Interactive ggplot with ggiraph: : **CHEAT SHEET**

Basics

ggiraph package is an extension for **ggplot2**, which enables users to produce interactive ggplot graphs.

Instead of basic function **geom_*()** in ggplot2, ggiraph use **geom_*_interactive()** as basic interactive function.
Also, users can make
interactive scale with **scale_*_*_interactive()**
Interactive guide with **guide_*_interactive()**
Interactive theme elements with **element_*_interactive()**

To add interactivity to the graph, be sure to use at least one of the three aesthetic :
tooltip, **data_id** and **onclick**

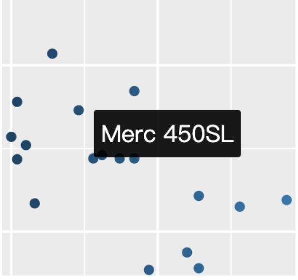
Note: to make the final output interactive, we must include usage of **girafe()** function, or the output will be a single png!

Aesthetic

cars <- mtcars
start <- ggplot(cars, aes(x=mpg, y=wt, color= mpg))

- tooltip:

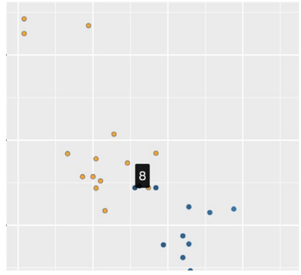
show a tooltip when mouse hang over.



a <- start+
geom_point_interactive(aes(tooltip = row.names(cars)))
# set the tooltip to be the factor you want to show
girafe(print(a)) *#use code print() to get interactive figure output*

- data_id:

animate all elements associated with the data_id upon mouse over.



b <- start+
geom_point_interactive(aes(tooltip = cyl, data_id= cyl))
# set points into group based on factor x
girafe(ggobj = b) *#use factor ggobj=* to get interactive output (2nd way)*

- onclick:

associate mouse click action with a JavaScript function execution. Object must be a **string column containing valid JavaScript instruction.**

cars$on_click<- sprintf("window.open(\"%s%s\")", "https://www.google.com/search?q=", as.character(row.names(cars)))
# create a new column, use sprintf function to instruct to open a hyperlink
c <- start+ geom_point_interactive(aes(tooltip = row.names(cars), onclick=on_click))
#link the JavaScript function to mouse click action
girafe(ggobj = c) # get output

In the sample code, a hyperlink that searches the row name (brand of car) in Google is opened when clinking on the point.

Girafe

create a interactive girafe object as output.

code: the plotting code to excute graph
ggobj: indicate the ggplot object to be print
Only one argument needed

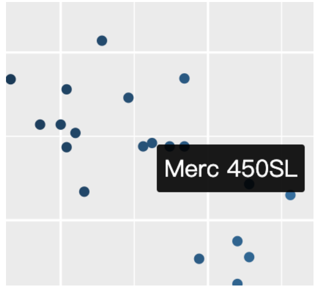
girafe(code = NULL, ggobj = a, pointsize = 12, width_svg = 6, height_svg = 5)

pointsize: point size of plotted text in pixels
width_svg & height_svg: width and height of the graphics region in inches.

Note: when **ggobj** is indicated, argument **code** will be ignored.

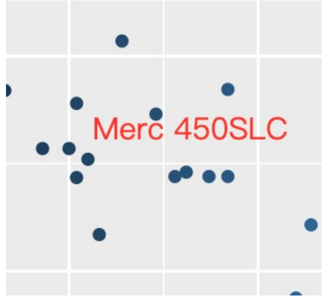
Customize Options

- tooltip options:



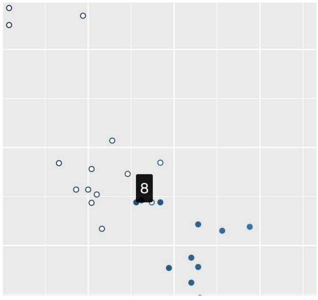
girafe(ggobj = a, options = list(opts_tooltip(offx = 20, offy = 20)))
*# use **opts_tooltip** to customize tooltip*

*# **offx/offy** arguments set tooltip position to specific horizontal/vertical pixels*

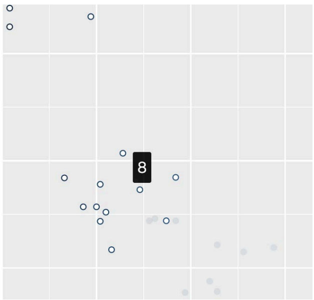


tooltip_css <- "background-color:transparent;color:red;"
# property names and values are separated by colons; each pair end with semicolon
girafe(ggobj = a, options = list(opts_tooltip(css = tooltip_css)))
*# use **css** to customize tooltip style*

- data_id options: (hover effect)



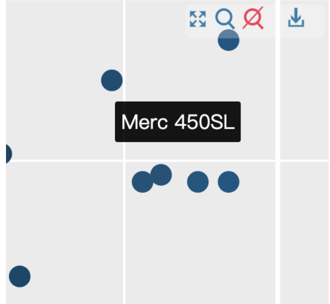
girafe(ggobj = b, options = list(opts_hover(css = "fill:white;")))
*# use **opts_hover** and **css** to customize the color of hover effect*



girafe(ggobj = b, options = list(opts_hover_inv(css = "opacity:0.1;"), opts_hover(css = "fill:white;")))
*# use **opts_hover_inv** to alter aspect of non hovered elements*

Zoom

control the zoom icon in the toolbar(by default at top right of ggiraph). When the zoom icon is activated, viewers can zoom in with mouse.



girafe(ggobj = a, options = list(opts_zoom(max = 5)))
# when set maximum zoom factor to value>1, toolbar with zoom icon will appear when mouse is over the graphic

RStudio® is a trademark of RStudio, Inc. • CC BY SA April Pan • yqpanap@gmail.com • Learn more at http://davidgohel.github.io/ggiraph/index.html • package version 0.7.10 • Updated: 2022-04