

EDAV Fall 2019 PSet 5, part B

Use SVG/JavaScript/D3 to create a visual that will help a user learn about boxplots. It should include a textual description of what it is showing and have the following functionality: When the user adds points to the canvas, summary statistics are shown on the screen and a boxplot is drawn. As points are added and removed, the summary statistics and boxplot are updated. The boxplot transitions smoothly from one state to the next. (See video links below.)

Use `d3.quantile()` to determine Q1 and Q3. Outliers are equal to or below Q1 minus 1.5 times the interquartile range, or equal to or above Q3 plus 1.5 times the interquartile range. (This method should produce the same boxplots as **ggplot2**, not base R, which uses Tukey's five number summary instead of quartiles.)

Use the following method: each time a point is added, bind the data value (not pixel value) to the circle. To get the data values for all existing circles, use `d3.select("svg").selectAll("circle").data();`. Note that when you remove a circle, the data value goes with it, so you don't have to keep track of the data values separately from the circles. (To put it more strongly: don't do it. Resist the temptation to keep an independent array of data values. For this particular visual, such an approach is less elegant, more prone to error, and more difficult in general.) Note as well that you don't need to use the general update pattern. In fact, you do not need to use `enter()` or `append()` at all since we don't have a data source external to the SVG elements.

Video links:

[YouTube](#)

[YouTube slo-mo](#)

[CourseWorks](#)

[CourseWorks slo-mo](#)

Suggested workplan

Follow the minimal viable product model: at each stage of development you should have something that works. At each of those stages save your work. Note that you only need to turn in the last stage, as it will contain the functionality of all the previous stages.

Your plan might look something like this:

1. `PSet5B-1points.html`: Create a rectangle on which points appear if you click.
2. `PSet5B-2buttons.html`: Add buttons to allow for points to be removed.
3. `PSet5B-3scales.html`: Add scales.
4. `PSet5B-4axis.html`: Add a y-axis.
5. `PSet5B-5stats.html`: Calculate summary statistics as indicated and display them on the right side of the screen.
6. `PSet5B-6boxplot.html`: Use the summary statistics to create a boxplot, which updates each time a point is added or removed. *Do not include transitions.*
7. `PSet5B-7transitions.html`: Add transitions.

Grading

Functionality	Points
points can be added or removed	5 points
y-axis	2 points
summary stats are correct, rounded and ordered	6 points
informative textual description	2 points
boxplot drawn properly	5 points
transitions	5 points

You do not need to follow the model in the video for: specific color choices (be sure though that outliers are a different color), boxplot width, html layout, cosmetic features in general

Initial SVG elements may be drawn with SVG or D3.

You may borrow code that you find online, just cite your source in code comments.

Make sure your statistics and boxplot are correct. There is a lot of variability in how boxplots are drawn.