# Package ‘BioVizSeq’

May 6, 2026

**Type** Package

**Title** Visualizing the Elements Within Bio-Sequences

**Version** 1.0.5

**Maintainer** Shiqi Zhao <`zhaosq89@163.com`>

**Description** Visualizing the types and distribution of elements within bio-sequences. At the same time, We have developed a geom layer, geom_rrect(), that can generate rounded rectangles. No external references are used in the development of this package.

**Encoding** UTF-8

**License** Artistic-2.0

**Language** en-US

**RoxygenNote** 7.2.3

**Repository** CRAN

**Imports** dplyr, ggplot2, httr, RColorBrewer, seqinr, stringr, tidyr, magrittr, ggtree, treeio, ggh4x, shiny

**NeedsCompilation** no

**Author** Shiqi Zhao [aut, cre, cph] (ORCID: <`https://orcid.org/0000-0002-2360-956X`>)

**Date/Publication** 2026-01-10 12:00:02 UTC

## Contents

---

## `biovizseq` *BioVizSeq shiny app start function.*

---

### Description

BioVizSeq shiny app start function.

### Usage

```
biovizseq()
```

### Value

Shinyapp: BioVizSeq shiny app.

### Author(s)

Shiqi Zhao

### Examples

```
# 1. Library BioVizSeq package
library(BioVizSeq)
```

---

## cdd_plot *cdd_plot*

---

### Description

Visualization of domain in CDD file

### Usage

```
cdd_plot(
  cdd_file,
  fasta_file,
  the_order = NULL,
  domain_select = NULL,
  shape = "RoundRect",
  r = 0.3,
  legend_size = 15,
  domain_color = NULL
)
```

### Arguments

| | |
|---|---|
| `cdd_file` | The path of cdd file. |
| `fasta_file` | The path of fasta file. |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |
| `domain_select` | The domain ID which you want to align with. |
| `shape` | RoundRect or Rect. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |
| `domain_color` | The color set of domain. |

### Value

p

### Author(s)

Shiqi Zhao

### Examples

```
hitdata_path <- system.file("extdata", "hitdata.txt", package = "BioVizSeq")
fa_path <- system.file("extdata", "idpep.fa", package = "BioVizSeq")
cdd_plot(hitdata_path, fa_path)

order_path <- system.file("extdata", "order.csv", package = "BioVizSeq")
cdd_plot(hitdata_path, fa_path, the_order = order_path)
```

---

## `cdd_to_loc` *cdd_to_loc*

---

### Description

Extract the location information of domain from cdd file

### Usage

```
cdd_to_loc(cdd_file)
```

### Arguments

| | |
|---|---|
| `cdd_file` | CDD file. |

### Value

data.frame

### Author(s)

Shiqi Zhao

### Examples

```
hitdata_path <- system.file("extdata", "hitdata.txt", package = "BioVizSeq")
cdd_file <- readLines(hitdata_path)
domain_loc <- cdd_to_loc(cdd_file)
```

---

## `combi_p` *combi_p*

---

### Description

Get ggplot2 files to facilitate free combination in patchwork

### Usage

```
combi_p(
  tree_path,
  gff_path = NULL,
  meme_path = NULL,
  pfam_path = NULL,
  cdd_path = NULL,
  fa_path = NULL,
  smart_path = FALSE,
```

```
  plantcare_path = NULL,
  promoter_length = NULL,
  renamefile = NULL,
  groupfile = NULL,
  shape = "RoundRect",
  r = 0.3,
  legend_size = 6
)
```

## Arguments

| | |
|---|---|
| `tree_path` | The path of tree file (.newick). |
| `gff_path` | The path of .gff/gtf file. |
| `meme_path` | The path of .meme/mast file. |
| `pfam_path` | The path of pfam result file (.tsv). |
| `cdd_path` | The path of cdd result file (.txt). |
| `fa_path` | The path of protein file (.fa/fasta). |
| `smart_path` | Do SMART or not. (TRUE or FALSE) |
| `plantcare_path` | The path of plantcare file (.tab). |
| `promoter_length` | The length of promoter. |
| `renamefile` | Rename file. Two cols: new_name and old_name. |
| `groupfile` | Group information. Two cols: label and Group. |
| `shape` | RoundRect or Rect. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |

## Value

list

## Author(s)

Shiqi Zhao

## Examples

```
tree_path <- system.file("extdata", "idpep.nwk", package = "BioVizSeq")
plot_file <- combi_p(tree_path)
```

---

`combi_p2` *combi_p2*

---

## Description

Get ggplot2 files to facilitate free combination in patchwork

## Usage

```
combi_p2(
  order_path,
  gff_path = NULL,
  meme_path = NULL,
  pfam_path = NULL,
  cdd_path = NULL,
  fa_path = NULL,
  smart_path = FALSE,
  plantcare_path = NULL,
  promoter_length = NULL,
  renamefile = NULL,
  shape = "RoundRect",
  r = 0.3,
  legend_size = 6
)
```

## Arguments

| | |
|---|---|
| `order_path` | The path of order file (.txt/.csv). |
| `gff_path` | The path of .gff/gtf file. |
| `meme_path` | The path of .meme/mast file. |
| `pfam_path` | The path of pfam result file (.tsv). |
| `cdd_path` | The path of cdd result file (.txt). |
| `fa_path` | The path of protein file (.fa/fasta). |
| `smart_path` | Do SMART or not. (TRUE or FALSE) |
| `plantcare_path` | The path of plantcare file (.tab). |
| `promoter_length` | The length of promoter. |
| `renamefile` | Rename file. Two cols: new_name and old_name. |
| `shape` | RoundRect or Rect. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |

## Value

list

## Author(s)

Shiqi Zhao

## Examples

```
order_path <- system.file("extdata", "order.csv", package = "BioVizSeq")
gff_path <- system.file("extdata", "idpro.gff3", package = "BioVizSeq")
plot_file <- combi_p2(order_path, gff_path = gff_path)
```

---

# `fastaleng` *fastaleng*

## Description

Statistical sequence length

## Usage

```
fastaleng(fasta_file)
```

## Arguments

| | |
|---|---|
| `fasta_file` | The path of protein fasta file. |

## Value

data.frame

## Author(s)

Shiqi Zhao

## Examples

```
fasta_path <- system.file("extdata", "idpep.fa", package = "BioVizSeq")
fastaleng(fasta_path)
```

---

## `geom_rrect` *geom_rrect*

---

### Description

Rounded rectangle

### Usage

```
geom_rrect(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  r = 0.2,
  ...,
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

### Arguments

| | |
|---|---|
| `mapping` | Set of aesthetic mappings created by `aes`. If specified and `inherit.aes = TRUE` (the default), it is combined with the default mapping at the top level of the plot. You must supply `mapping` if there is no plot mapping. |
| `data` | A data.frame, or other object, will override the plot data. All objects will be fortified to produce a data frame. |
| `stat` | Name of stat to modify data. |
| `position` | The position adjustment to use for overlapping points on this layer. |
| `r` | The radius of rounded corners. |
| `...` | additional parameter, e.g. color, linewidth, alpha. |
| `na.rm` | If "FALSE" (default), missing values are removed with a warning. If "TRUE", missing values are silently removed, logical. |
| `show.legend` | Whether to show legend, logical. |
| `inherit.aes` | Whether to inherit aesthetic mappings, logical, defaults to "TRUE". |

### Details

draws rounded rectangle by using the locations of the four corners (xmin, xmax, ymin and ymax) like geom_rect().

### Value

ggplot object

### Aesthetics

`geom_rrect()` understands the following aesthetics (required aesthetics are in bold):

- `xmin`
- `xmax`
- `ymin`
- `ymax`
- `alpha`
- `colour`
- `fill`
- `group`
- `linetype`

Learn more about setting these aesthetics in `vignette("ggplot2-specs")`.

### Author(s)

Shiqi Zhao

### Examples

```
library(ggplot2)
df <- data.frame(
 xmin = c(1, 2, 3),
 xmax = c(2, 3, 4),
 ymin = c(1, 2, 3),
 ymax = c(2, 3, 4),
 category = c("A", "B", "C")
 )

 p <- ggplot(df) +
   geom_rrect(aes(xmin = xmin, xmax = xmax,
              ymin = ymin, ymax = ymax, fill = category),
             r = 0.4, linewidth = 1, colour = "black")

 print(p)
```

---

## `gff_plot` *gff_plot*

### Description

Visualization of element in gff or gtf file

### Usage

```
gff_plot(
  gff_file,
  the_order = NULL,
  shape = "Rect",
  r = 0.3,
  legend_size = 15,
  element_color = NULL
)
```

### Arguments

| | |
|---|---|
| `gff_file` | The path of gff file. |
| `the_order` | The path of order of mRNA. It is also the mRNA you want to showcase. A List of Gene ID , One ID Per Line. |
| `shape` | RoundRect or Rect. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |
| `element_color` | The color set of element. |

### Value

p

### Author(s)

Shiqi Zhao

### Examples

```
gff_path <- system.file("extdata", "test.gff", package = "BioVizSeq")
gff_plot(gff_path)
```

---

## `gff_statistics` *gff_statistics*

### Description

Extract the information of element from gff or gtf file

### Usage

```
gff_statistics(gff_data, mRNA_ID = NULL)
```

### Arguments

| | |
|---|---|
| `gff_data` | gff file. |
| `mRNA_ID` | The mRNA you selected. If NULL, it means selecting all mRNAs. |

### Value

data.frame

### Author(s)

Shiqi Zhao

### Examples

```
gff_path <- system.file("extdata", "idpro.gff3", package = "BioVizSeq")
gff_data <- read.table(gff_path, header = FALSE, sep = '\t')
gff_statistics_data <- gff_statistics(gff_data)
```

---

## `gff_to_loc` *gff_to_loc*

### Description

Extract the location information of element from gff or gtf file

### Usage

```
gff_to_loc(gff_data, mRNA_ID = NULL)
```

### Arguments

| | |
|---|---|
| `gff_data` | gff file. |
| `mRNA_ID` | The mRNA you selected. If NULL, it means selecting all mRNAs. |

### Value

list

### Author(s)

Shiqi Zhao

## Examples

```
gff_path <- system.file("extdata", "test.gff", package = "BioVizSeq")
gff_data <- read.table(gff_path, header = FALSE, sep = '\t')
gff_loc <- gff_to_loc(gff_data)

ID_path <- system.file("extdata", "ID_select.csv", package = "BioVizSeq")
mRNA_ID <- readLines(ID_path)
gff_loc <- gff_to_loc(gff_data, mRNA_ID=mRNA_ID)
```

---

# `meme_plot` *meme_plot*

## Description

Visualization of motif in meme file or mast file

## Usage

```
meme_plot(
  meme_file,
  the_order = NULL,
  motif_select = NULL,
  shape = "RoundRect",
  show_motif_id = FALSE,
  r = 0.3,
  legend_size = 15,
  motif_color = NULL
)
```

## Arguments

| | |
|---|---|
| `meme_file` | The path of meme file or mast file. |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |
| `motif_select` | The motif ID which you want to align with. |
| `shape` | RoundRect or Rect. |
| `show_motif_id` | Display the name of the motif. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |
| `motif_color` | The color set of motif. |

## Value

p

### Author(s)

Shiqi Zhao

### Examples

```
mast_path <- system.file("extdata", "mast.xml", package = "BioVizSeq")
meme_plot(mast_path)

meme_plot(mast_path, motif_select="1", show_motif_id = TRUE)

order_path <- system.file("extdata", "order.csv", package = "BioVizSeq")
meme_plot(mast_path, the_order=order_path, motif_select="1")
```

---

## `meme_seq` *meme_seq*

### Description

Get motif sequence from meme file or mast file

### Usage

```
meme_seq(meme_file)
```

### Arguments

| | |
|---|---|
| `meme_file` | The path of meme file or mast file. |

### Value

data.frame

### Author(s)

Shiqi Zhao

### Examples

```
mast_path <- system.file("extdata", "mast.xml", package = "BioVizSeq")
mast_file <- readLines(mast_path)
motifseq<- meme_seq(mast_file)
```

---

`meme_to_loc` *meme_to_loc*

---

### Description

Extract the location information of motif from mast or meme file

### Usage

```
meme_to_loc(motif_file)
```

### Arguments

| | |
|---|---|
| `motif_file` | The motif data of mast or meme file. |

### Value

list

### Author(s)

Shiqi Zhao

### Examples

```
mast_path <- system.file("extdata", "mast.xml", package = "BioVizSeq")
mast_file <- readLines(mast_path)
motif_loc <- meme_to_loc(mast_file)
```

---

`motif_plot` *motif_plot*

---

### Description

Draws multiple rounded rectangle.

### Usage

```
motif_plot(
  motif_loc,
  gene_length,
  the_order = NULL,
  motif_select = NULL,
  shape = "RoundRect",
  show_motif_id = FALSE,
```

```
  r = 0.3,
  legend_size = 15,
  motif_color = NULL
)
```

## Arguments

| | |
|---|---|
| `motif_loc` | A data.frame contains for columuns: ID, motif, start, end. |
| `gene_length` | A data.fram of the length of biosequences. Two columns: ID, length. |
| `the_order` | A List of Gene ID , One ID Per Line. |
| `motif_select` | The motif ID which you want to align with. |
| `shape` | RoundRect or Rect. |
| `show_motif_id` | Display the name of the motif. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |
| `motif_color` | The color set of motif. |

## Details

motif_plot() draws multiple rounded rectangle to represent the above elements of biosequences, but not limited to biosequences

## Value

p

## Author(s)

Shiqi Zhao

## Examples

```
df <- data.frame(
 ID = rep(c("geneA", "geneB", "geneC"), each = 3),
 motif = rep(c("1", "2", "3"), times = 3),
 start = c(1, 3, 6, 1, 6, 10, 10, 7, 17),
 end = c(3, 5, 11, 3, 8, 15, 12, 9, 22)
 )

 length_data <- data.frame(
 ID = c("geneA", "geneB", "geneC"),
 length = c(15, 27, 30)
 )

 order_data <- c("geneB", "geneA", "geneC")

 motif_plot(df, length_data)
 motif_plot(df, length_data, the_order = order_data)
```

---

`pfam_plot` *pfam_plot*

---

## Description

Visualization of domain in pfam result file

## Usage

```
pfam_plot(
  pfam_file,
  the_order = NULL,
  domain_select = NULL,
  shape = "RoundRect",
  r = 0.3,
  legend_size = 15,
  domain_color = NULL
)
```

## Arguments

| | |
|---|---|
| `pfam_file` | The path of meme file or mast file. |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |
| `domain_select` | The domain ID which you want to align with. |
| `shape` | RoundRect or Rect. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |
| `domain_color` | The color set of domain. |

## Value

p

## Author(s)

Shiqi Zhao

## Examples

```
pfam_path <- system.file("extdata", "iprscan.tsv", package = "BioVizSeq")
order_path <- system.file("extdata", "order.csv", package = "BioVizSeq")
pfam_plot(pfam_path)
pfam_plot(pfam_path, the_order=order_path)
```

---

## `pfam_to_loc` *pfam_to_loc*

---

### Description

Extract the location information of domain from pfam result

### Usage

```
pfam_to_loc(pfam_data)
```

### Arguments

| | |
|---|---|
| `pfam_data` | The result file (.tsv) of pfam (via InterPro). |

### Value

list

### Author(s)

Shiqi Zhao

### Examples

```
pfam_path <- system.file("extdata", "iprscan.tsv", package = "BioVizSeq")
pfam_file <- read.table(pfam_path, sep='\t', header = FALSE)
motif_loc <- pfam_to_loc(pfam_file)
```

---

## `plantcare_classify` *plantcare_classify*

---

### Description

Classify the functions of cis element from Plantcare

### Usage

```
plantcare_classify(plantcare_file)
```

### Arguments

| | |
|---|---|
| `plantcare_file` | The result file (.tab) of Plantcare. |

### Value

data.frame

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_file <- read.table(plantcare_path, header = FALSE, sep = '\t', quote="")
plantcare_data <- plantcare_classify(plantcare_file)
```

---

## `plantcare_plot` *plantcare_plot*

### Description

Visualization of cis-element in plantcare result file

### Usage

```
plantcare_plot(
  plantcare_file,
  promoter_length = 2000,
  the_order = NULL,
  shape = "Rect",
  r = 6,
  legend_size = 15,
  element_color = NULL
)
```

### Arguments

| | |
|---|---|
| `plantcare_file` | The path of plantcare result file (.tab). |
| `promoter_length` | The promoter length. |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |
| `shape` | RoundRect or Rect. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |
| `element_color` | The color set of cis-element. |

### Value

p

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_plot(plantcare_path, promoter_length = 2000)
```

---

## `plantcare_plot1` *plantcare_plot1*

---

### Description

Heatmap of the number of cis-element in plantcare result file

### Usage

```
plantcare_plot1(plantcare_file, the_order = NULL)
```

### Arguments

| | |
|---|---|
| `plantcare_file` | The path of plantcare result file (.tab). |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |

### Value

p

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_plot1(plantcare_path)
```

---

## `plantcare_plot2` *plantcare_plot2*

---

### Description

Bar plot1 of the number of cis-element in plantcare result file

### Usage

```
plantcare_plot2(plantcare_file, the_order = NULL)
```

### Arguments

| | |
|---|---|
| `plantcare_file` | The path of plantcare result file (.tab). |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |

### Value

p

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_plot2(plantcare_path)
```

---

## `plantcare_plot3` *plantcare_plot3*

### Description

Bar plot2 of the number of cis-element in plantcare result file

### Usage

```
plantcare_plot3(plantcare_file, the_order = NULL)
```

### Arguments

| | |
|---|---|
| `plantcare_file` | The path of plantcare result file (.tab). |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |

### Value

p

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_plot3(plantcare_path)
```

---

## `plantcare_statistic1` *plantcare_statistic1*

---

### Description

Count the number of cis element from Plantcare for heatmap

### Usage

```
plantcare_statistic1(plantcare_data)
```

### Arguments

| | |
|---|---|
| `plantcare_data` | The result of plantcare_classify(). |

### Value

data.frame

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_file <- read.table(plantcare_path, header = FALSE, sep = '\t', quote="")
plantcare_data <- plantcare_classify(plantcare_file)
statistic_data1 <- plantcare_statistic1(plantcare_data)
```

---

## `plantcare_statistic2` *plantcare_statistic2*

---

### Description

Count the number of cis element from Plantcare for Bar chart

### Usage

```
plantcare_statistic2(plantcare_data)
```

### Arguments

| | |
|---|---|
| `plantcare_data` | The result of plantcare_classify(). |

### Value

data.frame

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_file <- read.table(plantcare_path, header = FALSE, sep = '\t', quote="")
plantcare_data <- plantcare_classify(plantcare_file)
statistic_data2 <- plantcare_statistic2(plantcare_data)
```

---

## `plantcare_to_loc` *plantcare_to_loc*

---

### Description

Extract the location information of cis-element from Plantcare

### Usage

```
plantcare_to_loc(plantcare_data)
```

### Arguments

| | |
|---|---|
| `plantcare_data` | The result of plantcare_classify(). |

### Value

data.frame

### Author(s)

Shiqi Zhao

### Examples

```
plantcare_path <- system.file("extdata", "plantCARE_output.tab", package = "BioVizSeq")
plantcare_file <- read.table(plantcare_path, header = FALSE, sep = '\t', quote="")
plantcare_data <- plantcare_classify(plantcare_file)
plantcare_loc <- plantcare_to_loc(plantcare_data)
```

---

`ProtParam_calc` *ProtParam_calc*

---

## Description

Extract the information of protein sequence

## Usage

```
ProtParam_calc(input_file)
```

## Arguments

| | |
|---|---|
| `input_file` | The path of potein fasta file. |

## Value

data.frame

## Author(s)

Shiqi Zhao

---

`smart_plot` *smart_plot*

---

## Description

Visualization of domain in SMART result file

## Usage

```
smart_plot(
  fasta_file,
  the_order = NULL,
  domain_select = NULL,
  shape = "RoundRect",
  r = 0.3,
  legend_size = 15,
  domain_color = NULL
)
```

## Arguments

| | |
|---|---|
| `fasta_file` | The path of protein fasta file. |
| `the_order` | The path of order file. A List of Gene ID , One ID Per Line. |
| `domain_select` | The domain ID which you want to align with. |
| `shape` | RoundRect or Rect. |
| `r` | The radius of rounded corners. |
| `legend_size` | The size of legend. |
| `domain_color` | The color set of domain. |

## Value

p

## Author(s)

Shiqi Zhao

---

# `smart_to_loc` *smart_to_loc*

## Description

Extract the location information of domain from SMART result

## Usage

```
smart_to_loc(input_file, do_pfam = TRUE)
```

## Arguments

| | |
|---|---|
| `input_file` | The path of potein fasta file. |
| `do_pfam` | Include the pfam domain or not. |

## Value

list

## Author(s)

Shiqi Zhao

---

`upload_fa_to_plantcare`

## *upload_fa_to_plantcare*

---

### Description

Upload the promoter file to Plantcare database

### Usage

```
upload_fa_to_plantcare(fasta_file, email)
```

### Arguments

| | |
|---|---|
| `fasta_file` | The path of promoter file. |
| `email` | e-mail address. |

### Details

Due to the file size limitation of plantcare on fasta, upload_fa_to_plantcare() first splits fasta file. Then uploads the splited fasta files to the plantcare database, and automatically returns the results to the email provided by the user.

upload_fa_to_plantcare("the path/test.fasta", "your e-mail address")

### Value

plantcare_result

### Author(s)

Shiqi Zhao

# Index