

# Package ‘cageminer’

August 19, 2025

**Title** Candidate Gene Miner

**Version** 1.15.0

**Description** This package aims to integrate GWAS-derived SNPs and coexpression networks to mine candidate genes associated with a particular phenotype. For that, users must define a set of guide genes, which are known genes involved in the studied phenotype. Additionally, the mined candidates can be given a score that favor candidates that are hubs and/or transcription factors. The scores can then be used to rank and select the top n most promising genes for downstream experiments.

**License** GPL-3

**URL** <https://github.com/almeidasilvaf/cageminer>

**BugReports** <https://support.bioconductor.org/t/cageminer>

**biocViews** Software, SNP, FunctionalPrediction, GenomeWideAssociation, GeneExpression, NetworkEnrichment, VariantAnnotation, FunctionalGenomics, Network

**Encoding** UTF-8

**LazyData** false

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.3

**Imports** ggplot2, rlang, ggbio, ggtext, GenomeInfoDb, GenomicRanges, IRanges, reshape2, methods, BioNERO

**Depends** R (>= 4.1)

**Suggests** testthat (>= 3.0.0), SummarizedExperiment, knitr, BiocStyle, rmarkdown, covr, sessioninfo

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/cageminer>

**git\_branch** devel

**git\_last\_commit** 8c85060

**git\_last\_commit\_date** 2025-04-15

**Repository** Bioconductor 3.22

**Date/Publication** 2025-08-19

**Author** Fabrício Almeida-Silva [aut, cre] (ORCID:

<<https://orcid.org/0000-0002-5314-2964>>),

Thiago Venancio [aut] (ORCID: <<https://orcid.org/0000-0002-2215-8082>>)

**Maintainer** Fabrício Almeida-Silva <[fabricao\\_almeidasilva@hotmail.com](mailto:fabricao_almeidasilva@hotmail.com)>

## Contents

|                                 |    |
|---------------------------------|----|
| chr_length . . . . .            | 2  |
| gcn . . . . .                   | 3  |
| gene_ranges . . . . .           | 3  |
| guides . . . . .                | 4  |
| hubs . . . . .                  | 4  |
| mine2 . . . . .                 | 5  |
| mined_candidates . . . . .      | 5  |
| mine_candidates . . . . .       | 6  |
| mine_step1 . . . . .            | 7  |
| mine_step2 . . . . .            | 8  |
| mine_step3 . . . . .            | 9  |
| pepper_se . . . . .             | 10 |
| plot_snp_circos . . . . .       | 10 |
| plot_snp_distribution . . . . . | 11 |
| score_genes . . . . .           | 11 |
| simulate_windows . . . . .      | 12 |
| snp_pos . . . . .               | 13 |
| tfs . . . . .                   | 14 |

|              |           |
|--------------|-----------|
| <b>Index</b> | <b>15</b> |
|--------------|-----------|

---

|            |                                  |
|------------|----------------------------------|
| chr_length | <i>Pepper chromosome lengths</i> |
|------------|----------------------------------|

---

## Description

Lengths of pepper chromosomes 1-12 in a GRanges object. The genome for which lengths were calculated (v1.55) was downloaded from <http://peppergenome.snu.ac.kr/download.php>

## Usage

```
data(chr_length)
```

## Format

A GRanges object

## Examples

```
data(chr_length)
```

---

|     |                                                                               |
|-----|-------------------------------------------------------------------------------|
| gcn | <i>Simulation of the output list from BioNERO::exp2gcn() with pepper data</i> |
|-----|-------------------------------------------------------------------------------|

---

**Description**

This object is a list as returned by BioNERO::exp2gcn(), but only the element genes\_and\_modules is included. For running time issues, only genes in the cyan module were kept in the element genes\_and\_modules. All other list elements have been assigned NULL. The network was inferred using the code from the vignette.

**Usage**

```
data(gcn)
```

**Format**

A list with the elements returned by BioNERO::exp2gcn().

**Examples**

```
data(gcn)
```

---

|             |                                            |
|-------------|--------------------------------------------|
| gene_ranges | <i>Genomic coordinates of pepper genes</i> |
|-------------|--------------------------------------------|

---

**Description**

GRanges object with genomic coordinates of pepper genes downloaded from <http://peppergenome.snu.ac.kr/download.php>

**Usage**

```
data(gene_ranges)
```

**Format**

A GRanges object

**Examples**

```
data(gene_ranges)
```

---

guides

*Guide genes associated with defense and resistance to oomycetes*

---

### Description

The GO annotation was retrieved from PLAZA 4.0 Dicots.

### Usage

```
data(guides)
```

### Format

A data frame with genes in the first column and GO description in the second column.

### References

Van Bel, M., Diels, T., Vancaester, E., Kreft, L., Botzki, A., Van de Peer, Y., ... & Vandepoele, K. (2018). PLAZA 4.0: an integrative resource for functional, evolutionary and comparative plant genomics. *Nucleic acids research*, 46(D1), D1190-D1196.

### Examples

```
data(guides)
```

---

hubs

*Example hub genes for the network stored in the gcn object*

---

### Description

The data frame was created using the code from the vignette.

### Usage

```
data(hubs)
```

### Format

Data frame with gene IDs, module and intramodular degree.

### Examples

```
data(hubs)
```

---

`mine2`*Example output from mine\_step2()*

---

**Description**

The list was created using the example code from mine\_step().

**Usage**

```
data(mine2)
```

**Format**

List with elements 'candidates' (character vector) and 'enrichment' (data frame).

**Examples**

```
data(mine2)
```

---

`mined_candidates`*Example output from mined\_candidates()*

---

**Description**

The data frame was created using the code from the vignette.

**Usage**

```
data(mined_candidates)
```

**Format**

Data frame with an example of the output from mined\_candidates

**Examples**

```
data(mined_candidates)
```

---

|                 |                                                              |
|-----------------|--------------------------------------------------------------|
| mine_candidates | <i>Mine high-confidence candidate genes in a single step</i> |
|-----------------|--------------------------------------------------------------|

---

## Description

Mine high-confidence candidate genes in a single step

## Usage

```
mine_candidates(
  gene_ranges = NULL,
  marker_ranges = NULL,
  window = 2,
  expand_intervals = TRUE,
  gene_col = "ID",
  exp = NULL,
  gcn = NULL,
  guides = NULL,
  metadata,
  metadata_cols = 1,
  sample_group,
  min_cor = 0.2,
  alpha = 0.05,
  ...
)
```

## Arguments

|                  |                                                                                                                                                                                                                          |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| gene_ranges      | A GRanges object with genomic coordinates of all genes in the genome.                                                                                                                                                    |
| marker_ranges    | Genomic positions of SNPs. For a single trait, a GRanges object. For multiple traits, a GRangesList or CompressedGRangesList object, with each element of the list representing SNP positions for a particular trait.    |
| window           | Sliding window (in Mb) upstream and downstream relative to each SNP. Default: 2.                                                                                                                                         |
| expand_intervals | Logical indicating whether or not to expand markers that are represented by intervals. This is particularly useful if users want to use a custom interval defined by linkage disequilibrium, for example. Default: TRUE. |
| gene_col         | Column of the GRanges object containing gene ID. Default: "ID", the default for gff/gff3 files imported with rtracklayer::import.                                                                                        |
| exp              | Expression data frame with genes in row names and samples in column names or a SummarizedExperiment object.                                                                                                              |
| gcn              | Gene coexpression network returned by BioNERO::exp2gcn().                                                                                                                                                                |
| guides           | Guide genes as a character vector or as a data frame with genes in the first column and gene annotation class in the second column.                                                                                      |
| metadata         | Sample metadata with samples in row names and sample information in the first column. Ignored if exp is a SummarizedExperiment object, as the colData will be extracted from the object.                                 |

|               |                                                                                                                                                                                                                                                                         |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| metadata_cols | A vector (either numeric or character) indicating which columns should be extracted from column metadata if <b>exp</b> is a SummarizedExperiment object. The vector can contain column indices (numeric) or column names (character). By default, all columns are used. |
| sample_group  | Level of sample metadata to be used for filtering in gene-trait correlation.                                                                                                                                                                                            |
| min_cor       | Minimum correlation value for BioNERO::gene_significance(). Default: 0.2                                                                                                                                                                                                |
| alpha         | Numeric indicating significance level. Default: 0.05                                                                                                                                                                                                                    |
| ...           | Additional arguments to BioNERO::gene_significance.                                                                                                                                                                                                                     |

### Value

A data frame with mined candidate genes and their correlation to the condition of interest.

### Examples

```
data(pepper_se)
data(snp_pos)
data(gene_ranges)
data(guides)
data(gcn)
set.seed(1)
candidates <- mine_candidates(gene_ranges, snp_pos, exp = pepper_se,
                             gcn = gcn, guides = guides$Gene,
                             sample_group = "PRR_stress")
```

---

|            |                                                                            |
|------------|----------------------------------------------------------------------------|
| mine_step1 | <i>Step 1: Get all putative candidate genes for a given sliding window</i> |
|------------|----------------------------------------------------------------------------|

---

### Description

For a user-defined sliding window relative to each SNP, this function will subset all genes whose genomic positions overlap with the sliding window.

### Usage

```
mine_step1(gene_ranges, marker_ranges, window = 2, expand_intervals = TRUE)
```

### Arguments

|                  |                                                                                                                                                                                                                          |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| gene_ranges      | A GRanges object with genomic coordinates of all genes in the genome.                                                                                                                                                    |
| marker_ranges    | Genomic positions of SNPs. For a single trait, a GRanges object. For multiple traits, a GRangesList or CompressedGRangesList object, with each element of the list representing SNP positions for a particular trait.    |
| window           | Sliding window (in Mb) upstream and downstream relative to each SNP. Default: 2.                                                                                                                                         |
| expand_intervals | Logical indicating whether or not to expand markers that are represented by intervals. This is particularly useful if users want to use a custom interval defined by linkage disequilibrium, for example. Default: TRUE. |

**Value**

A GRanges or GRangesList object with the genomic positions of all putative candidate genes.

**See Also**

[findOverlaps-methods](#)

**Examples**

```
data(snp_pos)
data(gene_ranges)
genes <- mine_step1(gene_ranges, snp_pos, window = 2)
```

---

|            |                                                                  |
|------------|------------------------------------------------------------------|
| mine_step2 | <i>Step 2: Get candidates in modules enriched in guide genes</i> |
|------------|------------------------------------------------------------------|

---

**Description**

Step 2: Get candidates in modules enriched in guide genes

**Usage**

```
mine_step2(exp, gcn, guides, candidates, ...)
```

**Arguments**

|            |                                                                                                                                     |
|------------|-------------------------------------------------------------------------------------------------------------------------------------|
| exp        | Expression data frame with genes in row names and samples in column names or a SummarizedExperiment object.                         |
| gcn        | Gene coexpression network returned by <code>BioNERO::exp2gcn()</code> .                                                             |
| guides     | Guide genes as a character vector or as a data frame with genes in the first column and gene annotation class in the second column. |
| candidates | Character vector of all candidates genes to be inspected.                                                                           |
| ...        | Additional arguments to <code>BioNERO::module_enrichment</code>                                                                     |

**Value**

A list of 2 elements:

**candidates** Character vector of candidates after step 2

**enrichment** Data frame of results for enrichment analysis

**Examples**

```
data(pepper_se)
data(guides)
data(gcn)
set.seed(1)
mine2 <- mine_step2(
  exp = pepper_se,
  gcn = gcn,
  guides = guides$Gene,
  candidates = rownames(pepper_se)
)
```



mine\_step3

*Step 3: Select candidates based on gene significance***Description**

Step 3: Select candidates based on gene significance

**Usage**

```
mine_step3(
  exp,
  metadata,
  metadata_cols = 1,
  candidates,
  sample_group,
  min_cor = 0.2,
  alpha = 0.05,
  ...
)
```

**Arguments**

|               |                                                                                                                                                                                                                                                                         |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| exp           | Expression data frame with genes in row names and samples in column names or a SummarizedExperiment object.                                                                                                                                                             |
| metadata      | Sample metadata with samples in row names and sample information in the first column. Ignored if exp is a SummarizedExperiment object, as the colData will be extracted from the object.                                                                                |
| metadata_cols | A vector (either numeric or character) indicating which columns should be extracted from column metadata if <b>exp</b> is a SummarizedExperiment object. The vector can contain column indices (numeric) or column names (character). By default, all columns are used. |
| candidates    | Character vector of candidate genes to be inspected.                                                                                                                                                                                                                    |
| sample_group  | Level of sample metadata to be used for filtering in gene-trait correlation.                                                                                                                                                                                            |
| min_cor       | Minimum correlation value for <code>BioNERO::gene_significance()</code> . Default: 0.2                                                                                                                                                                                  |
| alpha         | Numeric indicating significance level. Default: 0.05                                                                                                                                                                                                                    |
| ...           | Additional arguments to <code>BioNERO::gene_significance</code> .                                                                                                                                                                                                       |

**Value**

A data frame with mined candidate genes and their correlation to the condition of interest.

**Examples**

```
data(pepper_se)
data(snp_pos)
data(gene_ranges)
data(guides)
data(gcn)
data(mine2)
set.seed(1)
```

```
mine3 <- mine_step3(
  exp = pepper_se,
  candidates = mine2$candidates,
  sample_group = "PRR_stress"
)
```

---

pepper\_se

*Gene expression data from Kim et al., 2018.*

---

### Description

The data were filtered to keep only the top 4000 genes with highest RPKM values in PRR stress-related samples.

### Usage

```
data(pepper_se)
```

### Format

A SummarizedExperiment object.

### References

Kim, MS., Kim, S., Jeon, J. et al. Global gene expression profiling for fruit organs and pathogen infections in the pepper, *Capsicum annuum* L.. Sci Data 5, 180103 (2018). <https://doi.org/10.1038/sdata.2018.103>

### Examples

```
data(pepper_se)
```

---

plot\_snp\_circos

*Circos plot of SNP distribution across chromosomes*

---

### Description

Circos plot of SNP distribution across chromosomes

### Usage

```
plot_snp_circos(genome_ranges, gene_ranges, marker_ranges)
```

### Arguments

|               |                                                                                                                                                                                                                       |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| genome_ranges | A GRanges object with chromosome lengths.                                                                                                                                                                             |
| gene_ranges   | A GRanges object with genomic coordinates of all genes in the genome.                                                                                                                                                 |
| marker_ranges | Genomic positions of SNPs. For a single trait, a GRanges object. For multiple traits, a GRangesList or CompressedGRangesList object, with each element of the list representing SNP positions for a particular trait. |

**Value**

A ggplot object with a circos plot of molecular marker distribution across chromosomes.

**Examples**

```
data(snp_pos)
data(gene_ranges)
data(chr_length)
p <- plot_snp_circos(chr_length, gene_ranges, snp_pos)
```

---

plot\_snp\_distribution *Plot a barplot of SNP distribution across chromosomes*

---

**Description**

Plot a barplot of SNP distribution across chromosomes

**Usage**

```
plot_snp_distribution(marker_ranges)
```

**Arguments**

**marker\_ranges** Genomic positions of SNPs. For a single trait, a GRanges object. For multiple traits, a GRangesList or CompressedGRangesList object, with each element of the list representing SNP positions for a particular trait. List elements must have names for proper labelling.

**Value**

A ggplot object.

**Examples**

```
data(snp_pos)
p <- plot_snp_distribution(snp_pos)
```

---

score\_genes *Score candidate genes and select the top n genes*

---

**Description**

Score candidate genes and select the top n genes

**Usage**

```
score_genes(
  mined_candidates,
  hubs = NULL,
  tfbs = NULL,
  pick_top = 10,
  weight_tf = 2,
  weight_hub = 2,
  weight_both = 3
)
```

**Arguments**

|                  |                                                                                                                                     |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| mined_candidates | Data frame resulting from mine_candidates() or mine_step().                                                                         |
| hubs             | Character vector of hub genes.                                                                                                      |
| tfbs             | Character vector of transcription factors.                                                                                          |
| pick_top         | Number of top genes to select. Default: 10.                                                                                         |
| weight_tf        | Numeric scalar with the weight to which correlation coefficients will be multiplied if the gene is a TF. Default: 2.                |
| weight_hub       | Numeric scalar with the weight to which correlation coefficients will be multiplied if the gene is a hub. Default: 2.               |
| weight_both      | Numeric scalar with the weight to which correlation coefficients will be multiplied if the gene is both a TF and a hub. Default: 3. |

**Value**

Data frame with top n candidates and their scores.

**Examples**

```
data(tfbs)
data(hubs)
data(mined_candidates)
set.seed(1)
scored <- score_genes(mined_candidates, hubs$Gene, tfbs$Gene_ID)
```

---

|                  |                                                         |
|------------------|---------------------------------------------------------|
| simulate_windows | <i>Simulate number of genes for each sliding window</i> |
|------------------|---------------------------------------------------------|

---

**Description**

This function counts genes that are contained in sliding windows related to each SNP.

**Usage**

```
simulate_windows(
  gene_ranges,
  marker_ranges,
  windows = seq(0.1, 2, by = 0.1),
  expand_intervals = TRUE
)
```

**Arguments**

|                  |                                                                                                                                                                                                                          |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| gene_ranges      | A GRanges object with genomic coordinates of all genes in the genome.                                                                                                                                                    |
| marker_ranges    | Genomic positions of SNPs. For a single trait, a GRanges object. For multiple traits, a GRangesList or CompressedGRangesList object, with each element of the list representing SNP positions for a particular trait.    |
| windows          | Sliding windows (in Mb) upstream and downstream relative to each SNP. Default: seq(0.1, 2, by = 0.1).                                                                                                                    |
| expand_intervals | Logical indicating whether or not to expand markers that are represented by intervals. This is particularly useful if users want to use a custom interval defined by linkage disequilibrium, for example. Default: TRUE. |

**Details**

By default, the function creates 20 sliding windows by expanding upstream and downstream boundaries for each SNP from 0.1 Mb (100 kb) to 2 Mb.

**Value**

A ggplot object summarizing the results of the simulations.

**See Also**

[findOverlaps-methods](#)

**Examples**

```
data(snp_pos)
data(gene_ranges)
simulate_windows(gene_ranges, snp_pos)
```

---

|         |                                                                                         |
|---------|-----------------------------------------------------------------------------------------|
| snp_pos | <i>Capsicum annuum</i> SNPs associated with resistance to <i>Phytophthora</i> root rot. |
|---------|-----------------------------------------------------------------------------------------|

---

**Description**

The SNPs in this data set were retrieved from Siddique et al., 2019, and they are associated to resistance to *Phytophthora* root rot.

**Usage**

```
data(snp_pos)
```

**Format**

A GRanges object.

## References

Siddique, M.I., Lee, H.Y., Ro, N.Y. et al. Identifying candidate genes for *Phytophthora capsici* resistance in pepper (*Capsicum annuum*) via genotyping-by-sequencing-based QTL mapping and genome-wide association study. *Sci Rep* 9, 9962 (2019). <https://doi.org/10.1038/s41598-019-46342-1>

## Examples

```
data(snp_pos)
```

---

|     |                                     |
|-----|-------------------------------------|
| tfs | <i>Pepper transcription factors</i> |
|-----|-------------------------------------|

---

## Description

Pepper transcription factors and their families retrieved from PlantTFDB 4.0.

## Usage

```
data(tfs)
```

## Format

A data frame with gene IDs in the first column and TF families in the second column.

## References

Jin, J., Tian, F., Yang, D. C., Meng, Y. Q., Kong, L., Luo, J., & Gao, G. (2016). PlantTFDB 4.0: toward a central hub for transcription factors and regulatory interactions in plants. *Nucleic acids research*, gkw982.

## Examples

```
data(tfs)
```

# Index

## \* datasets

- chr\_length, [2](#)
- gcn, [3](#)
- gene\_ranges, [3](#)
- guides, [4](#)
- hubs, [4](#)
- mine2, [5](#)
- mined\_candidates, [5](#)
- pepper\_se, [10](#)
- snp\_pos, [13](#)
- tfs, [14](#)

chr\_length, [2](#)

gcn, [3](#)  
gene\_ranges, [3](#)  
guides, [4](#)

hubs, [4](#)

mine2, [5](#)  
mine\_candidates, [6](#)  
mine\_step1, [7](#)  
mine\_step2, [8](#)  
mine\_step3, [9](#)  
mined\_candidates, [5](#)

pepper\_se, [10](#)  
plot\_snp\_circos, [10](#)  
plot\_snp\_distribution, [11](#)

score\_genes, [11](#)  
simulate\_windows, [12](#)  
snp\_pos, [13](#)

tfs, [14](#)