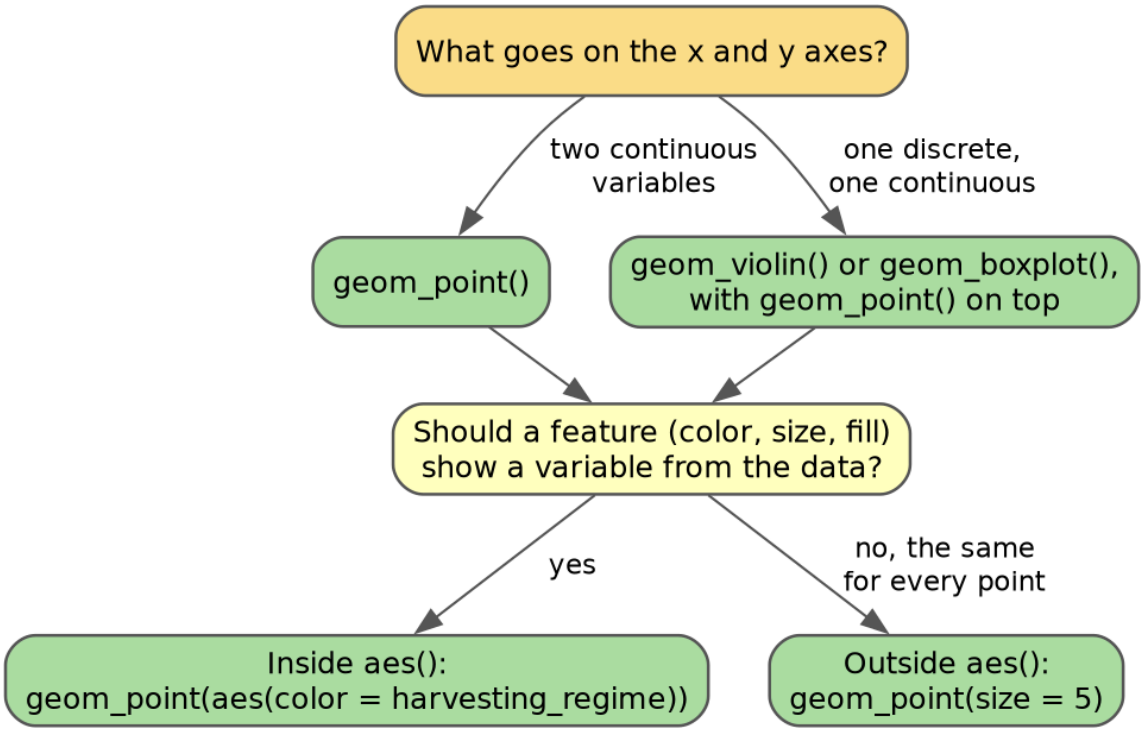
data visualization I: summary

Integrated Bioanalytical Chemistry · thebustalab.github.io/integrated_bioanalytics



Objects store data (like: `my_object <- 7`), and functions act on it: `filter()` keeps only the rows that meet a logical condition, such as `filter(my_data, chemical_species == "FAs")`. A ggplot is then built in three steps: name the data with `ggplot(data = ...)`, map variables onto the axes with `aes()`, and add one or more geoms with +. Geoms are geometric objects with features (size, color, shape, etc.). A feature mapped to a **variable** in the data goes **inside** `aes()` (`geom_point(aes(color = harvesting_regime))`), and a feature set to a **constant** goes **outside** `aes()` (`geom_point(size = 5)`). Swapping mappings around is cheap, so it is worth trying a few to see which plot best communicates the aspect of your data that you are interested in.

Function or operator	What it does	What we vary
`<-`	stores a value or a result in an object	the object name on the left
`filter()`	keeps only the rows that meet a condition	the condition: ==, !=, <, <=, >, >=, `%in% c(...)`
`ggplot()`	starts a plot and names the data	`data =`
`aes()`	maps variables onto the axes and onto features of the shapes	x, y, `color`, `size`, `fill`
+	adds a geom to the plot, and several geoms can be stacked	end the line with + to continue on the next
`geom_point()`	draws one point per observation	`aes(color = ...)`, `aes(size = ...)`, or constants such as `size = 5`, `color = "black"`
`geom_violin()`	draws the spread of values in each group	`aes(fill = ...)`
`geom_boxplot()`	draws a box plot for each group	nothing extra in this chapter