

Second Project

Assignment 2.4. A Critical Data Visualization

Posted March 18

Due: Before Next Class

Score: 20 pts

Overview

Our second project asks you to bring to completion and thoroughly document your data visualization project. The result of your efforts should be an original software that visualizes data: a “Critical Data Visualization” that reveals unseen aspects of data of your choice and allows us to reflect upon the data itself.

1. Develop your project

From your conceptual sketches, drawings and pseudocode, develop an original piece of software that visualizes data interactively. As before, you will likely face many small difficulties as you develop something from concept to resolution—things you don’t know how to code, libraries you don’t know how to use, or problems in the code that seem intractable. Try to identify the problems and move forward. See Week 4 advise on debugging.

2. Document it

Document your visualization in the class blog. Your post should consider the categories of data visualization we discussed in class, as well as our discussions about Drucker’s paper, and address the following points:

- Discuss whether your visualization is an *information display*, or a *knowledge generator*, and explain why.
- Discuss the intellectual purpose of your visualization, and how the visual language you chose helps you achieve that purpose. To address this, refer to the Drucker paper and our class discussion: consider whether your visualization belongs in one or several of the following categories: *knowledge trees*; *dynamic systems*; *time-keeping*; *administration/record-keeping*; and *space-making*.
- Discuss the difficulties you encountered during the data-collection and analysis stages, as well as during the development of the software itself. Include sketches and previous versions.
- Starting from Drucker’s ideas about data vs capta, address the following questions:
 - How was the data “constructed”?
 - What interpretations or assumptions are embedded in the data itself, as well as in the visual languages you chose to represent it?
 - Is your visualization making these interpretations and assumptions explicit, or visible?
- Include a video in the blog (around 1 min) of your visualization (make sure you do not show your Windows desktop).
- Upload your working code to your github account. Use comments to make it readable and easy to maintain.

Grading criteria

- Documentation: a) github post is complete, and b) the blog post is complete, well written, well illustrated, and includes a video of your visualization.
- Criticality: is your visualization revealing something new about the data, or generating new knowledge? Are you aware of the embedded assumptions in both your data and visual languages?
- Technical resolution: your project should show an understanding of fundamental issues of computer programming such as iteration, loops, parsing data, and functions.
- Quality of interaction: your project should be engaging.
- References: Precedents and examples including portions of the code you are using from other projects, are properly acknowledged in comments in the code.