MAE or SCE Input

Description

Wrapper around `build_gp_links()` that extracts peak coordinates and gene symbols from a `MultiAssayExperiment` or `SingleCellExperiment`, then runs the standard `GPlinksR` workflow.

Usage

```

build_gp_links_wrapper(
  x,
  peak_experiment = NULL,
  gene_experiment = NULL,
  peak_col = NULL,
  gene_col = NULL,
  ver = 19,
  enh_file = NULL,
  enh_coords_file = NULL,
  gene_map = NULL,
  mart = NULL
)

```

Arguments

<code>x</code>	A <code>MultiAssayExperiment</code> , <code>SingleCellExperiment</code> , <code>SummarizedExperiment</code> , or <code>RangedSummarizedExperiment</code> .
<code>peak_experiment</code>	For <code>MultiAssayExperiment</code> , the experiment name holding peak features. For <code>SingleCellExperiment</code> , the <code>altExp()</code> name holding peak features.
<code>gene_experiment</code>	For <code>MultiAssayExperiment</code> , the experiment name holding gene features. Ignored for <code>SingleCellExperiment</code> , where the main experiment is used for genes.
<code>peak_col</code>	Optional <code>rowData()</code> column containing peak coordinates in "chr:start-end" format.
<code>gene_col</code>	Optional <code>rowData()</code> column containing gene symbols.
<code>ver</code>	Integer (17, 18, or 19). PEREGRINE enhancer-gene link version.
<code>enh_file</code>	Optional local path to enhancer-gene link file (.tsv).
<code>enh_coords_file</code>	Optional local path to the corresponding PEREGRINE enhancer-coordinate file passed to <code>build_gp_links()</code> .

gene_map	Optional data.frame with columns hgnc_symbol and hgnc_id passed to build_gp_links().
mart	Optional biomaRt::Mart object passed to build_gp_links(). Supply an object created with biomaRt::useEnsembl() to use an Ensembl mirror or custom host.

Details

Peak features are extracted from rowRanges(), rowData(), or row names. Gene symbols are extracted from rowData() or row names.

Value

A data.frame with columns returned by build_gp_links().

Examples

```
if (requireNamespace("SingleCellExperiment", quietly = TRUE) &&
    requireNamespace("SummarizedExperiment", quietly = TRUE)) {
  sce <- SingleCellExperiment::SingleCellExperiment(
    assays = list(counts = matrix(seq_len(12), nrow = 3)),
    rowData = S4Vectors::DataFrame(symbol = c("TP53", "GATA1", "SPI1"))
  )

  SingleCellExperiment::altExp(sce, "ATAC") <-
    SummarizedExperiment::SummarizedExperiment(
      assays = list(counts = matrix(seq_len(12), nrow = 3)),
      rowData = S4Vectors::DataFrame(
        PeakRegion = c(
          "chr1:1000-1200",
          "chr1:5000-5200",
          "chr2:9000-9300"
        )
      )
    )

  sce
  # Example wrapper call:
  # build_gp_links_wrapper(
  #   x = sce, peak_experiment = "ATAC", gene_col = "symbol"
  # )
}
```

get_peregrine_file	<i>Download Enhancer-Gene Link File from PANTHER Peregrine Database</i>
--------------------	---

Description

Download Enhancer-Gene Link File from PANTHER Peregrine Database

Usage

```
get_peregrine_file(version = 19)
```

Arguments

version Integer (17, 18, or 19). Which PANTHER enhancer-gene link version to download.

Value

The full path to the downloaded .tsv file stored in the BiocFileCache.

Examples

```
if (interactive()) {  
  f <- get_peregrine_file(19)  
  f  
  readLines(f, n = 3)  
}
```

gp_example_inputs	<i>Example Gene and Peak Inputs for GPlinksR</i>
-------------------	--

Description

A small example dataset for demonstrating GPlinksR workflows. The object contains 300 peak coordinates and 100 gene symbols selected from a KPMP-derived kidney single-cell analysis. The first few entries yield enhancer-, promoter-, and closest-based links in the executable examples.

Usage

```
data(gp_example_inputs)
```

Format

A list with two elements:

pk A character vector of 300 peak coordinates in "chr:start-end" format.

gn A character vector of 100 gene symbols.

Examples

```
data(gp_example_inputs)  
length(gp_example_inputs$pk)  
length(gp_example_inputs$gn)
```

gp_example_links*Example Link Table for the Packaged GPlinksR Demo Subset*

Description

A reference gene-peak link table corresponding to the leading entries in gp_example_inputs. The table contains the expected enhancer-, promoter-, and closest-based links from the package's executable offline example.

Usage

```
data(gp_example_links)
```

Format

A data.frame with 7 rows and 3 columns:

Peak Peak coordinate in "chr:start-end" format.

Gene Gene symbol linked to the peak.

Src Link source, one of "enh", "prom", or "clo".

Examples

```
data(gp_example_links)
head(gp_example_links)
table(gp_example_links$Src)
```

Index

* **internal**

GPLinksR-package, [2](#)

build_gp_links, [2](#)

build_gp_links_wrapper, [4](#)

get_peregrine_file, [5](#)

gp_example_inputs, [6](#)

gp_example_links, [7](#)

GPLinksR (GPLinksR-package), [2](#)

GPLinksR-package, [2](#)