

# Package ‘VennDetail’

November 21, 2024

**Type** Package

**Title** A package for visualization and extract details

**Version** 1.23.0

**Author** Kai Guo, Brett McGregor

**Maintainer** Kai Guo <guokai8@gmail.com>

**Description** A set of functions to generate high-resolution Venn,Vennpie plot,extract and combine details of these subsets with user datasets in data frame is available.

**License** GPL-2

**Encoding** UTF-8

**LazyData** True

**Imports** utils, grDevices, stats, methods, dplyr, purrr, tibble, magrittr, ggplot2, UpSetR, VennDiagram, grid, futile.logger

**Suggests** knitr, rmarkdown, testthat, markdown

**VignetteBuilder** knitr

**URL** <https://github.com/guokai8/VennDetail>

**RoxygenNote** 6.1.1

**biocViews** DataRepresentation,GraphAndNetwork

**NeedsCompilation** no

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

**git\_branch** devel

**git\_last\_commit** cecf8f1

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.21

**Date/Publication** 2024-11-20

## Contents

.add_colnames	2
.make.table	3
detail	3
dplot	4
getFeature	5
getSet	6
make.subset	7
merge.Venn	7
plot.Venn	8
result	10
rowjoin	11
setcolor	12
show Venn	12
summary.Venn	13
T2DM	14
Venn-class	14
venndetail	15
vennpie	16
<b>Index</b>	<b>18</b>

---

.add_colnames	<i>Give first colname as RowNxyz</i>
---------------	--------------------------------------

---

### Description

Give first colname as RowNxyz

### Usage

```
.add_colnames(x)
```

### Arguments

x                    data frame

### Value

return data frame with the first colnames change to "RowNxyz"

---

.make.table	<i>make table for venndetail modified from make.truth.table (VennDiagram)</i>
-------------	---

---

**Description**

make table for venndetail modified from make.truth.table (VennDiagram)

**Usage**

```
.make.table(x)
```

**Arguments**

x                    A list with input groups

**Value**

A data frame with logical vector columns and  $2^{\text{length}(x)-1}$  rows.

**Author(s)**

Kai Guo

---

detail	<i>Detail function provides a way to display the amount of members in each group</i>
--------	--

---

**Description**

The objective of this function is to summarize the overlaps across groups identified by venndetail without creating diagram.

**Usage**

```
detail(object)
```

```
## S4 method for signature 'Venn'  
detail(object)
```

**Arguments**

object                Venn object

**Value**

Numeric vector with set names and amounts for each set

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- vennndetail(list(A = A, B = B, C = C))
detail(res)
```

---

**dplot***Dplot function allows users to visualize the detail function in the form of a barplot*

---

**Description**

The amount of members within each group determined by `venndetail` will be displayed as a bar plot. This will include all groups such as shared, pairwise, and unique. The order of the figure can be adjusted by the users by using the `order` argument. The text size argument will allow users to change the size of the numbers above the bars indicating the total number of members within each group.

**Usage**

```
dplot(object, order = FALSE, textsize = 5)
```

```
## S4 method for signature 'Venn'
dplot(object, order = FALSE, textsize = 5)
```

**Arguments**

<code>object</code>	Venn object
<code>order</code>	Boolean indicating whether to sort the bar (default: FALSE).
<code>textsize</code>	Numeric vector giving the text size above the bar.

**Value**

Produces a bar plot displaying the total counts within each group

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- vennDetail(list(A = A, B = B, C = C))
dplot(res, order = TRUE, textsize = 3)
```

---

getFeature	<i>getFeature provides a way to combine list of user supplied data frames with Venn object</i>
------------	--

---

**Description**

GetFeature allows users to extract subsets from venn object into a table format along with accompanying information from the data frames provided in the rlist argument

**Usage**

```
getFeature(object, subset, rlist, userowname = TRUE, gind = NULL,
  sep = "_", wide = FALSE)
```

```
## S4 method for signature 'Venn'
getFeature(object, subset, rlist, userowname = TRUE,
  gind = NULL, sep = "_", wide = FALSE)
```

**Arguments**

object	Venn object
subset	Character vector giving the names of the user-defined subset to extract
rlist	List of user-supplied data frames to combine with vennDetail result
userowname	Boolean indicating whether to use row names to join data frames or not (default: TRUE)
gind	Column name or index of each user-supplied data.frame to use to join data frames(valid only when userowname=FALSE)
sep	Character string used to separate the terms when concatenating group names into new separation character for new column names in the resulting data frame
wide	Boolean indicating whether to use wide format(default:FALSE)

**Value**

data.frame with subsets information and details from the user supplied data frame

**Author(s)**

Kai Guo

**Examples**

```

A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
dA <- data.frame(A = A, "FC" = rnorm(40))
dB <- data.frame(B = B, "FC" = rnorm(60))
dC <- data.frame(C = C, "FC" = rnorm(40))
res <- venndetail(list(A = A, B = B, C = C))
rhs <- getFeature(res, subset = "Shared", rlist = list(dA, dB, dC),
  userowname= FALSE, gind = rep(1, 3))

```

---

getSet

*getSet function provides a way to extract subsets*


---

**Description**

getSet function provides a way to extract subsets from venndetail object

**Usage**

```
getSet(object, subset = NULL, min = 0, wide = FALSE)
```

```
## S4 method for signature 'Venn'
```

```
getSet(object, subset = NULL, min = 0, wide = FALSE)
```

**Arguments**

object	Venn object
subset	Character vector giving the subset names
min	The minimum number of input groups that a subset must belong to e.g. min = 2 will only report those subsets with elements shared by 2 or more input groups.
wide	Boolean indicating return wide format (default: FALSE).

**Value**

Specific subset information

**Author(s)**

Kai Guo

**Examples**

```

A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- venndetail(list(A = A, B = B, C = C))
getSet(res, "A")

```

---

make.subset	<i>Get subset from list of input groups</i>
-------------	---

---

**Description**

Get subset from list of input groups

**Usage**

```
make.subset(x, sep = "_")
```

**Arguments**

x	A list with input groups
sep	symbol character used when concatenating group names into subset names

**Value**

A list of subsets. The names on the list are the subset names and the list elements are the subset details.

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
x <- list(A = A, B = B, C = C)
out <- make.subset(x)
```

---

merge.Venn	<i>Merge two or more venndetail obejcts</i>
------------	---

---

**Description**

Merge will combine multiple venn diagrams to allow comparison between multiple groups

**Usage**

```
## S3 method for class 'Venn'
merge(x, y, ignore.case = FALSE, useupper = TRUE,
      plot = FALSE, ...)
```

**Arguments**

x	Venn object
y	Venn object
ignore.case	Boolean indicating whether to ignore case of group names (default: FALSE)
useupper	Boolean indicating whether to use uppercases for group names (default: TRUE)
plot	Boolean indicating whether to plot figure or not (default: FALSE)
...	arguments for vennndetail

**Value**

venn object

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res1 <- vennndetail(list(A = A, B = B))
res2 <- vennndetail(list(A = A, C = C))
res <- merge(res1, res2)
```

---

plot.Venn

*Plot Venn object*

---

**Description**

The plot function allows users to graphically display the groups and overlap between groups in their venn class object through a variety of graph types such as a bar plot, traditional venn, or venn pie chart.

**Usage**

```
## S3 method for class 'Venn'
plot(x, type = "venn", col = "black", sep = "_",
     mycol = c("dodgerblue", "goldenrod1", "darkorange1", "seagreen3",
              "orchid3"), cat.cex = 1.5, alpha = 0.5, cex = 2,
     cat.fontface = "bold", margin = 0.05, text.scale = c(1.5, 1.5, 1.5,
              1.5, 1.5, 1.5), filename = NULL, piecolor = NULL,
     revcolor = "lightgrey", any = NULL, show.number = TRUE,
     show.x = TRUE, log = FALSE, base = NULL, percentage = FALSE,
     sets.x.label = "Set Size", mainbar.y.label = "Intersection Size",
     nintersects = 40, abbr = FALSE, abbr.method = "both.sides",
     minlength = 3, ...)
```



**Arguments**

x	Venn object
type	Use venn, vennpie or upset (default: venn)
col	Character vector giving the color of the circles.
sep	Character string used to separate the terms when concatenating group names into new column names (colnames)(vennpie).
mycol	Character vector giving the filled color for VennDiagram circles.
cat.cex	Numeric vector giving the size of the category names.
alpha	A number giving the transparency value.
cex	A numerical value giving the text size for venndiagram
cat.fontface	A character giving the fontface (font style) for category name.
margin	Number giving the amount of whitespace around the diagram in grid units
text.scale	Numeric vector of text sizes for upset diagram (ylab, yaxis, xlab, subset name, xaxis, insetion).
filename	Filename for output figure.
piecolor	Character vector giving the colors of the subsets(vennpie).
revcolor	Character giving the color for the non-selected subsets(vennpie).
any	Number to indicate selected subsets, such as 1 means any unique subsets, 2 means any subsets shared by two groups(vennpie).
show.number	Boolean indicating whether to display the element numbers of the subsets or not (default: TRUE)(vennpie).
show.x	Boolean indicating whether to show subset labels outside the circle (default: TRUE)(vennpie).
log	Boolean indicating whether to transform the data in log scale(vennpie).
base	Base value for log transformation(vennpie).
percentage	Boolean indicating whether to display subset percentages (default: FALSE)(vennpie).
sets.x.label	x-axis label (upset)
mainbar.y.label	y-axis label (upset)
nintersects	Number of intersections to plot. If subset to NA, all intersections will be plotted.
abbr	Boolean indicating whether to abbreviate subset names (default: FALSE).
abbr.method	a character string specifying the method used. Partial matches allowed. (default: both side).
minlength	Minmal length for the subset name.
...	further arguments passed to or from other methods

**Value**

different type of graphics based on user chose

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- venndetail(list(A = A, B = B, C = C))
plot(res, type = "venn")
```

---

`result`*Extract the result from venn object*

---

**Description**

Result will return output in a table format including the contents of the subsets included in the `venndetail` object

**Usage**

```
result(object, wide = FALSE)

## S4 method for signature 'Venn'
result(object, wide = FALSE)
```

**Arguments**

<code>object</code>	Venn object
<code>wide</code>	Boolean indicating whether to return wide format(default:FALSE)

**Value**

return dataframe and print header of dataframe

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- venndetail(list(A = A, B = B, C = C))
result <- result(res)
```

---

rowjoin	<i>Join data.frame based on rownames</i>
---------	--

---

**Description**

join two dataframes by rownames

**Usage**

```
rowjoin(x, y, fun = "fun_join")

## S4 method for signature 'data.frame,data.frame'
rowjoin(x, y, fun = "full_join")
```

**Arguments**

x	data.frame x
y	data.frame y
fun	Different join format: left_join, full_join, right_join (default:full_join)

**Value**

dataframe with join results

**Author(s)**

Kai Guo

**Examples**

```
library(dplyr)
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
dA <- data.frame(A = A, "FC" = rnorm(40))
dB <- data.frame(B = B, "FC" = rnorm(60))
rownames(dA) <- A
rownames(dB) <- B
rowjoin(dA, dB)
```

---

setcolor	<i>return colors with given a vector</i>
----------	--

---

**Description**

Setcolor will provide a list of color vectors based on the number used as an input.

**Usage**

```
setcolor(x)
```

**Arguments**

x	Number of color
---	-----------------

**Value**

color vector

**Author(s)**

Kai Guo

**Examples**

```
mycol <- setcolor(10)
mycol
```

---

show Venn	<i>Show the summary of venn object</i>
-----------	--

---

**Description**

This function provides a summary of the venn object, including a full results and subsets as well as an summary information.

**Usage**

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

**Arguments**

object	venn object
--------	-------------

**Value**

summary information for the venn object

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- venndetail(list(A = A, B = B, C = C))
show(res)
```

---

summary.Venn

*Give summary information of Venn object*

---

**Description**

print the summary information of Venn object

**Usage**

```
## S3 method for class 'Venn'
summary(object, ...)
```

**Arguments**

object            Venn object  
...                other arguments ignored (for compatibility with generic)

**Value**

summary information

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- venndetail(list(A = A, B = B, C = C))
summary(res)
```

---

T2DM

*T2DM*


---

**Description**

T2DM data are differential expression genes (DEGs) with annotation from the publication by Hinder et al. The data contains three DEG sets from three different tissues (Cortex,SCN,Glom). DEGs were determined by using Cuffdiff with a false discovery rate (FDR) < 0.05 between groups with or without pioglitazone treatment.

**Usage**

T2DM

**Format**

A list of data frame with five columns individually:

**Entrez** Entrez gene IDs**Symbol** HGNC symbols**Annotation** Gene function**log2FC** log2 Fold Change**FDR** False Discovery Rate**Examples**

T2DM

---

Venn-class

*Class 'Venn' This class includes all information from venndetail*


---

**Description**

Class 'Venn' This class includes all information from venndetail

**Slots**

input original input datasets

raw summary of the input datasets

sep separation character

GroupNames input group names

result shared or unique sets

detail shared of unique number belongs to each sets

wide result in wide format

**Author(s)**

Kai Guo

---

`venndetail`*Extract shared and unique subsets*

---

**Description**

Extracts shared and unique elements from groups provided to the function. This base function will create a formal class venn object and can also graphically plot the amount of objects in each group. The plot will be in the form of a traditional venn diagram as default. And users can also use `vennpie` or `upset` methods to display the result.

**Usage**

```
venndetail(x, sep = "_", abbr = FALSE, minlength = 3,  
           abbr.method = "both side")
```

**Arguments**

<code>x</code>	A list of variables with group names.
<code>sep</code>	symbol character used when concatenating group names into subset names (default: <code>'_'</code> ).
<code>abbr</code>	Boolean indicating whether to abbreviate subset names (default: <code>FALSE</code> ).
<code>minlength</code>	Minimal length for the subset name.
<code>abbr.method</code>	a character string specifying the method used. Partial matches allowed. (default: <code>both side</code> ).

**Details**

Extract shared and unique subsets

**Value**

venn object and figures

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)  
B <- sample(1:100, 60, replace = FALSE)  
C <- sample(1:100, 40, replace = FALSE)  
res <- venndetail(list(A = A, B = B, C = C))
```

---

 vennpie

*Pie plot shows shared and unique sets*


---

### Description

Vennpie uses the venn object and to creates a figure in the form of a venn pie diagram rather than a traditional venn diagram. Users can highlight a specific sections of the venn pie.

### Usage

```
vennpie(object, subset = NULL, top = 31, min = 0, color = NULL,
  revcolor = "lightgrey", any = NULL, show.number = TRUE,
  show.x = TRUE, sep = "_", log = FALSE, base = NULL,
  percentage = FALSE)
```

```
## S4 method for signature 'Venn'
vennpie(object, subset = NULL, top = 31, min = 0,
  color = NULL, revcolor = "lightgrey", any = NULL,
  show.number = TRUE, show.x = TRUE, sep = "_", log = FALSE,
  base = NULL, percentage = FALSE)
```

### Arguments

object	Venn object
subset	Character vector giving the subset users want to highlight.
top	number of subsets with largest to display (default: 31)
min	The minimum number of input groups that a subset must belong to e.g. min = 2 will only report those subsets with elements shared by 2 or more input groups.
color	Character vector giving the colors of the subsets.
revcolor	Character giving the color for the non-selected subsets.
any	Number to indicate selected subsets, such as 1 means any unique subsets, 2 means any subsets shared by two groups.
show.number	Boolean indicating whether to display the element numbers of the subsets or not (default: TRUE).
show.x	Boolean indicating whether to show subset labels outside the circle (default: TRUE).
sep	Character string used to separate the terms when concatenating group names into new column names (colnames).
log	Boolean indicating whether to transform the data in log scale .
base	Base value for log transformation.
percentage	Boolean indicating whether to display subset percentages (default: FALSE).



**Value**

vennpie figure

**Author(s)**

Kai Guo

**Examples**

```
A <- sample(1:100, 40, replace = FALSE)
B <- sample(1:100, 60, replace = FALSE)
C <- sample(1:100, 40, replace = FALSE)
res <- venndetail(list(A = A, B = B, C = C))
vennpie(res)
```

# Index

- \* **classes**
  - Venn-class, [14](#)
- \* **datasets**
  - T2DM, [14](#)
- .add\_colnames, [2](#)
- .make\_table, [3](#)
  
- detail, [3](#)
- detail, Venn-method (detail), [3](#)
- dplot, [4](#)
- dplot, Venn-method (dplot), [4](#)
  
- getFeature, [5](#)
- getFeature Venn (getFeature), [5](#)
- getFeature, Venn-method (getFeature), [5](#)
- getSet, [6](#)
- getSet, Venn-method (getSet), [6](#)
  
- make\_subset, [7](#)
- merge.Venn, [7](#)
  
- plot.Venn, [8](#)
  
- result, [10](#)
- result, Venn-method (result), [10](#)
- rowjoin, [11](#)
- rowjoin, data.frame, data.frame-method (rowjoin), [11](#)
  
- setcolor, [12](#)
- show Venn, [12](#)
- show, Venn-method (show Venn), [12](#)
- summary.Venn, [13](#)
  
- T2DM, [14](#)
  
- Venn-class, [14](#)
- venndetail, [15](#)
- vennpie, [16](#)
- vennpie, Venn-method (vennpie), [16](#)