Assignment 5 Interactive Data Visualization

STAT141B Spring 2025

Professor Duncan Temple Lang

Due: June 12, 10pm - ** NO EXTENSIONS **

Submit via Canvas

Your task is to create an interactive HTML display with interactive plots.

Clean the Data

Before you start creating plots, explore the data and identify outliers and possible errors in the data. Include in your report what observations you removed or corrected and why.

The Display

Your page will have some text describing to the viewer

- what they are seeing,
- how to interact with the elements, and
- also a description of one or more insights from you exploring the data, e.g., guiding them towards some things to explore and how to find the insights.

We'll use the data set from the properties for sale I collected from 8 counties near Davis on June 2nd. This is an RDS file that you can read into R via `readRDS()` after downloading the file to your computer.

Below the introductory text on the HTML page you create, include

- a map/plot showing each home via its latitude and longitude values,
- a scatter-plot of square-footage and price,
- a table that shows the details of each given rental unit, i.e., column of field descriptions and a column of the corresponding field values.
  - The contents of this table will be displayed when the user clicks on a rental unit in the "map" plot.

Color code the points on the plots to indicate the value of a 3rd variable, e.g., the type of property, number of bedrooms, number of bathrooms.

Customize the tooltips on the plots to show more information including a link to the post for the rental unit that the user can click to visit that page in a separate tab/window.

When one clicks on a point in the map,

- the corresponding point in the scatter-plot should become highlighted, and
- the table should show the details for the corresponding rental unit.

When one clicks on a different point, the previous point in the scatter-plot should become unhighlighted.

You can use R to create the plots and/or the HTML.

You can add JavaScript when creating the document from R via `saveWidget` or in a separate programmatic step.

You can also create the content directly in HTML and JavaScript.

You can generate HTML in R at page-creation-time and also in JavaScript at run-time.

As you write JavaScript functions, ask lots of questions. We want to do the basics and not focus on the language.

Explore how the user can focus on a subset counties.

Keep It Simple (***KISS principle***)

This is a simple use of interactivity to learn the fundamentals. It is not intended to be very sophisticated or rich in user interface/experience details or functionality.

Focus on the mechanisms to get the interactivity and focus less on making the plots and page pretty. (Do make certain to put labels on each axis, legend, etc.)

Don't forget to make use of the Developer Tools in the Web browser to observe the interactive page and its elements and the output via the Console.

Strategy

I suggest working first on the longitude/latitude "map" and making each point clickable so that it displays the details of that post as an HTML table.

Then add the second plot. Then connect it to the events for the first plot.

Make both plots interactive so that the user can click on a point in either plot and the points and the table are updated.

What to Submit

- The collection of self-contained display documents that allow me to view the interactive display.
  - Zip all of the related files needed to view the page
  - This cannot be a link to a Web page
  - This cannot be a single file that references other files.
- PDF containing your report describing how you created the interactive display.
- The R code you used to create the display.
- Any additional JavaScript or CSS code you used.

Possibly useful packages

- plotly
- crosstalk
- ggiraph
- patchwork
- htmlwidgets
- htmltools
- leaflet

Some possibly useful functions

- `ggplotly()`, `plot_ly()`
- `highlight()`, `highlight_key()`
- `HTML()`, `tags$...`, e.g., `tags$script()`,, `tags$a()`, `tags$table()`
- `prependContent()`, `appendContent()`
- `onRender()`
- `style()`
- `subplot()`
- `layout()`
- `saveWidget()`

Links

- JavaScript tutorial
- Plotly book
  - Chapter 16 - linking plots
  - Chapter 20 - Event Handlers
  - Chapter 21 - Custom Data
  - Chapter 10 - Saving widgets/pages
  - Chapter 25 - Tooltips

Useful JavaScript Information/Functions/...

- `document` - the DOM object for the page.
- `getElementById()` - get the element with the specified id attribute.
- `innerHTML` attribute - specify the HTML content for a given DOM object which will be displayed when the attribute is set.

Data

The data were created by combining different sources for each property. Therefore, they may not agree.

Ideally, the price and price.1 variables would be the same. However, there are many properties where `price` is not `NA` but `price.1` is `NA`. Also, there are a few properties where both `price` and `price.1` are **not** `NA`, but are not the same.