CS/INFO 3300; INFO 5100
Homework 3
Due 11:59pm Tuesday February 23

1. In problem 2 on HW2 you recreated Fig. 2 from the Wickham reading using SVG elements. Now create the same plot again, but this time using d3 functions. First create x and y scale functions that map from data coordinates to SVG coordinates (10 pts). Add circles and rectangles, with positions given by the x and y scales. You don't need to use `data()` or `enter()` functions: it's fine if you do a separate command for each shape (10 pts). Add d3 axes, again using the x and y scale functions (10 pts). Now add an event listener that changes the color of a circles or rectangles to blue when it is clicked, using d3 selections (10 pts).

2. In this problem we're going to plot some data about English word frequencies from Google Books. The file http://mimno.infosci.cornell.edu/info3300/zipf.json contains a Javascript snippet that defines an array of objects. Each object represents a word, sorted by its frequency. The most frequent word, "of", occurs 15 billion times. The 512th most frequent word, "middle", occurs 45 million times. Copy this snippet into a script element in your homework document. Add appropriate d3 axes for each figure.

A. Create a 200x200 pixel SVG element using d3 functions. Create two linear scale functions: an x scale for the "rank" and a y scale for the "count". Choose the "range" attributes to be appropriate for "rank" and "count". Use d3 to add text elements to the plot for each word in the data set. Use a loop or a "forEach" statement; you may not use a separate command for each word. Is this visualization useful? Why or why not? (10 pts)

AXES LABELS

B. In this section we'll transform the data as you create the text elements. Create a second 200x200 SVG element. Add the same points, but this time calculate the log of word's rank (use `Math.log()`), convert that value to a pixel value with a linear scale, and set "x" to that scaled log value. Similarly set "y" to the scaled log of the count. You will need to create new x and y scale functions using appropriate values for the "domain". How does this version differ from the previous version? (5 pts)

C. Now rather than transforming the data, let's change the scale functions. Create a third 200x200 SVG element, and create two **log** scale functions using the same "domain" values as in part A. See the d3js.org API documentation as necessary. Use d3 to add text elements to this new plot, again using the original values for "rank" and "count". (5 pts)

3. Line plots. You are in charge of an artillery company. Your gun has a maximum initial projectile velocity of 150 m/s and an initial angle of 10 to 90 degrees. There is a tank 3m tall and 6m long that is 190m in front of you (measured from the nearest part). Your gun's muzzle is at elevation 1m. Create an SVG element and x and y scales with appropriate domains and ranges for this scenario. Add a `<rect>` for the tank, with height and width to

scale. Simulate a projectile trajectory with a discrete approximation, using steps every 0.2 seconds: keep four variables, `x`, `y`, `dx`, and `dy`, which represent the current position and the velocity (change in the current position after each fifth of a second). At each step calculate the new position of the projectile and the new velocity in the `x` and `y` dimensions. Assume that the `x` velocity is unchanging, and that the only force acting on the `y` velocity is gravity, which subtracts 9.81 / 5 every 0.2 seconds. Use `d3.svg.line()` to create a 25% opacity 5-pixel-wide red `<path>` element tracing this trajectory. Set your angle and initial velocity to hit the tank. (Historical note: this is what ENIAC was designed to compute; it also had to account for air resistance, which is the hard part.) (20 pts)

4. Map yourself! Find the longitude and latitude coordinates of three places you have lived, or want to live. Use d3 to create a map of the US or the world, as appropriate (10 pts). Use the JSON geographic files included with the class notes on GitHub or find your own. Select a projection for the map. Consult the d3 documentation for options. If you choose, you may want to use one of the projections from the d3-geo-projection package, which will require an additional javascript library file, available at https://github.com/d3/d3-geo-projection/. Place colored circles and text labels on the map in the locations you selected. (10 pts)