

Q4 [20 points] Interactive Visualization

Goal	Create line charts in D3 that use interactive elements to display additional data. Then implement a bar chart that appears when you mouse over a point on the line chart.
Technology	D3 Version 5 (included in the lib folder) Chrome v131.0.0 (or higher): the browser for grading your code Python http server (for local testing)
Allowed Libraries	D3 library is provided to you in the lib folder. You must NOT use any D3 libraries (d3*.js) other than the ones provided. On Gradescope, these libraries are provided for you in the auto-grading environment.
Deliverables	[Gradescope] Q4. (html/js/css) : The HTML, JavaScript, CSS to render the visualization in Q4. Do not include the D3 libraries or average-rating.csv dataset.

Use the dataset `average-rating.csv` provided in the Q4 folder to create an interactive [frequency polygon](#) line chart. This dataset contains a list of games, their ratings and supporting information like the numbers of users who rated a game and the year a game was published. In the data sample below, each row under the header represents a game name, year of publication, average rating, and the number of users who rated the game. Helpful resource to work with nested data in D3: <https://gist.github.com/phoebebright/3176159>

```
name,year,average_rating,users Rated
Codenames,2015,7.71148,51209
King of Tokyo,2011,7.23048,48611
```

1. **[3 points] Create a line chart.** Summarize the data by displaying the count of board games by rating for each year. Round each rating down to the nearest integer, using [Math.floor\(\)](#). For example, a rating of 7.71148 becomes 7. For each year, sum the count of board games by rating. Display one plot line for each of the 5 years (2015-2019) in the dataset. **Note:** the dataset comprises year data from 2011 to 2020; this question asks to plot lines for the years 2015-2019. If some of the datapoints in the chart do not have ratings, generate dummy values (0s) to be displayed on the chart for the required years.

All axes must start at 0, and their upper limits must be automatically adjusted based on the data. **Do not hard-code the upper limits.** **Note:** if you are losing points on Gradescope for axis or scale, ensure that you are using the proper [margin convention](#) without any additional paddings or translations.

- The vertical axis represents the count of board games for a given rating. Use a [linear scale](#).
- The horizontal axis represents the ratings. Use a linear scale.

2. **[3 points] Line styling, legend, title and username.**

- For each line, use a different color of your choosing. Display a filled circle for each rating-count data point.
- Display a legend on the right-hand portion of the chart to show how line colors map to years.
- Display the title "Board games by Rating 2015-2019" at the top of the chart.
- Add your GT username (usually includes a mix of lowercase letters and numbers, e.g., gburdell3) beneath the title (see example Figure 4.1).

Figure 4.1 shows an example line chart design. Yours may look different but can earn full credit if it meets all stated requirements.

Note: The data provided in `average-rating.csv` requires some processing for aggregation. **All aggregation must only be performed in JavaScript; you must NOT modify `average-rating.csv`.** That is, your code should first read the data from `.csv` file as is, then you may process the loaded data using JavaScript. If you are getting a `MoveTargetOutOfBoundsException`, (a) check that your [margin convention](#) is

correct, and (b) make sure to check the Dropbox linked screenshot of your graph to get a good idea of how the plot could be Improved compared to the sample graph provided.

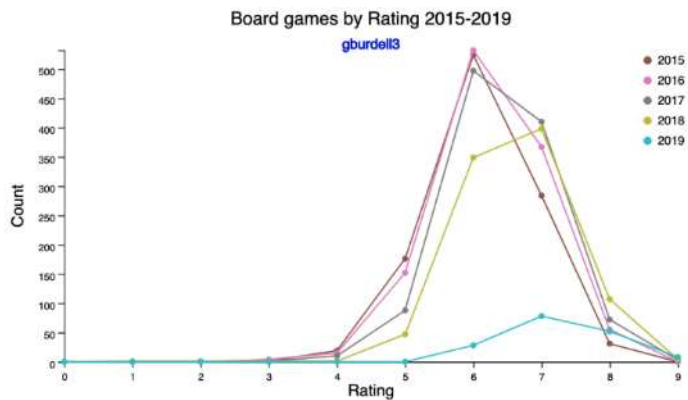


Figure 4.1: Line chart representing count of board games by rating for each year. Your chart may appear different but can earn full credit if it meets all stated requirements.

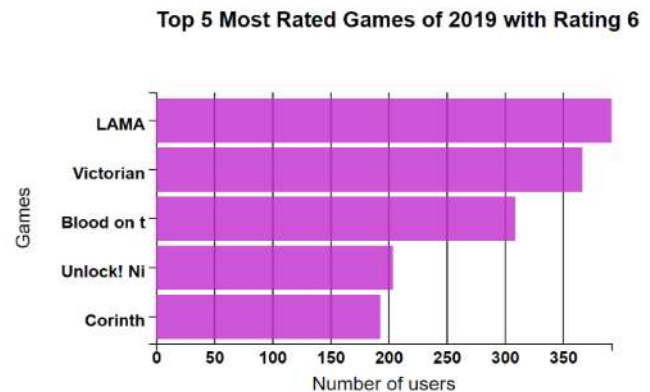


Figure 4.2: Bar chart representing the number of users who rated the top 5 board games with the rating 6 in year 2019. Your chart may appear different but can earn full credit if it meets all stated requirements.

Interactivity and sub-chart. In the next few sub-questions, you will create event handlers to detect mouseover and mouseout events over each circle that you added in Q4.2.

- [8 points] Create a horizontal bar chart**, so that when hovering over a circle, that bar chart will be shown *below* the line chart. The bar chart displays the top 5 board games that received the highest numbers of user ratings (`users Rated`), for the hovered year and rating. For example, hovering over the rating-6 circle for 2019 will display the bar chart for the number of users who rated the top 5 board games. If a certain year/rating combination has fewer than 5 entries, it should display as many as there are. Figure 4.2 shows an example design. Show one bar per game. The bar length represents the number of users who rated the game.

Note: No bar chart should be displayed when the count of games is 0 for hovered year and rating.

Axes: All axes should be automatically adjusted based on the data. Do not hard-code any values.

- The vertical axis represents the board games. Sort the game names in descending order, such that the game with the highest `users Rated` is at the top, and the game with the smallest `users Rated` is at the bottom. Some boardgame names are quite long. For each game name, display its first 10 characters (if a name has fewer than 10 characters, display them all). A space counts as a character. The horizontal axis represents the number of users who rated the game (for the hovered year and rating). Use a linear scale.
 - Set horizontal axis label to 'Number of users' and vertical axis label to 'Games'.
- [2 points] Bar styling, grid lines and title**
 - Bars:** All bars should have the same color regardless of year or rating. All bars for the specific year should have a uniform bar thickness.
 - Grid lines** should be displayed.
 - Title:** Display a title with the format "Top 5 Most Rated Games of <Year> with Rating <Rating>" at the top of the chart where <Year> and <Rating> are what the user hovers over in the line chart. For example, hovering over rating 6 in 2015, the title would read: "Top 5 Most Rated Games of 2015 with Rating 6"
 - [2 points] Mouseover Event Handling**
 - The bar chart and its title should only be displayed during mouseover events for a circle in the line chart.

- The circle in the line chart should change to a larger size during mouseover to emphasize that it is the selected point.
- When count of games is 0 for hovered year and rating, no bar chart should be displayed. The hovered-over circle on the line graph should still change to a larger size to show it is selected.

Hint: `.attr()` is generally used for describing the size, shape, location, etc. of an element, whereas `.style()` is used for other design aspects like color, opacity, etc.

6. [2 points] Mouseout Event Handling

The bar chart and its title should be hidden from view on mouseout and the circle previously mouseover-ed should return to its original size in the line chart.

The graph should exhibit interactivity similar to Figure 4.6 where the mouse is over the larger circle.

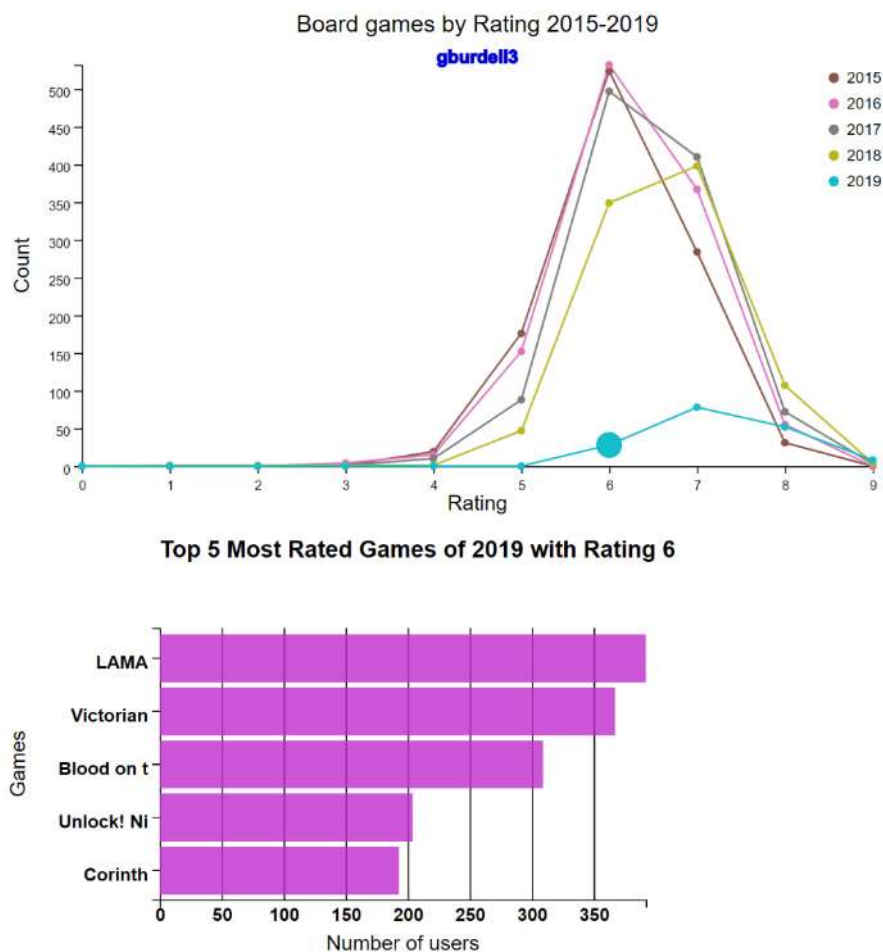


Figure 4.6: Line chart and bar chart demonstrating interactivity. Your chart may appear different, but you can earn full credit if it meets all stated requirements.

Note: Your D3 visualization **MUST** produce the following [DOM structure](#).

```
<svg id="line_chart"> containing line chart
|
+-- <g id="container">
|   |
|   +-- <g id="lines"> element containing all line elements
|   |   |
|   |   +-- <path> elements for plotted lines
|   |
|   +-- <g id="x-axis-lines"> element for x-axis
|   |
|   +-- <g id="y-axis-lines"> element for y-axis
|   |
|   +-- <g id="circles"> element for all circular elements
|   |   |
|   |   +-- <circle> elements
|   |
|   +-- <text id="line_chart_title"> element for line chart title
|   |
|   +-- <text id="credit"> element for GT username
|   |
|   +-- <g id="legend"> element for legend
|   |   |
|   |   +-- (<circle> elements for legend)
|   |   |
|   |   +-- (<text> elements for legend)
|   |
|   +-- <text> element for x axis label
|   |
|   +-- <text> element for y axis label
|
<div id="bar_chart_title" > containing bar chart title

<svg id="bar_chart" > containing bar chart
|
+-- <g id="container_2">
|   |
|   +-- <g id="bars"> element for bars
|   |   |
|   |   +-- <rect> elements for bars
|   |
|   +-- <g id="x-axis-bars"> element for x-axis
|   |
|   +-- <g id="y-axis-bars"> element for y-axis
|   |
|   +-- <text id="bar_x_axis_label"> element for x axis label
|   |
|   +-- <text id="bar_y_axis_label"> element for y axis label
```