Package ‘hicVennDiagram’

July 10, 2025

Title Venn Diagram for genomic interaction data

Version 1.6.0

Description A package to generate high-resolution Venn and Upset plots for genomic interaction data from HiC, ChIA-PET, HiChIP, PLAC-Seq, Hi-TrAC, HiCAR and etc. The package generates plots specifically crafted to eliminate the deceptive visual representation caused by the counts method.

License GPL-3

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

VignetteBuilder knitr

biocViews DNA3DStructure, HiC, Visualization

Depends R (>= 4.3.0)

Imports GenomeInfoDb, GenomicRanges, IRanges, InteractionSet, rtracklayer, ggplot2, ComplexUpset, reshape2, eulerr, S4Vectors, methods, utils, htmlwidgets, svglite

Suggests BiocStyle, knitr, rmarkdown, testthat, ChIPpeakAnno, grid, TxDb.Hsapiens.UCSC.hg38.knownGene

URL `https://github.com/jianhong/hicVennDiagram`

BugReports `https://github.com/jianhong/hicVennDiagram/issues`

git_url https://git.bioconductor.org/packages/hicVennDiagram

git_branch RELEASE_3_21

git_last_commit 30f55ec

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-07-09

Author Jianhong Ou [aut, cre] (ORCID: <`https://orcid.org/0000-0002-8652-2488`>)

Maintainer Jianhong Ou <`jou@morgridge.org`>

Contents

`browseVenn` *Browse the venn plot*

Description

Brow the venn plot in a web browser to adjust the plot and export the result.

Usage

```
browseVenn(plot, width = NULL, height = NULL)
```

Arguments

`plot`	plots of vennPlot or upsetPlot
`width`	width of the figure
`height`	height of the figure

Value

An object of class htmlwidget that will intelligently print itself into HTML in a variety of contexts including the R console, within R Markdown documents, and within Shiny output bindings.

Examples

```
pd <- system.file("extdata", package = "hicVennDiagram", mustWork = TRUE)
fs <- dir(pd, pattern = ".bedpe", full.names = TRUE)
vc <- vennCount(fs)
p <- vennPlot(vc)
browseVenn(p)
```

`browseVenn-shiny` *Shiny bindings for browseVenn*

Description

Output and render functions for using browseVenn within Shiny applications and interactive Rmd documents.

Usage

```
browseVennOutput(outputId, width = "100%", height = "400px")

renderbrowseVenn(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

`outputId`	output variable to read from
`width, height`	Must be a valid CSS unit (like `'100%'`, `'400px'`, `'auto'`) or a number, which will be coerced to a string and have `'px'` appended.
`expr`	An expression that generates a browseVenn
`env`	The environment in which to evaluate `expr`.
`quoted`	Is `expr` a quoted expression (with `quote()`)? This is useful if you want to save an expression in a variable.

Value

An output or render function that enables the use of the widget within Shiny applications.

`createGIbackground` *Create background by input GInteractions*

Description

Create background based on the distance distribution of input GInteractions.

Usage

```
createGIbackground(gi, size = 2 * lengths(gi))
```

Arguments

`gi`	A vector of bedpe files or a list of genomic interaction data (Pairs or GInteractions).
`size`	The maximal size of the background

Value

A GInteractions object.

Examples

```
pd <- system.file("extdata", package = "hicVennDiagram", mustWork = TRUE)
fs <- dir(pd, pattern = ".bedpe", full.names = TRUE)[1]
set.seed(123)
# createGIbackground(fs)
```

`gleamTest` *Perform GLEAM test*

Description

Run Genomic Loops Enrichment Analysis Method (GLEAM) test. GLEAM identifies the query is significantly over-represented within the subject by given background if subject is available. Otherwise, the GLEAM will be tested among the queries.

Usage

```
gleamTest(query, subject, background, method = c("binom", "hyper"), ...)
```

Arguments

`query, subject`	A vector of bedpe files or a list of genomic interaction data (Pairs or GInteractions) or a list of GRanges object. 'subject' is optional if length of query > 1.
`background`	The test will restricted within the region. The background is the background of subject if subject is available. Otherwise, the background is the the background of second element of comparison group.
`method`	Distribution type for p-value.
`...`	parameters used by findOverlaps.

Value

A data.frame of test results.

Examples

```
# example code
pd <- system.file("extdata", package = "hicVennDiagram", mustWork = TRUE)
fs <- dir(pd, pattern = ".bedpe", full.names = TRUE)
library(TxDb.Hsapiens.UCSC.hg38.knownGene)
## set.seed(123)
## background <- createGIbackground(fs)
## gleamTest(fs, background = background)
```

```
## gleamTest(fs, background = background, method = 'hyper')
grl <- GRangesList(exons=reduce(exons(TxDb.Hsapiens.UCSC.hg38.knownGene)),
 genes=reduce(genes(TxDb.Hsapiens.UCSC.hg38.knownGene)))
gleamTest(fs[seq_along(grl)], grl, background = grl[['exons']])
gleamTest(grl[c(2, 1)], grl, background = grl[['exons']])
gleamTest(grl, background = grl[['genes']])
```

`upsetPlot` *UpSet plot for the Venn count table*

Description

Plot the overlaps counts by ComplexUpset.

Usage

```
upsetPlot(
  vennTable,
 label_all = list(na.rm = TRUE, color = "gray30", alpha = 0.7, label.padding = unit(0.1,
    "lines")),
  coln_prefix = NULL,
  ...
)
```

Arguments

`vennTable`	An vennTable object, the first element in the output of vennCount.
`label_all`	A list of parameters used by geom_label for text labels of counts for each group. If it set to FALSE or length of the list is zero, the labels will be ignored.
`coln_prefix`	The prefix to be removed for column names of vennTable.
`...`	Parameters could be passed to upset except `data` and `intersect`.

Value

A ggplot object.

Examples

```
pd <- system.file("extdata", package = "hicVennDiagram", mustWork = TRUE)
fs <- dir(pd, pattern = ".bedpe", full.names = TRUE)
vc <- vennCount(fs)
upset_themes_fix <- lapply(ComplexUpset::upset_themes, function(.ele){
.ele[names(.ele) %in% names(formals(ggplot2::theme))]
})
upsetPlot(vc, theme = upset_themes_fix)
## change the font size of lables and numbers
themes <- ComplexUpset::upset_modify_themes(
 ## get help by vignette('Examples_R', package = 'ComplexUpset')
```

```
        list('intersections_matrix'=
            ggplot2::theme(axis.text.y=ggplot2::element_text(size=24)))
)
themes <- lapply(themes, function(.ele){
.ele[names(.ele) %in% names(formals(ggplot2::theme))]
})
upsetPlot(vc, label_all=list(
                        na.rm = TRUE,
                        color = 'gray30',
                        alpha = .7,
                        label.padding = grid::unit(0.1, "lines"),
                        size = 5
), themes = themes)
```

---

vennCount *Construct intersections of sets*

Description

Given a collection of bedpe files or a list of genomic interaction data, `vennCount` will compute all possible combinations of interactions and return an object of class vennTable, storing the combinations as well as the number of elements in each intersection.

Usage

```
vennCount(gi, FUN = min, ...)
```

Arguments

`gi`	A vector of bedpe files or a list of genomic interaction data (Pairs or GInteractions)
`FUN`	Function to summarize the overlapping number.
`...`	parameters used by findOverlaps

Value

An object of vennTable

Examples

```
pd <- system.file("extdata", package = "hicVennDiagram", mustWork = TRUE)
fs <- dir(pd, pattern = ".bedpe", full.names = TRUE)
vc <- vennCount(fs)
```

`vennPlot` *Venn diagram for the Venn count table*

Description

Plot the overlaps counts by eulerr.

Usage

```
vennPlot(vennTable, shape = "circle", ...)
```

Arguments

`vennTable`	An vennTable object, the first element in the output of vennCount.
`shape`	Geometric shape used in the diagram used by euler.
`...`	parameters to update fills and edges with and thereby a shortcut to set these parameters plot.euler.

Value

A grid object.

Examples

```
pd <- system.file("extdata", package = "hicVennDiagram", mustWork = TRUE)
fs <- dir(pd, pattern = ".bedpe", full.names = TRUE)
vc <- vennCount(fs)
vennPlot(vc)
## change the font size of venn plot lables and numbers,
## both cex or fontsize should work
vennPlot(vc, quantities=list(fontsize=24), labels=list(cex=1.5))
```

`vennTable-class` *Class* `"vennTable"`

Description

An object of class `"vennTable"` represents Venn counts.

Usage

```
vennTable(...)

## S4 method for signature 'vennTable'
x$name

## S4 replacement method for signature 'vennTable'
x$name <- value

## S4 method for signature 'vennTable,ANY,ANY'
x[[i]]

## S4 replacement method for signature 'vennTable,ANY,ANY'
x[[i]] <- value

## S4 method for signature 'vennTable'
show(object)
```

Arguments

`...`	Each argument in ... becomes an slot in the new vennTable.
`x`	an object of vennTable
`name`	slot name of vennTable
`value`	values to assign
`i`	slot name of vennTable
`object`	an object of vennTable.

Value

An object of vennTable.

Slots

`combinations` A logical `"matrix"`, specify the combinations.

`counts` A `"numeric"` vector, the overall counts number for each combination.

`vennCounts` A `"matrix"` object, specify the counts number for each sample in the combination.

`overlapList` `"list"`, overlapping list of the genomic interactions.

Examples

```
vt <- vennTable()
```

Index