# Interactive Input Functions

## Functionality

- As well as `input_slider()`, ggvis provides `input_checkbox()`, `input_checkboxgroup()` and more (*See next slide*).
- See the examples in the documentation for how you might use each one.
- You can also use keyboard controls with `left_right()` and `up_down()`.

# Interactive Input Functions

| **Function** | **Description** |
|---|---|
| `input_slider` | Slider to select values or ranges of values |
| `input_checkbox` | A single check box |
| `input_checkboxgroup` | A group of check boxes |
| `input_numeric` | A spin box |
| `input_radiobuttons` | Selection of a single value from a set of options |
| `input_select` | A drop down text selection |
| `input_text` | Text input |

## Tooltips

- `add_tooltip` allows us to include other behaviour when we hover or click on a point
- We can provide a single function that takes as input a list of the data stored in a given point

# Interactivity Exercise

**Exercise**

- Update the previous plot of *mpg* against *wt* so points change colour when they hover over
- Add a tooltip that shows the value of *mpg* when the point is hovered over
- Add a slider for the span of the smooth line so that values can be set between 0 and 1

## Controlling with Sliders

The following example allows you to control the size and opacity of points with two sliders:

```
mtcars  %>%
   ggvis(~wt, ~mpg,
      size := input_slider(10, 100),
      opacity := input_slider(0, 1)) %>%
   layer_points()
```

You can also connect interactive components to other plot parameters like the width and centers of histogram bins:

```
mtcars %>%
  ggvis(~wt) %>%
  layer_histograms(width =  input_slider(0, 2,
                           step = 0.10,
                           label = "width"),
                   center = input_slider(0, 2,
                           step = 0.05,
                           label = "center"))
```

```
keys_s <- left_right(10,1000,step = 50)

mtcars  %>%
  ggvis(~wt, ~mpg, size := keys_s,
        opacity := 0.5) %>%
    layer_points()
```

# Interactvity : Tooltips

You can also add on more complex types of interaction like tooltips:

```
mtcars  %>%
  ggvis(~wt, ~mpg) %>%
     layer_points() %>%
     add_tooltip(function(df) df$wt)
```

**Vignette** : Youll learn more about complex interaction in the **Interactivity** vignette.

## Shiny

- Behind the scenes, interactive plots are built with shiny, and you can currently only have one running at a time in a given R session.
- To finish with a plot, press the stop button in Rstudio, or close the browser window and then press Escape or Ctrl + C in R.