

# Package ‘CCPlotR’

April 5, 2026

**Title** Plots For Visualising Cell-Cell Interactions

**Version** 1.9.1

**Description** CCPlotR is an R package for visualising results from tools that predict cell-cell interactions from single-cell RNA-seq data. These plots are generic and can be used to visualise results from multiple tools such as Liana, CellPhoneDB, NATMI etc.

**Imports** plyr, tidyr, dplyr, ggplot2, forcats, ggraph, igraph, scatterpie, circlize, ComplexHeatmap, tibble, grid, stringr, ggtext, ggh4x, patchwork, RColorBrewer, scales, viridis, grDevices, graphics, stats, methods

**URL** <https://github.com/Sarah145/CCPlotR>

**BugReports** <https://github.com/Sarah145/CCPlotR/issues>

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** false

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.3

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

**VignetteBuilder** knitr

**biocViews** SingleCell, Network, Visualization, CellBiology, SystemsBiology

**Config/testthat/edition** 3

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

**git\_branch** devel

**git\_last\_commit** 6e010b7

**git\_last\_commit\_date** 2026-03-16

**Repository** Bioconductor 3.23

**Date/Publication** 2026-04-05

**Author** Sarah Ennis [aut, cre] (ORCID: <<https://orcid.org/0000-0001-6100-8573>>),  
Pilib Ó Broin [aut],  
Eva Szegezdi [aut]

**Maintainer** Sarah Ennis <ennissarah94@gmail.com>

## Contents

cc_arrow . . . . .	2
cc_circos . . . . .	3
cc_dotplot . . . . .	4
cc_heatmap . . . . .	5
cc_network . . . . .	6
cc_sigmoid . . . . .	7
geom_sigmoid . . . . .	7
paletteMartin . . . . .	9
StatSigmoid . . . . .	9
toy_data . . . . .	9
toy_exp . . . . .	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

cc_arrow	<i>Paired Arrow Plot Function</i>
----------	-----------------------------------

---

### Description

This function plots interactions between a pair of cell types

### Usage

```
cc_arrow(
  cc_df,
  cell_types = NULL,
  option = "A",
  n_top_ints = 15,
  exp_df = NULL,
  colours = setNames(paletteMartin(n = 2), cell_types),
  palette = "BuPu"
)
```

### Arguments

cc_df	A dataframe with columns 'source', 'target', 'ligand', 'receptor' and 'score'. See toy_data for example.
cell_types	A vector of which two cell types to plot.
option	Either 'A' or 'B'. Option A will plot the top n_top_ints interactions between cell_types and their scores. Option B will plot the top n_top_ints interactions between cell_types, their scores and the expression of the ligand/receptor genes in the sender/receiver cell types.
n_top_ints	The number of top interactions to plot.
exp_df	A dataframe containing the mean expression values for each ligand/receptor in each cell type. See toy_exp for an example. Only required for option B.

colours	A named vector of colours for each cell type. Default is paletteMartin(), a colourblind-friendly palette. Only used for option A.
palette	Which colour palette to use to show the mean expression. Should be one of the RColorBrewer sequential palettes. Only used for option B.

**Value**

Returns a plot generated with the ggplot2 package

**Examples**

```
data(toy_data, toy_exp, package = 'CCPlotR')
cc_arrow(toy_data, cell_types = c("B", "CD8 T"), colours = c(`B` = "hotpink", `CD8 T` = "orange"))
cc_arrow(toy_data,
  cell_types = c("NK", "CD8 T"), option = "B", exp_df = toy_exp,
  n_top_ints = 10, palette = "OrRd"
)
```

---

 cc\_circos

*Circos Plot Function*


---

**Description**

This function creates a circos plot

**Usage**

```
cc_circos(
  cc_df,
  option = "A",
  n_top_ints = 15,
  exp_df = NULL,
  cell_cols = NULL,
  palette = "BuPu",
  cex = 1,
  show_legend = TRUE,
  scale = FALSE,
  ...
)
```

**Arguments**

cc_df	A dataframe with columns 'source', 'target', 'ligand', 'receptor' and 'score'. See toy_data for example.
option	Either 'A', 'B' or 'C'. Option A will plot the number of interactions between pairs of cell types, option B will plot the top n_top_ints interactions and their scores. Option C will plot the top n_top_ints interactions, their scores and the mean expression of the ligands/receptors in the sending/receiver cell types.

n_top_ints	The number of top interactions to plot. Only required for options B and C.
exp_df	A dataframe containing the mean expression values for each ligand/receptor in each cell type. See toy_exp for an example. Only required for option C.
cell_cols	A named vector of colours for each cell type. Default uses paletteMartin(), a colourblind-friendly palette.
palette	Which colour palette to use to show the mean expression. Should be one of the RColorBrewer sequential palettes.
cex	Determines text size
show_legend	TRUE or FALSE - whether to add legend or not. Only required for options B and C.
scale	TRUE or FALSE - whether to scale each sector to same width. Only required for options B and C.
...	Additional parameters passed to chordDiagram function.

**Value**

Returns a chord diagram generated by the circlize R package

**Examples**

```
data(toy_data, toy_exp, package = 'CCPlotR')
cc_circos(toy_data)
cc_circos(toy_data, option = "B", n_top_ints = 10, cex = 0.5)
cc_circos(toy_data,
  option = "C", n_top_ints = 15, exp_df = toy_exp,
  cell_cols = c(`B` = "hotpink", `NK` = "orange", `CD8 T` = "cornflowerblue"),
  palette = "PuRd", cex = 0.5
)
```

---

cc\_dotplot

*Dotplot Function*


---

**Description**

This function plots a dotplot

**Usage**

```
cc_dotplot(cc_df, option = "A", n_top_ints = 30)
```

**Arguments**

cc_df	A dataframe with columns 'source', 'target', 'ligand', 'receptor' and 'score'. See toy_data for example.
option	Either 'A', 'B', 'CellPhoneDB' or 'Liana'. Option A will plot the number of interactions between pairs of cell types, option B will plot the top n_top_ints interactions and their scores. The 'CellPhoneDB' and 'Liana' options will generate a dotplot in the style of these popular tools.
n_top_ints	The number of top interactions to plot. Only required for option B.

**Value**

Returns a plot generated with the ggplot2 package

**Examples**

```
data(toy_data, package = 'CCPlotR')
cc_dotplot(toy_data)
cc_dotplot(toy_data, option = "B", n_top_ints = 10)
cc_dotplot(toy_data, option = "Liana", n_top_ints = 15)
```

---

cc_heatmap	<i>Heatmap Function</i>
------------	-------------------------

---

**Description**

This plots a heatmap

**Usage**

```
cc_heatmap(cc_df, option = "A", n_top_ints = 30)
```

**Arguments**

cc_df	A dataframe with columns 'source', 'target', 'ligand', 'receptor' and 'score'. See toy_data for example.
option	Either 'A', 'B', 'CellPhoneDB' or 'Liana'. Option A will plot the number of interactions between pairs of cell types, option B will plot the top n_top_ints interactions and their scores. The 'CellPhoneDB' and 'Liana' options will generate a heatmap in the style of these popular tools.
n_top_ints	The number of top interactions to plot. Only required for option B.

**Value**

Returns a plot generated with the ggplot2 package

**Examples**

```
data(toy_data, package = 'CCPlotR')
cc_heatmap(toy_data)
cc_heatmap(toy_data, option = "B", n_top_ints = 10)
cc_heatmap(toy_data, option = "CellPhoneDB")
```

---

cc\_network

*Network Plot Function*


---

**Description**

This function plots a network of representing the number of interactions between cell types

**Usage**

```
cc_network(
  cc_df,
  colours = paletteMartin(),
  option = "A",
  n_top_ints = 20,
  node_size = 2.75,
  label_size = 4,
  layout = "kk"
)
```

**Arguments**

cc_df	A dataframe with columns 'source', 'target', 'ligand', 'receptor' and 'score'. See toy_data for example.
colours	A vector of colours for each cell type. Default is paletteMartin(), a colourblind-friendly palette.
option	Either 'A' or 'B'. Option A will plot the number of interactions between pairs of cell types, option B will plot the top n_top_ints interactions and their scores.
n_top_ints	The number of top interactions to plot. Only required for option B.
node_size	Point size for nodes in option B.
label_size	Size for labels in option B.
layout	Algorithm for determining layout in option B. One of 'dh', 'drl', 'fr', 'gem', 'graphopt', 'kk', 'lgl', 'mds', 'nicely'. See iGraph layouts for more details.

**Value**

Returns a plot generated with the ggplot2 package

**Examples**

```
data(toy_data, package = 'CCPlotR')
cc_network(toy_data)
cc_network(toy_data, colours = c("orange", "cornflowerblue", "hotpink"), option = "B")
```

---

`cc_sigmoid`*Sigmoid Plot Function*

---

**Description**

This function plots interactions using the `geom_sigmoid` function from the `ggbump` R package

**Usage**

```
cc_sigmoid(cc_df, n_top_ints = 20, colours = paletteMartin())
```

**Arguments**

<code>cc_df</code>	A dataframe with columns 'source', 'target', 'ligand', 'receptor' and 'score'. See <code>toy_data</code> for example.
<code>n_top_ints</code>	The number of top interactions to plot.
<code>colours</code>	A named vector of colours for each cell type. Default is <code>paletteMartin()</code> , a colourblind-friendly palette.

**Value**

Returns a plot generated with the `ggplot2` package

**Examples**

```
data(toy_data, package = 'CCPlotR')
cc_sigmoid(toy_data)
cc_sigmoid(toy_data, colours = c(
  `B` = "hotpink", `CD8 T` = "orange",
  `NK` = "cornflowerblue"
), n_top_ints = 25)
```

---

`geom_sigmoid`*geom\_sigmoid This function is copied from the ggbump package*

---

**Description**

Creates a `ggplot` that makes a smooth rank over time. To change the `smooth` argument you need to put it outside of the `aes` of the `geom`. Uses the `x`, `xend`, `y` and `yend` aesthetics. Make sure each sigmoid curve is its own group.

## Usage

```
geom_sigmoid(  
  mapping = NULL,  
  data = NULL,  
  geom = "line",  
  position = "identity",  
  na.rm = FALSE,  
  show.legend = NA,  
  smooth = 8,  
  direction = "x",  
  inherit.aes = TRUE,  
  ...  
)
```

## Arguments

mapping	provide you own mapping. both x, xend, y and yend need to be numeric.
data	provide you own data
geom	xhange geom
position	change position
na.rm	remove missing values
show.legend	show legend in plot
smooth	how much smooth should the curve have? More means steeper curve.
direction	the character x or y depending of smoothing direction
inherit.aes	should the geom inherits aesthetics
...	other arguments to be passed to the geom

## Value

ggplot layer

## Examples

```
library(ggplot2)  
df <- data.frame(x = 1:6,  
  y = 5:10,  
  xend = 7,  
  yend = -3:2)  
  
ggplot(df, aes(x = x, xend = xend, y = y, yend = yend, color = factor(x))) +  
  geom_sigmoid()
```



---

paletteMartin	<i>Discrete palette generator</i>
---------------	-----------------------------------

---

**Description**

Generate a palette of up to 15 colours. The colours are from the paletteMartin palette in the color-Blindness R package.

**Usage**

```
paletteMartin(n = 15)
```

**Arguments**

n                      Number of colours to return. Max = 15.

**Value**

Returns a vector of colours of length n.

**Examples**

```
scales::show_col(paletteMartin(n = 9))
```

---

StatSigmoid	<i>This function is copied from the ggbump package</i>
-------------	--

---

**Description**

This function is copied from the ggbump package

---

toy_data	<i>Toy data for CCPlotR</i>
----------	-----------------------------

---

**Description**

A toy dataset of ligand-receptor interactions to demonstrate cell-cell interaction plots.

**Usage**

```
toy_data
```

**Format**

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 735 rows and 5 columns.

**Value**

toy\_data:

A data frame with 735 rows and 5 columns:

**source** Cell type expressing the ligand

**target** Cell type expressing the receptor

**ligand** Ligand

**receptor** Receptor

**score** A score for each interaction e.g.  $-\log_{10}(\text{aggregate\_rank})$  returned by Liana

**Source**

This is a modified version of the toy dataset that comes with the Liana R package.

---

toy\_exp

*Toy expression data for CCPlotR*

---

**Description**

A dataframe showing the mean expression values for each ligand and receptor in each cell type.

**Usage**

toy\_exp

**Format**

An object of class `grouped_df` (inherits from `tbl_df`, `tbl`, `data.frame`) with 477 rows and 3 columns.

**Value**

toy\_exp:

A data frame with 477 rows and 3 columns:

**cell\_type** Cell type

**gene** Ligand/receptor gene

**mean\_exp** Mean (normalised) expression of ligand/receptor gene in cell type

# Index

## \* datasets

StatSigmoid, 9

toy\_data, 9

toy\_exp, 10

cc\_arrow, 2

cc\_circos, 3

cc\_dotplot, 4

cc\_heatmap, 5

cc\_network, 6

cc\_sigmoid, 7

geom\_sigmoid, 7

paletteMartin, 9

StatSigmoid, 9

toy\_data, 9

toy\_exp, 10