UNIVERSITY OF REGINA
Department of Computer Science

CS 837 – Information Visualization
Fall 2015

Assignment #2: Experiments with Visual Variables

Due: October 15, 2015 by 11:55 PM

In this assignment, you will use a data set that contains nutritional information about 77 different breakfast cereals. However, unlike the previous assignment in which the choice of how to represent the data was rather straightforward, this dataset contains 15 different attributes. Your goal in this assignment is to write a program in which you can experiment with a number of different methods for visually encoding the data using some of the visual variables that are at our disposal.

The Data

The breakfast cereal dataset is provided by the authors of the textbook, and can be accessed online at:

http://www.idvbook.com/teaching-aid/data-sets/the-breakfast-cereal-data-set/

As with the previous assignment, your software should read the data file once, and then store the data in some internal data structure within your code. This data structure should be flexible enough so that it can drive the visual representation of the data in a number of different user-controllable ways (see below).

The Visualization

The core visualization method you should use for this data will be a scatterplot, created using Processing, D3, or some other visualization toolkit. Feel free to reuse some of your code from Assignment 1, or to start from scratch. Unlike Assignment 1, the goal here is not to create a single effective visualization of the data, but to build software that can allow a user (or you) to experiment with the different methods by which the data can be encoded using some of the visual variables discussed in class. The specific visual variables your software should support are:

- Position (x dimension)
- Position (y dimension)
- Mark
- Size
- Brightness
- Colour

Your software should include an interface that allows the user to specify which of the attributes to encode with each of these visual variables. If a particular visual variable is not matched to an attribute of the data, then a default value should be used (e.g., if you select to not match an attribute to colour, then the colour will be set to black). Once a particular selection is made in this interface, the visualization should be generated, which you can use to consider whether the particular mapping is appropriate.

Your software should also implement a simple focus operation to allow the user to inspect a particular data point (which you may find useful for debugging and validation purposes).

The Discussion

Your task in the discussion is to experiment with your software and the data set to find out which are good and which are bad visual variables for encoding the different data types within the data set. Create one screenshot that provides a good visual representation of the data, and one screenshot that provides a bad visual representation of the data. These screenshots should illustrate at least four different attributes from the data; however, they do not need to be the same attributes for each screenshot. For each of these screenshots, provide a brief explanation of why a user will or will not be able to easily perceive and interpret the visual representation.

The Deliverables

The following deliverables are required to be submitted electronically via URCourses (zipped together into a single file):

- a single PDF that contains the following elements
  - cover page that includes your name, student number, and email address
  - a screenshot of your software
  - instructions for compiling and running your program
  - two screenshots and explanations of why the visual representations generated are good or bad
- the source code of your program, organized in some reasonable directory structure

Academic Integrity

While you may talk about this assignment with your classmates in order to help each other understand what is needed, all the work submitted must be your own individual software development and writing. You are welcome to use whatever third-party visualization frameworks, tools, and extensions you can find. However, these details must be clearly documented both within the written submission and your source code.