

# Package ‘NetWeaver’

January 11, 2017

**Type** Package

**Title** Graphic Presentation of Complex Genomic and Network Data  
Analysis

**Version** 0.0.2

**Date** 2016-12-20

**Author** Minghui Wang, Bin Zhang

**Maintainer** Minghui Wang <m.h.wang@live.com>

**Description** Implements various simple function utilities and flexible pipelines to generate circular images for visualizing complex genomic and network data analysis features.

**Depends** R (>= 3.3.0)

**License** GPL-3

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-01-11 01:56:08

## R topics documented:

Modules . . . . .	2
NetWeaver . . . . .	3
rc.get.coordinates . . . . .	4
rc.get.params . . . . .	5
rc.initialize . . . . .	5
rc.plot.area . . . . .	7
rc.plot.grColLegend . . . . .	8
rc.plot.heatmap . . . . .	9
rc.plot.histogram . . . . .	10
rc.plot.ideogram . . . . .	12
rc.plot.line . . . . .	13
rc.plot.link . . . . .	14

rc.plot.mHistogram . . . . .	15
rc.plot.ribbon . . . . .	16
rc.plot.text . . . . .	17
rc.plot.track . . . . .	18
rc.plot.track.id . . . . .	19
rc.reset.params . . . . .	20
rc.track.pos . . . . .	20
<b>Index</b>	<b>22</b>

---

Modules	<i>Example Module Dataset</i>
---------	-------------------------------

---

## Description

This example dataset and sample code illustrate the use of NetWeaver to visualize the complex features of gene coexpression network modules.

## Usage

```
data(Modules)
```

## Details

This dataset contains a data.frame called `Modules`. Each row is a module, with the module id in the first column. The second column is ranking score. The next 4 columns are coefficients of module-trait correlations. The rest columns are P values of enrichment for various gene signatures. The sample R code shows a pipeline for visualizing the module feature data using a circos style plot.

## Author(s)

Minghui Wang <m.h.wang@live.com>, Bin Zhang <bin.zhang@mssm.edu>

## See Also

[rc.initialize](#)

## Examples

```
## Not run:
#see a tutorial for how to plot this dataset
vignette("netweaver")

## End(Not run)
```

## Description

NetWeaver is motivated towards developing a simple and flexible pipeline for visualizing the complex features of enrichment and correlation of gene coexpression network modules. While circos style 2D track plot is one natural choice for such practice, existing packages are designed primarily for handling genome structure and intervals. They are either too complicated to use, requiring certain level of knowledge of scripting, or limited in applications to only genomic structure data. To address these issues, particularly extend beyond applications in genomic structure data, NetWeaver offers a lightweight implementation of circular track plot, providing simple and flexible R function utilities and pipelines to generate circular images for visualizing different types of structure/relationship data.

The original version of this package was developed for Figure 7 of Wang et al (2016) *Genome Medicine* 8:104, which illustrates more than 20 properties for 50 coexpression network modules with a circular track plot. Please try to cite the paper when you use results from this software in a publication:

Wang M, Roussos P, McKenzie A, Zhou X, Kajiwara Y, Brennand K, DeLuca GC, Crary JF, Casaccia P, Buxbaum J et al. 2016. Integrative Network Analysis of Nineteen Brain Regions Identifies Molecular Signatures and Networks Underlying Selective Regional Vulnerability to Alzheimer's Disease. *Genome Medicine* 8: 104.

## Details

There are two sample pipelines available:

1. Analysis of a real dataset of gene coexpression network modules can be reached through `vignette("netweaver")`.
2. Analysis of a hypothetical data is shown in sample code of `rc.initialize`.

## Author(s)

Minghui Wang <m.h.wang@live.com>, Bin Zhang <bin.zhang@mssm.edu>

## References

Wang M, Roussos P, McKenzie A, Zhou X, Kajiwara Y, Brennand K, DeLuca GC, Crary JF, Casaccia P, Buxbaum J et al. 2016. Integrative Network Analysis of Nineteen Brain Regions Identifies Molecular Signatures and Networks Underlying Selective Regional Vulnerability to Alzheimer's Disease. *Genome Medicine* 8: 104.

**See Also**

[rc.initialize](#), [Modules](#)

---

rc.get.coordinates      *Get Coordinates*

---

**Description**

Retrieve x and y coordinates in a track.

**Usage**

```
rc.get.ringCoordinates(track.id, Start, End, Chr=NULL, degree=NULL, ringThickness=NULL)
rc.get.coordinates(track.id, Pos, Chr=NULL, degree=NULL, innerSide=TRUE)
```

**Arguments**

track.id	number of track from outermost.
Start, End	start/end position on a chromosome or since first chromosome. See Details.
Pos	a vector of positions on a chromosome or since first chromosome. See Details.
Chr	a chromosome id. See Details.
degree	the angle of the arc rotation, overwrites Start, End, Pos and Chr.
ringThickness	thickness of a ring.
innerSide	whether to compute coordinates on the inner side of a track. Outer side coordinates will be returned if FALSE.

**Details**

If Chr is NULL, Start, End and Pos will be considered as cumulative positions since first chromosome.

**Value**

A list of x and y coordinates.

**Author(s)**

Minghui Wang <m.h.wang@live.com>

**See Also**

[rc.initialize](#), [rc.reset.params](#)

**Examples**

```
##not run###
```

---

rc.get.params	<i>Get Parameters</i>
---------------	-----------------------

---

**Description**

Retrieve parameters of Circos plot.

**Usage**

```
rc.get.params()
```

**Value**

A list.

**Author(s)**

Minghui Wang <m.h.wang@live.com>

**See Also**

[rc.initialize](#), [rc.reset.params](#)

**Examples**

```
##not run###
```

---

rc.initialize	<i>Initialize Circos Plot Parameters</i>
---------------	--

---

**Description**

Setup default parameter for Circos plot.

**Usage**

```
rc.initialize(cyto.info, num.tracks=NULL, chr.order=NULL,  
stepUnit=10^7, params=list())
```

**Arguments**

cyto.info	data.frame, see Details.
num.tracks	integer, number of tracks.
chr.order	character vector of chromosome ids specifying the ordering of chromosomes.
stepUnit	integer, smoothing factor for faster plotting.
params	a list of named items. See Details.

## Details

cyto.info is a data.frame of chromosomal position ordered cytobands, with columns: Chr, Start, End, Stain, and any additional information (like band color), where Chr is chromosome name, Start and End are the start and end positions on the chromosome, and Stain is the cyto stain. The stain is normally one of the "gneg", "acen", "stalk", "gvar", "gpos", "gpos100", "gpos75", "gpos66", "gpos50", "gpos33", and "gpos25", which will be plotted by color "white", "red", "steelblue", "light-grey", "black", "black", "gray40", "gray50", "gray60", "gray70" and "gray80", accordingly. Customized colors for the cyto bands can be specified in an additional column named "BandColor".

Additional plot settings can be specified through argument params, including: color.line (color for lines and links, default "black"), chr.padding (padding between chromosomes is a fraction of the total chromosome sizes, default 0.1), track.padding (padding between tracks is a fraction of the track height, default 0.1), track.height (track height, default 0.15), and radius (radius of the circles, default 1).

After initialization, the parameter settings can be retrieved by [rc.get.params](#).

Noted that while cyto.info requires input to be in a form of chromosome cytobands, the input is not limited to genomic features. As illustrated in example data [Modules](#), complex features of gene coexpression network modules can also be plotted with the current circos visualization technique.

## Author(s)

Minghui Wang <m.h.wang@live.com>

## See Also

[rc.get.params](#), [rc.reset.params](#), [rc.plot.ideogram](#), [rc.plot.histogram](#), [rc.plot.mHistogram](#), [rc.plot.link](#), [rc.plot.ribbon](#), [Modules](#)

## Examples

```
library(NetWeaver)
options(stringsAsFactors=FALSE)
## set fake Cyto data
Cyto=data.frame(Chr=paste0('C',1:20), Start=1, End=100, BandColor=colors()[2:21])

## initialize circos
#firstly set number of tracks, should be larger than the actual number of tracks
#used for plotting in order to leave sufficient sapce in the middle
num.tracks=8
rc.initialize(Cyto, num.tracks=num.tracks, params=list(chr.padding=0.1))

##retrieve parameters
params=rc.get.params()
#make plot area
rc.plot.area()

##plot ideogram on track 1 and 2 (start from the outside to inside)
```

```

track.num=1:2
rc.plot.ideogram(track.num, plot.band=TRUE, plot.chromosome.id=TRUE)

##plot histograms that span multiple chromosomes
MultHistData=data.frame(Chr1=paste0('C',seq(1,18,3)), Start1=50,
Chr2=paste0('C',seq(3,20,3)), End2=20, Col=1:6)
track.num=3
rc.plot.mHistogram(MultHistData, track.id=track.num, data.col=5,
color.col=5, track.border=NA, fixed.height=TRUE)

##plot histogram that occupies two tracks 4 and 5
HistData=data.frame(Chr=paste0('C',1:20), Start=1, End=50, Data=runif(20))
params$color.hist <- 'black'
rc.reset.params(params)
track.num=5
rc.plot.histogram(HistData, track.num, data.col=4, fixed.height=FALSE,
custom.track.height=params$track.height*2)

##plot heatmap on track 6
HeatData=data.frame(Chr=paste0('C',1:20), Start=1,End=100, Data=1:20)
colfuncHeat=function(n) rev(heat.colors(n))
track.num=track.num+1
rc.plot.histogram(HeatData, track.num, data.col=4, color.gradient=colfuncHeat(50),
fixed.height=TRUE)

##plot links in the middle
LinkData=data.frame(Chr1=sample(Cyto$Chr,40,replace=TRUE), Pos1=20,
Chr2=sample(Cyto$Chr,40,replace=TRUE),Pos2=20, Data=runif(20))
LinkData=LinkData[LinkData$Chr1 != LinkData$Chr2,]
params$color.line='blue'
rc.reset.params(params)
track.num=track.num+1
rc.plot.link(LinkData, track.num, data.col=4)
ribbonData=data.frame(Chr1=c('C1','C3'), Start1=c(10,10), End1=c(40,40),
Chr2=c('C17','C10'), Start2=20, End2=60, Col=c('red','brown'))
rc.plot.ribbon(ribbonData, track.num, color.col='Col', twist=TRUE)
#label track id
rc.plot.track.id(2:6, col=2)
#add text label
rc.plot.text(data.frame(Chr='C3',Pos=50,Label='GeneX'), track.id=3,srt=45,cex=0.8,col='blue')
#add line mark
rc.plot.line(data.frame(Chr='C19',Pos=seq(10,90,by=10),Col='red'), track.id=3, color.col=3)

```

---

rc.plot.area

*Setup Plot Area*


---

## Description

Create new graphics frame for circle plot .

**Usage**

```
rc.plot.area(size=1, oma=rep(0,4), mar=rep(0,4))
```

**Arguments**

size	a value between 0 to 1, specifying the effective size of the circle plot area in the current window. The smaller the size, the larger the blank area around the circle plot.
oma	A vector of the form <code>c(bottom, left, top, right)</code> giving the size of the outer margins in lines of text.
mar	A numerical vector of the form <code>c(bottom, left, top, right)</code> which gives the number of lines of margin to be specified on the four sides of the plot.

**Details**

After setting up, the extremes of the user coordinates of the plotting region can be found out by `par('usr')`.

**Author(s)**

Minghui Wang <m.h.wang@live.com>

**See Also**

[rc.initialize](#), [rc.reset.params](#)

**Examples**

```
##see example of rc.initialize() & ?Modules
```

---

rc.plot.grColLegend     *Plot Gradient Color Legend*

---

**Description**

Plot gradient color legend at given position.

**Usage**

```
rc.plot.grColLegend(x, y, cols, at=NULL, legend=at,  
title='Color', width=0.1, height=0.3, gap=0,  
direction=c('vertical', 'horizontal'), cex.text=1, cex.title=1)
```



**Arguments**

<code>x, y</code>	numeric values of coordinates where the legend is plotted. See details.
<code>cols</code>	a vector of colors.
<code>at</code>	integers; index legend text in the color vector <code>cols</code> . See details.
<code>legend</code>	character strings of legend text with the same length of vector <code>at</code> .
<code>title</code>	character string of the legend title.
<code>width</code>	numeric, width of the color legend bar.
<code>height</code>	numeric, height of the color legend bar.
<code>gap</code>	numeric, size of the gap between two color blocks.
<code>direction</code>	plot the color legend bar in either vertical or horizontal direction.
<code>cex.text</code>	numeric, magnitude of the legend text.
<code>cex.title</code>	numeric, magnitude of the legend title.

**Details**

`x` and `y` set the coordinates of bottom left starting point. The extremes of the user coordinates of the plotting region can be found out by `par('usr')`.

By default NULL, parameter `at` will be set as a vector with values 1, `ceiling(length(cols)/2)`, and `length(cols)`.

**Author(s)**

Minghui Wang <m.h.wang@live.com>

**See Also**

[rc.plot.link](#), [rc.plot.histogram](#)

**Examples**

```
##not run###
```

---

<code>rc.plot.heatmap</code>	<i>Plot Heat-map</i>
------------------------------	----------------------

---

**Description**

Add multiple tracks of heat-map on a circos plot.

**Usage**

```
rc.plot.heatmap(Data, track.id, color.gradient=NULL,
track.color=NA, track.border=NULL)
```

### Arguments

Data	matrix, data to be plotted. See Details.
track.id	integer, starting track id.
color.gradient	a vector of gradient colors. See details.
track.color	NA or color code for the track background.
track.border	the color to draw the track border. Use NA to omit borders.

### Details

This function employs `rc.plot.histogram` to plot heat-map. Every row of Data will be plotted as heat-map in one track. The column names of Data must be the chromosome ids.

If `color.gradient` is not NULL, the data will be scaled to positive integers in the range of 1~length(`color.gradient`) to index the colors in vector `color.gradient`. If `color.gradient` is NULL, the input Data must be a matrix of colors or anything that can be converted to colors.

### Author(s)

Minghui Wang <m.h.wang@live.com>

### See Also

[rc.plot.histogram](#)

### Examples

```
##see example of ?Modules
```

---

rc.plot.histogram	<i>Plot Histogram, Barplot or Heat-map</i>
-------------------	--

---

### Description

Add a track of histogram, barplot, or heat-map on a circos plot.

### Usage

```
rc.plot.histogram(Data, track.id, data.col=NULL, color.col=NULL,  
color.gradient=NULL, fixed.height=FALSE, track.color=NA, track.border=NULL,  
custom.track.height=NULL)
```

**Arguments**

Data	data.frame, data to be plotted. See Details.
track.id	integer, in which track to plot.
data.col	integer, specifying the column that contains the data for plotting.
color.col	integer, specifying the column that contains the color.
color.gradient	a vector of gradient colors. See details.
fixed.height	logical, whether to fix the histogram height which results in heat-map. TRUE if data.col is NULL.
track.color	NA or color code for the track background.
track.border	the color to draw the track border. Use NA to omit borders.
custom.track.height	NULL or numeric, specifying customized track height. See details.

**Details**

Data must have at least three columns. The first three columns must be named as Chr, Start and End, specifying the chromosomes and positions for the start and end points of the links. Additional columns can be used to specify data and color.

If color.gradient is not NULL, color.col is ignored and the data will be scaled to positive integers in the range of 1~length(color.gradient) to index the colors in vector color.gradient.

custom.track.height bypasses default track height, which will be useful for track(s) requiring bigger/smaller height than default. The default track height can be retrived by function [rc.get.params](#) after running [rc.initialize](#).

**Author(s)**

Minghui Wang <m.h.wang@live.com>

**See Also**

[rc.plot.link](#), [rc.plot.mHistogram](#)

**Examples**

```
##see example of rc.initialize() & ?Modules
```

---

rc.plot.ideogram      *Plot Ideogram*

---

### Description

Add chromosome ideogram on circos plot.

### Usage

```
rc.plot.ideogram(track.ids, plot.band=TRUE, plot.chromosome.id=TRUE,  
chrom.alias=NULL, color.chromosome.id=NULL, cex.text=1, track.border=NULL,  
las=0, custom.track.height=NULL)
```

### Arguments

track.ids	vector, two integers specifying the two tacks for plotting chromosome name and cyto band respectively.
plot.band	logic, whether to plot ideogram cyto band.
plot.chromosome.id	logic, whether to plot chromosome id.
chrom.alias	NULL or a chromosome named vector of alias.
color.chromosome.id	NULL or a chromosome named vector of colors.
cex.text	numeric, scale of text.
track.border	the color to draw the track border. Use NA to omit borders.
las	numeric in 0,1,2,3; the style of chromosome labels. 0, always parallel to the track [default]; 1, always horizontal; 2, always perpendicular to the track; 3, always vertical.
custom.track.height	NULL or numeric, specifying customized track height when plotting cyto band.

### Author(s)

Minghui Wang <m.h.wang@live.com>

### Examples

```
##see example of rc.initialize() & ?Modules
```

---

rc.plot.line	<i>Plot Line</i>
--------------	------------------

---

### Description

Plot lines at designated position.

### Usage

```
rc.plot.line(Data, track.id, color.col=NULL, custom.length=NULL, ...)
```

### Arguments

Data	a data.frame. See Details.
track.id	a vector of integers, specifying the tracks for plotting line.
color.col	integer, specifying the column that contains the color.
custom.length	NULL or numeric, specifying line length.
...	further graphical parameters (from <a href="#">par</a> ), such as lty and lwd.

### Details

Data is a data.frame, with at least two columns named 'Chr' and 'Pos', specifying the chromosomal positions for each line in every row. An additional column can be used to specify the line color. Default line length is determined by track height.

### Author(s)

Minghui Wang <m.h.wang@live.com>

### See Also

[rc.plot.histogram](#), [rc.plot.track](#)

### Examples

```
##see example of rc.initialize() & ?Modules
```

---

rc.plot.link	<i>Plot Link</i>
--------------	------------------

---

### Description

Add a track of links on circos plot.

### Usage

```
rc.plot.link(Data, track.id, data.col=NULL, color.col = NULL,  
             max.lwd=1, sort.links=TRUE)
```

### Arguments

Data	data.frame of at least four columns. See Details.
track.id	integer, the track number for plotting.
data.col	integer or character string, specifies the column of data.
color.col	integer or character string, specifies the column of colors.
max.lwd	integer, maximum line width.
sort.links	logical, whether to re-assign link start and end positions by sorting them. See details

### Details

Data must have at least four columns: Chr1, Pos1, Chr2 and Pos2, specifying the chromosomes and positions for the start and end points of the links. Additional columns may be used to specify the data, color and max line width.

If data.col is not NULL, the line width will be proportional to the values in column data.col with the maximum line width determined by max.lwd.

If sort.links is true, the links originate from the same chromosome will be sorted and equally spaced to minimize crossing.

### Author(s)

Minghui Wang <m.h.wang@live.com>

### See Also

[rc.plot.histogram](#), [rc.plot.ribbon](#)

### Examples

```
##see example of rc.initialize()
```

---

rc.plot.mHistogram      *Plot Histogram or Barplot Across Chromosomes*

---

## Description

Plot histogram or barplot that overlaps with multiple chromosomes.

## Usage

```
rc.plot.mHistogram(Data, track.id, data.col, color.col=NULL, color.gradient=NULL,
  fixed.height=FALSE, track.color=NA, track.border=NULL, custom.track.height=NULL)
```

## Arguments

Data	data.frame of at least four columns. See Details.
track.id	integer, the track number for plotting.
data.col	integer or character string, specifies the column of data.
color.col	integer or character string, specifies the column of colors.
color.gradient	a vector of gradient colors.
fixed.height	logical, whether to fix the histogram height, resulting in heat-map if TRUE.
track.color	NA or color code for the track background.
track.border	the color to draw the track border. Use NA to omit borders.
custom.track.height	NULL or numeric, specifying customized track height. See details.

## Details

Data must have at least four columns: Chr1, Start1, Chr2 and End2, specifying the chromosomes and positions for the start and end points. Additional columns may be used to specify the data, color, etc.

If `color.gradient` is not NULL, it will supersede `color.col`.

`custom.track.height` bypasses default track height, which will be useful for track(s) requiring bigger/smaller height than default. The default track height can be retrieved by function [rc.get.params](#) after running [rc.initialize](#).

## Author(s)

Minghui Wang <m.h.wang@live.com>

## See Also

[rc.plot.link](#), [rc.plot.histogram](#)

## Examples

```
##see example of rc.initialize()
```

---

rc.plot.ribbon	<i>Plot Ribbon</i>
----------------	--------------------

---

## Description

Add a track of ribbons on circos plot.

## Usage

```
rc.plot.ribbon(Data, track.id, color.col = NULL, twist=FALSE)
```

## Arguments

Data	data.frame of at least six columns. See Details.
track.id	integer, the track number for plotting.
color.col	integer or character string, specifies the column of colors.
twist	logic, whehter to twist the ribbon.

## Details

Data must have at least six columns: Chr1, Start1, End1, Chr2, Start2 and End2, specifying the chromosomes and positions for the start and end points of the ribbons. Additional columns may be used to specify the color.

## Author(s)

Minghui Wang <m.h.wang@live.com>

## See Also

[rc.plot.histogram](#), [rc.plot.link](#)

## Examples

```
##see example of rc.initialize()
```



---

rc.plot.text	<i>Plot Text</i>
--------------	------------------

---

### Description

Plot text labels at designated position.

### Usage

```
rc.plot.text(Data, track.id, col='black', custom.track.height=NULL, ...)
```

### Arguments

Data	a data.frame. See Details.
track.id	a vector of integers, specifying the tracks for plotting text.
col	color for the text.
custom.track.height	NULL or numeric, specifying customized track height.
...	further graphical parameters (from <a href="#">par</a> ), such as srt and family.

### Details

Data is a data.frame with at least three columns named 'Chr', 'Pos', and 'Label', specifying the chosomal positoin and text labels.

### Author(s)

Minghui Wang <m.h.wang@live.com>

### See Also

[rc.plot.histogram](#), [rc.plot.track](#)

### Examples

```
##see example of rc.initialize() & ?Modules
```

---

rc.plot.track	<i>Plot Track</i>
---------------	-------------------

---

### Description

Add a track to a circo plot.

### Usage

```
rc.plot.track(track.id, border='black', col=NA, custom.track.height=NULL)
```

### Arguments

track.id	integer, track id.
border	the color to draw the track border. Use NA to omit borders.
col	NA or color code for the track background.
custom.track.height	NULL or numeric, specifying customized track height. See details.

### Details

custom.track.height sets customized track height, which will be useful for track(s) requiring bigger/smaller height than default.

### Author(s)

Minghui Wang <m.h.wang@live.com>

### See Also

[rc.plot.histogram](#), [rc.plot.mHistogram](#)

### Examples

```
##see example of rc.initialize() & ?Modules
```

---

rc.plot.track.id	<i>Plot Track Id</i>
------------------	----------------------

---

### Description

Plot labels in designated tracks.

### Usage

```
rc.plot.track.id(track.id, labels=NULL, degree=0,  
col='black', custom.track.height=NULL, ...)
```

### Arguments

track.id	a vector of integers, specifying the tracks for plotting id.
labels	NULL or a vector of character string, specifying the text to be written.
degree	the angle of the arc rotation.
col	color for the text.
custom.track.height	NULL or numeric, specifying customized track height.
...	further graphical parameters (from par), such as srt and family.

### Details

If labels is NULL, values of track.id will be used as text labels.

### Author(s)

Minghui Wang <m.h.wang@live.com>

### See Also

[rc.plot.histogram](#), [rc.plot.track](#)

### Examples

```
##see example of rc.initialize() & ?Modules
```

---

rc.reset.params      *Reset Circos Plot Parameters*

---

**Description**

Reset parameters for Circos plot.

**Usage**

```
rc.reset.params(params)
```

**Arguments**

params      list.

**Author(s)**

Minghui Wang <m.h.wang@live.com>

**See Also**

[rc.initialize](#), [rc.get.params](#)

**Examples**

```
##not run###
```

---

rc.track.pos      *Get Track Positions*

---

**Description**

Calculate inner and outer radius of a track.

**Usage**

```
rc.track.pos(track.id)
```

**Arguments**

track.id      integer, number of the track from the outermost.

**Value**

A vector with two radius values: out.pos and in.pos.

*rc.track.pos*

21

**Author(s)**

Minghui Wang <m.h.wang@live.com>

**See Also**

[rc.initialize](#), [rc.reset.params](#)

**Examples**

```
##not run###
```

# Index

## \*Topic **datasets**

Modules, [2](#)

Modules, [2](#), [4](#), [6](#)

NetWeaver, [3](#)

NetWeaver-package (NetWeaver), [3](#)

par, [13](#), [17](#)

rc.get.coordinates, [4](#)

rc.get.params, [5](#), [6](#), [11](#), [15](#), [20](#)

rc.get.ringCoordinates  
(rc.get.coordinates), [4](#)

rc.initialize, [2–5](#), [5](#), [8](#), [11](#), [15](#), [20](#), [21](#)

rc.plot.area, [7](#)

rc.plot.grColLegend, [8](#)

rc.plot.heatmap, [9](#)

rc.plot.histogram, [6](#), [9](#), [10](#), [10](#), [13–19](#)

rc.plot.ideogram, [6](#), [12](#)

rc.plot.line, [13](#)

rc.plot.link, [6](#), [9](#), [11](#), [14](#), [15](#), [16](#)

rc.plot.mHistogram, [6](#), [11](#), [15](#), [18](#)

rc.plot.ribbon, [6](#), [14](#), [16](#)

rc.plot.text, [17](#)

rc.plot.track, [13](#), [17](#), [18](#), [19](#)

rc.plot.track.id, [19](#)

rc.reset.params, [4–6](#), [8](#), [20](#), [21](#)

rc.track.pos, [20](#)