\*\*create data set and a starter script\*\*

Plotly

Learning objectives:

* Hover information (tooltips) 🡪 information layering beyond what is directly plotted🡪 plotly takes info from layers in each object
* Zoom in and out/ pan
* 3D visualisations
* Repurposing ggplot code ggplotly(), 🡪 simple and little code
* Filtering
* Heat maps

What

* Interface to the plotly javascript graphing library – wraps javascript for multiple codes including R, Python, matlab, javascript
* Browser/ html based charts and visualisations
* Amazing graphics and visualisations

Why

* Popular news media (web based) has become very progressive in how they are displaying data (context of covid).
* Interactive and can deploy to the web as web-apps
* Powerful graphs with few code
* Can be useful for dense data or high dimensionality data 🡪 zooming and filtering
* Allows the improvement of existing code (ggplotly)
* Wide variety of graphs
* Active community and development - Open source
* Static vs. interactive

- Static useful for reports useful for displaying what you the creator has highlighted

- User can update an interactive graphic e.g. drill down to specific data points using hover info or focusing on subsets of data by selecting or deselecting groups

- Simple interaction improve ability of data exploration

-

How

* Can use Ggplotly to convert ggplot plots to be interactive``` ggplotly(static)```

Note

* Interactivity doesn’t equal good graphic 🡪 refer to best practices of data visualisation, think about syntax and design principles
* Not all ggplot can be converted to plotly objects

Limitations

Step 1:

Data set/ starter code \*\*\*iris data set? What are the standard R example data sets. (diamonds data set is dense)

* Link to repo to copy it locally and access data set
* Read in data set
* Explore the data set.
* Define how to do this- glimpse()
* Outline the parameters of the data set
* Find out what the starter code runs (ggplotly)
* Load in/ install plotly package

ggplotly()

\*\* have existing ggplot code to apply ggplotly too.

* Assign ggplot code to a variable
* Run ggplotly(variable)
* Explore new graphic 🡪 highlight hover info
* Customise hover info using text variable in variable of interest in ggplot code e.g. geom\_point(aes(text = paste0(variable e.g. i.d., “ \nAge : ”, age))
* Ignore the warning as it is for ggplot but not ggplotly
* Add tooltip = argument in ggplotly() with “text”
* EXTRA: can use pipes from dplyr

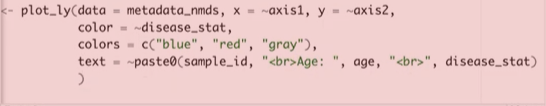
Save interactive plots Htmlwidgets

* htmlwidgets package
* use the saveWidget( plot\_name, “assigned\_name.html”, selfcontained = false, libdir = “lib/”)

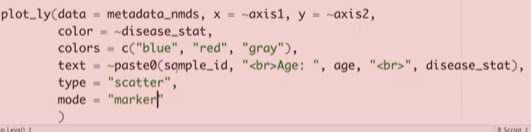
Plot customisation

* Title, using %>% layout(title = “title”) -🡪 can add in ggplot or in plotly

Plotting directly from plotly



* Type + scatter to avoid warning messages



* ~ refers to explained by notation