

CS171 Final Project Process Book

Ben Shryock

Table of Contents

[Initial Project Proposal](#)

[Milestone 1 Update](#)

[A Change in Data](#)

[Time Spent](#)

[A Change in Code Structure](#)

[Stacked Histogram: Proof of Concept](#)

[A Completed Prototype](#)

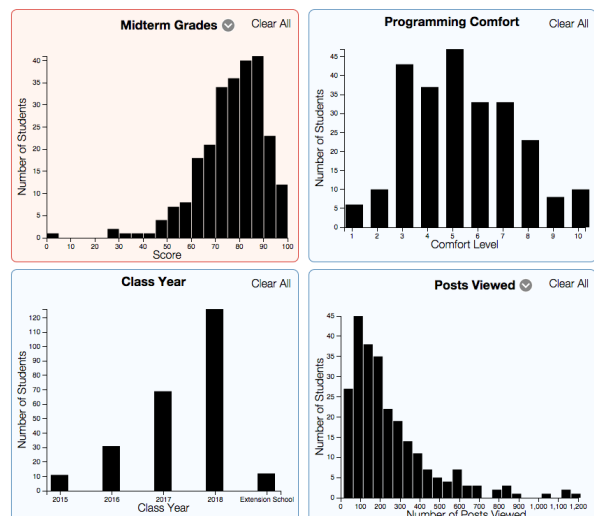
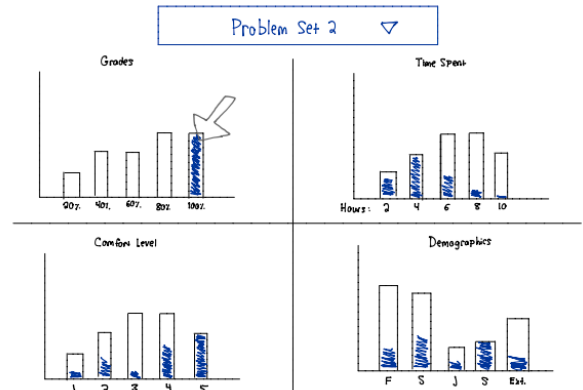
[Color](#)

[More Clicking](#)

[Final Solution](#)

[Results and Conclusions](#)

[Future Work](#)



Initial Project Proposal

Submitted April 3, 2015

Background and Motivation

This project involves data collected from the last two years of Computer Science 51 (Spring 2014 and Spring 2015) at Harvard. I acted as a Head TF of the course during this time, and have been part of the teaching staff for four other courses during my time at Harvard. I am interested in education (with most of my experience being in Computer Science education at the university level), and I believe that CS51 has unutilized data that could lend insights into the experience of students and how they progress over the course of a semester.

Project Objectives

There are multiple populations that this visualization could benefit, including both students and CS educators.

For students of the course, I hope to answer questions related to performance (grades) and effort (time spent per problem set). This allows students to better understand how they're doing in the class and what steps they should take to improve their experience.

For CS educators, there are a number of interesting questions that can be investigated from this data. What is the relationship between time spent on a problem set and score achieved? What is the relationship between a student's final grade and their behavior in the course's online discussion forum? How much progress do students that initially describe themselves as less comfortable make compared to students that initially describe themselves as more comfortable make over the course of the semester? I hope to create exploratory visualizations that reveal any answers to these questions, in addition to the more explanatory visualizations discussed above.

Data

The data has been collected over the past two years of CS51. Thus far, it comprises the results of an introductory survey, grades per problem set, time spent per problem set, and statistics from the online discussion forum (Piazza). This is four separate data sources, at least some of which can be linked through common fields (*i.e.*, email address, HUID). About ~300 students have taken the course each of the past two years, generating thousands of data points across these different sources.

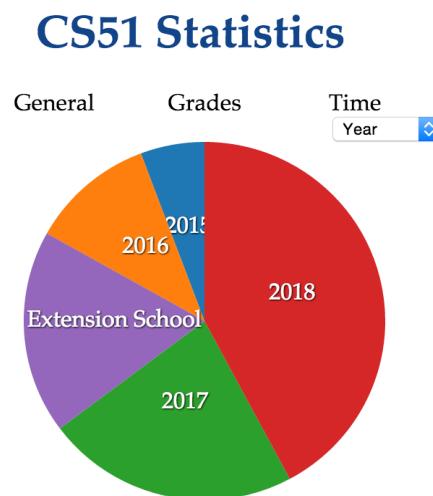
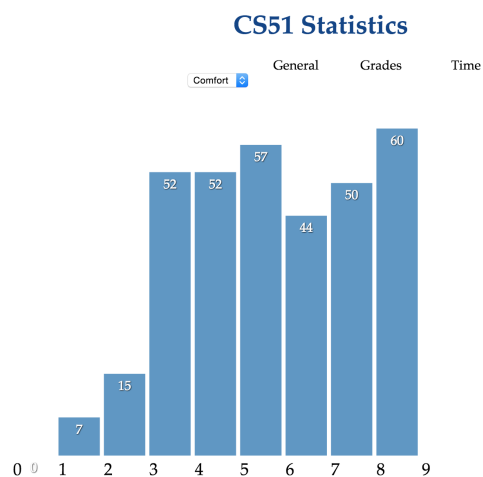
Data Processing

I've already performed most of the data cleanup for basic visualizations, such as histograms of time spent per problem set and grades per problem set. This has been done through the creation of Python scripts to perform general data wrangling and the removal of sensitive information. The outputted CSV or JSON files are then loaded using d3.

There will continue to be data processing necessary to hide any personally identifying information, particularly when dealing with sensitive information like grades. Further data processing will be necessary to connect these four separate data sources through shared attributes, unlocking a greater variety of comparative visualizations.

Visualization

The student facing designs will be relatively simple, designed to effectively communicate basic information about the course as a whole. I have already implemented the basics of these charts, a sample of which are shown below. The first is a histogram showing the self-described comfort level of students (on a 1-10 scale), taken from a survey given at the beginning of the course. The latter is a pie chart depicting the student makeup of CS51 2015 by year. (Note: these are not final products, and as such have elements like overlapping text and unclear labels)



Though the other student facing designs are primarily histograms (problem set scores, time spent per problem set), there are more interesting potential visualizations intended for CS educators. A couple possibilities are sketched and explained below.



Time Spent



Demographics

Another potential visualization, particularly useful for exploration, would be to examine the relationship between final grades and various discussion forum behaviors. For instance, are the students that frequently ask, view, or answer questions more likely to perform well in the course?

Another possible area of investigation is in the performance of low-comfort and high-comfort students on the first and last problem sets. Over the course of the semester, do the scores of the low-comfort improve? Do the students catch up to others that were initially more comfortable than themselves?

Must-Have Features

- A set of student-facing visualizations that answer the fundamental question of “how am I performing in this course” for at least 1 year of data
- A more interesting educator-facing visualization that provides insight into the performance and progress of students in an introductory CS course
- Some exploratory visualizations investigating the effect of various metrics on student grades

Optional Features

All Visualizations

- Incorporate multi-year data
- Incorporate transitions

Student-facing Visualization

- Add ability to input score/time spent, highlight appropriate histogram

Project Schedule

Deadlines

Sunday, April 5: Finish student-facing visualizations

Friday, April 10: Exploratory analysis of grade-related factors, linking data sources

Friday, April 17: *Milestone 1*. Educator-facing visualization draft

Friday, April 24: Finish educator-facing visualization

Friday, May 1: Set up website, finalize process book, complete peer assessment

Wednesday, May 5: *Project Due*. Submit project

Milestone 1 Update

April 17, 2015

Overview

The following few pages contain a discussion of the updates I've made to my final project (from plans to implementations) while preparing for the first milestone over the past two weeks. In general, I focus on only the things that have changed from the initial project proposal. Moving forward, I will update the process book on a more frequent basis.

Questions

As I continue to explore my data, I may revise the questions I'm seeking to answer. However, my visualizations currently focus on the question of "what factors relate to a student's performance in an introductory computer science course?".

Data

I have acquired data from three sources pertaining to Computer Science 51's 2015 offering. These include a welcome survey, student grades, and student behavior in our online discussion forum (Piazza).

Each of these data sources contains an email address, which I've used to connect the separate sources into singular data objects for each students. There were approximately 330 students that filled out the welcome survey. A number of these dropped the course (and some may have used different email addresses in these different data sources), which resulted in a total of 250 students for whom I have data from each of the three sources. Given our final enrollment numbers, this seems like a strong set of data.

Below is a list of the attributes I have extracted from each source:

Welcome Survey

- Class year (2015, 2016, 2017, 2018, or extension)
- Concentration
- Programming comfort level (1-10 scale)
- Operating System
- Whether or not the student has taken CS50

Grades

- Grades on PS1-PS7 and midterm

Discussion Forum

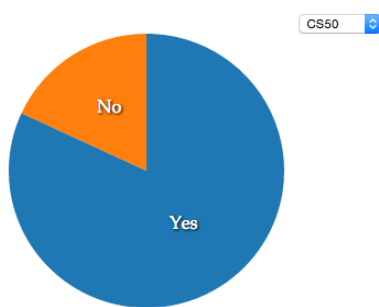
- Number of days on which the forum has been accessed
- Number of unique posts viewed
- Number of questions asked
- Number of answers given
- Number of contributions (includes questions, answers, and comments in followup discussions)

As mentioned in my initial project proposal, I also have access to data showing the amount of time students spent on each assignment. However, this data was only collected on a subset of the problem sets, only collected in 2014, and doesn't have a shared identifier with any of the other three data sources. As such, I may continue my project without utilizing this data.

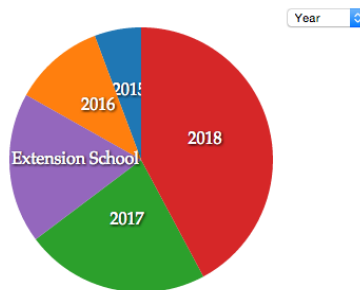
After using the students' email address to connect the three data sources, as much personally identifying information as possible was removed from the representation of a student. This is done with Python, and no identifying information should touch the browser.

Exploratory Data Analysis / Implementation

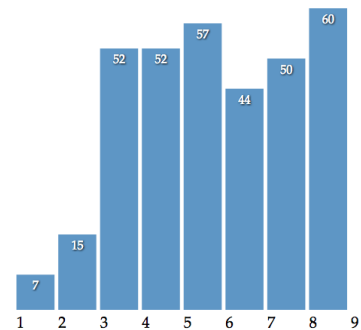
I began by separately analyzing the data in each of these sources. A sample of the created charts and their meanings are shown below.



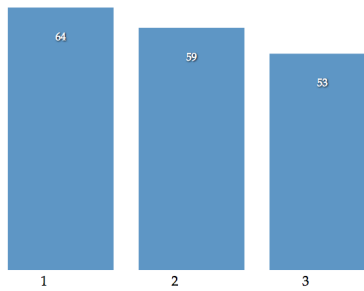
If a student has taken CS50



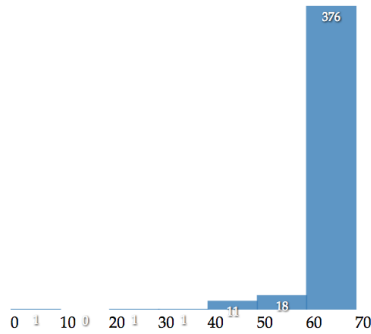
A student's class year



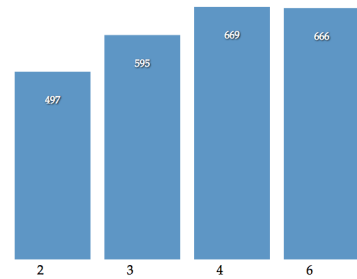
Comfort Level
(x: 1-10 scale, y: number of students)



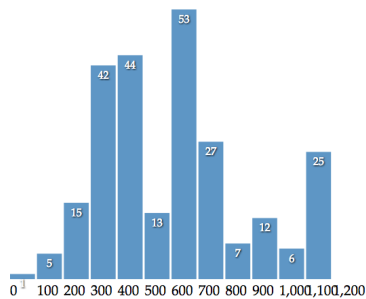
Average pset performance
(x: pset number, y: average score out of 75)



Performance on a specific pset (pset 1 shown)
(x: score bucket, y: number of students in range)



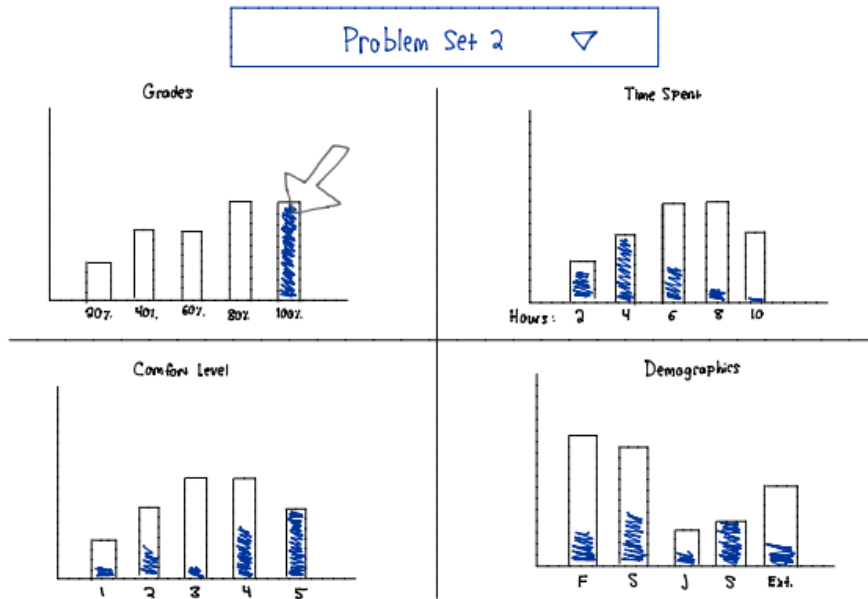
Average time spent on psets
(x: pset number, y: number of minutes spent)



Time spent on a specific pset (pset 3 shown)
(x: minutes spent, y: number of students)

Design Evolution

I still am pursuing one of the ideas initially discussed in my project proposal.



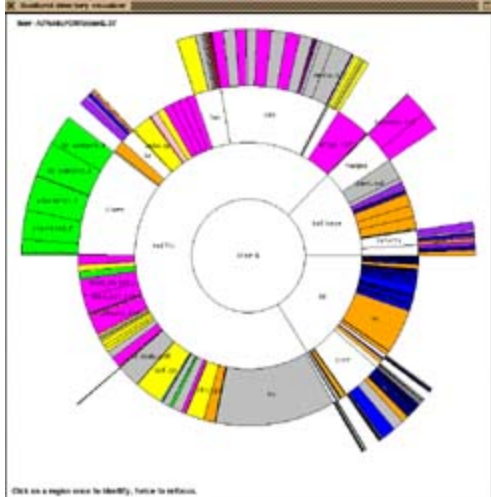
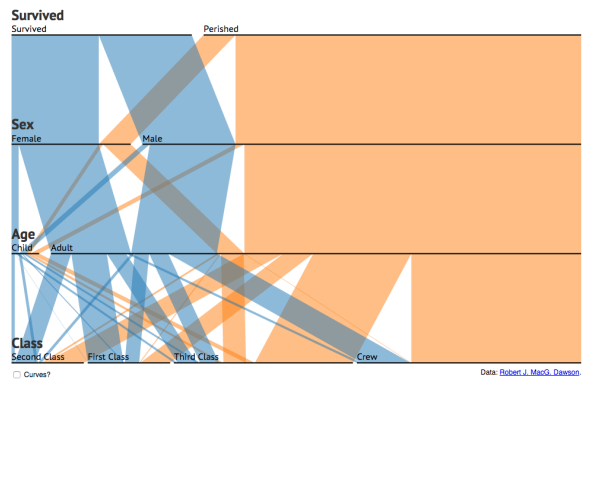
The general idea has not changed, and is reproduced below for ease of reference.

“This visualization shows several charts in one screen. It initially provides a general representation of things like grades and time spent on a specific problem set alongside things like comfort level data and demographics data. Though this could function in a variety of ways, one idea is to have the “grades” histogram be the main visualization. When a bar in the grades histogram is moused over, the bars in the other graphs fill with color (become stacked bar graphs) to represent the data that corresponds to the moused over bar and the data that doesn’t. In the example shown in the diagram, when the 100% score is moused over in the grades diagram, the other graphs reveal that those students that received a 100% were likely to spend less time on the problem set, come from a higher comfort level, and are evenly distributed across the various class years.”

I have made two revisions to this idea in response to feedback from my advisor and design studio peer review. The first is to remove the central cross, making use of proximity to show the relationship between the graphs. The second is to allow for multiple bars in the primary chart to be selected instead of just one.

I feel this is a strong primary visualization due to its explanatory power. However, it does lack in visual appeal.

There are other visualization formats I plan to pursue. As in my initial project proposal, there is still exploratory analysis I'd like to do to assess potential correlations between the data attributes I've collected. Additionally, I've been considering other primary methods of visualization that could be more visually appealing than group bar charts. These include the sunburst and parallel sets visualizations (shown below).

 <p>A sunburst chart titled "Standard Hierarchical Visualization" showing the hierarchical distribution of Titanic passengers. The inner ring represents the primary category "Survived", which is divided into "Survived" and "Perished". The middle ring shows the secondary category "Sex", divided into "Female" and "Male". The outer ring shows the tertiary category "Age", divided into "Child" and "Adult". The chart is color-coded by "Class", with "First Class" in blue, "Second Class" in orange, and "Third Class" in green. The chart is interactive, with a tooltip showing the count of passengers for each category.</p>	 <p>A parallel sets chart titled "Survived" showing the relationships between the categories "Survived", "Sex", "Age", and "Class". The chart is divided into four main sections: "Survived", "Perished", "Sex", and "Age". The "Survived" section is further divided into "Female" and "Male", and the "Perished" section is divided into "Child" and "Adult". The "Sex" section is divided into "Female" and "Male", and the "Age" section is divided into "Child" and "Adult". The "Class" section is divided into "First Class", "Second Class", and "Third Class". The chart uses a color scheme where blue represents "First Class", orange represents "Second Class", and green represents "Third Class". The chart is interactive, with a tooltip showing the count of passengers for each category.</p>
<p>Sunburst visualization</p>	<p>Parallel sets (visualization technique for multi-dimensional categorical data)</p>

A Change in Data

4/20/2015

At this point, the visualization I had created was a disjoint set of bar charts, pie charts, and scatter plots. It was split into three tabs (general statistics, grades, and time spent), each of which came from a different data source. These data sources were not linked.

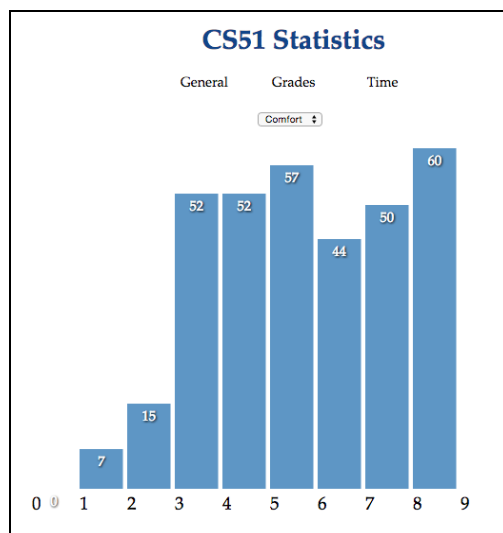


Figure 1. Old model

In striving to create my linked bar chart visualization, I needed to connect these data sources. All of the data sources contained email addresses, which could be used to link a student's demographic data to their grades and discussion forum usage. For 2015, the demographic survey (given at the beginning of the semester) had just over 300 responses. After linking grades and forum data by email address, I was left with 250 full student profiles. Given the attrition rate of the course, this is a solid sample of students.

May 1 Note: This effect was similarly observed when I added the data from the 2014 offering of the course.

Time Spent

4/20/2015

The data about how much time students spent on a problem set was unreliable for the following reasons. First, it is self reported data. Students are asked to estimate the amount of time spent on a problem set after completing it, and many provide false values (like the maximum value of an integer). Second, due to oversights when administering the course, the data was only available for half of the problem sets. Finally, though I wrote a python script to gather this data for the 2014 version of the course, the data was not easily accessible for the 2015 version of the course. For these reasons, I decided to remove time spent from my project.

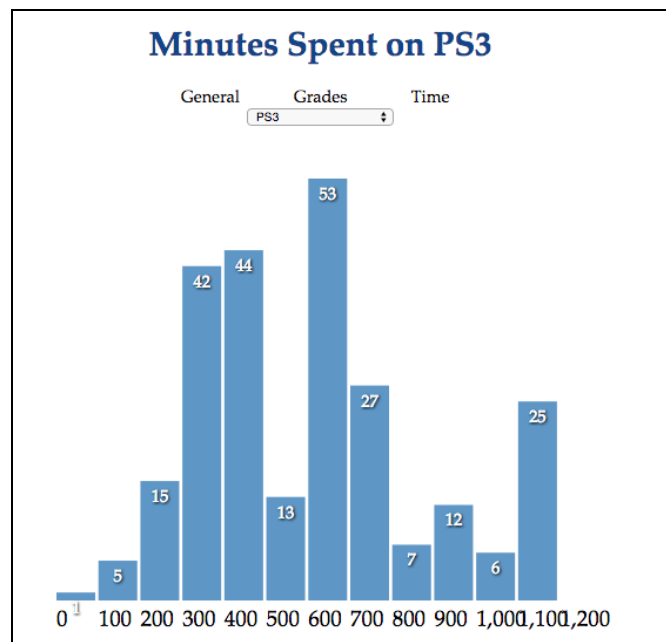


Figure 1. Goodbye, time spent data

A Change in Code Structure

April 27, 2015

I started my existing implementation before completing CS171's HW3. As such, I structured my code in the same format I had used on the first two problem sets. This format was acceptable for the single charts that I initially implemented, but I didn't feel it was suited for multiple, interactive charts. I also erred by making many functions in this initial implementation too general, to a point that it was difficult to read and understand the code.

For these reasons, I decided to rewrite the project using CS171's HW3 layout as a template. This implementation centers around the creation of separate classes for each visualization that appears on the web page, with the help of a shared event handler to make the visualizations respond to each other.

I was very happy with this choice, and though it involved re-implementing some aspects of the project in the context of the new framework, I believe it resulted in a much better product.

Stacked Histogram: Proof of Concept

April 27, 2015

One of the central aspects of my design is that of a stacked bar chart. When a bar in the grades chart is selected, the other charts should update to show where those students can be found.

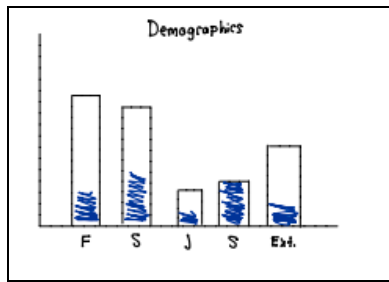


Figure 1. Initial sketch of stacked bar chart

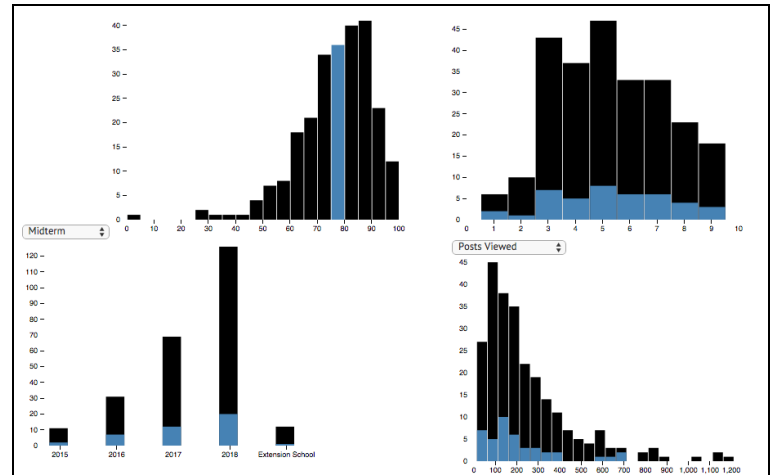
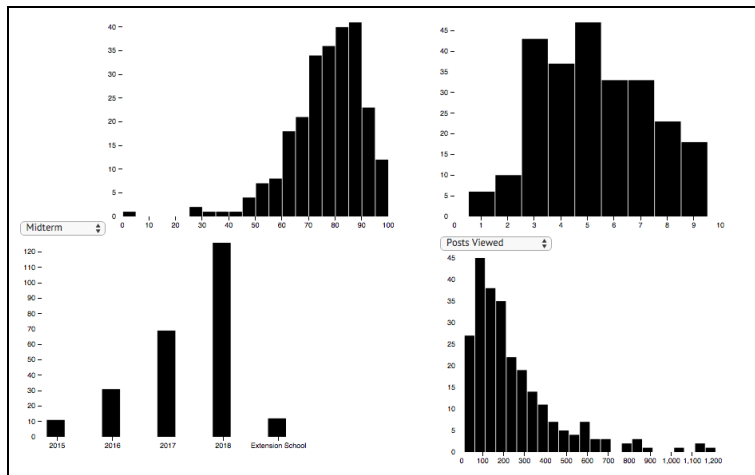
As in the image, this information is portrayed in a stacked bar chart. However, I implemented the grades, online forum, and comfort charts using the d3 histogram layout. And while some resources exist for this layout, and some resources exist for stacked bar charts, there are none that combine the two.

I solved the problem by creating a second histogram that would overlay the first, simulating a stacked bar chart. Creating this proof of concept was a significant step in showing that my project would be achievable with minimal changes.

A Completed Prototype

April 28, 2015

I finished an interactive prototype of the application. It consists of four bar charts, containing data for grades, comfort, online forum, and class year (clockwise from top left). Upon clicking a bar in the grades chart, the other charts update appropriately to show where those students appear.



Update. I have also added the ability to click on multiple bars within the grades chart. This allows the user to answer questions like, “what is the demographic data for the students that performed above the mean on an assignment?”

Color

4/28/2015

After creating this prototype, one flaw seemed to be a lack of indication of which charts were clickable (or had been clicked on). I initially intended only the grades chart to be clickable, but after clicking a few bars, there no visual encoding differentiating it from the other charts.

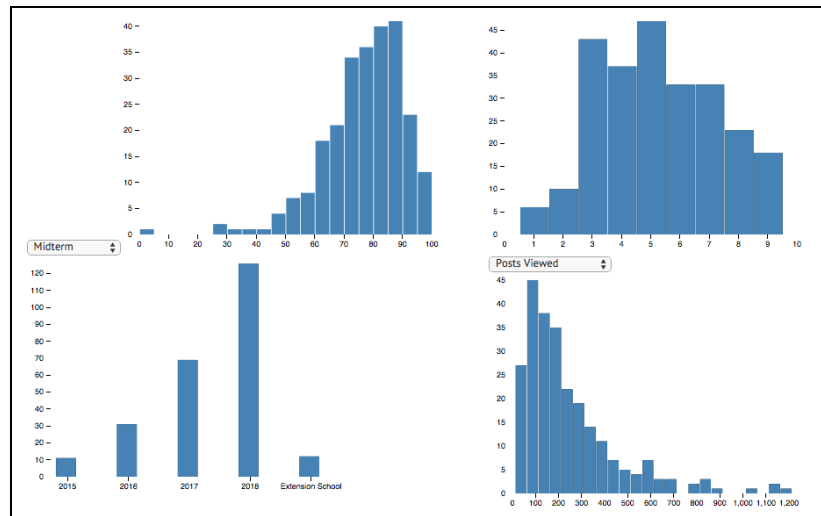


Figure 1. After clicking all the bars, there's no differentiation between the clickable and non-clickable charts

To combat this problem, I decided to make the clicked bars a different color. The independent variable (the students' grades) will be red, and the dependent variables will be blue, thus utilizing color as an indicator of how the application can be used.

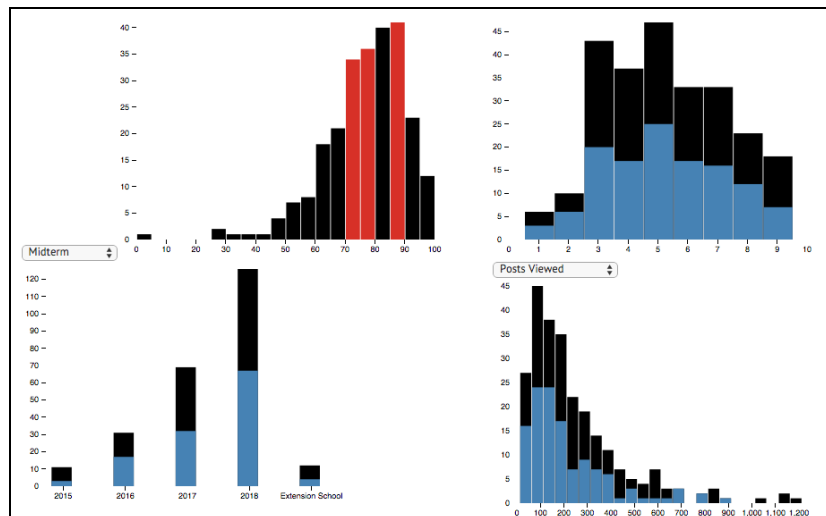


Figure 2. Clearer clicks

More Clicking

4/30/2015

Yesterday, I decided that the next useful extension to this project would be the ability to click on any of the charts. Though the initially intended idea was to allow only the grades chart to be clickable, it now seems that this would be an unnecessary limitation on how the data can be explored. For instance, it is within the intended objective of my application to be able to answer questions like, “how did the students that rated themselves as most comfortable perform on problem set 1?”

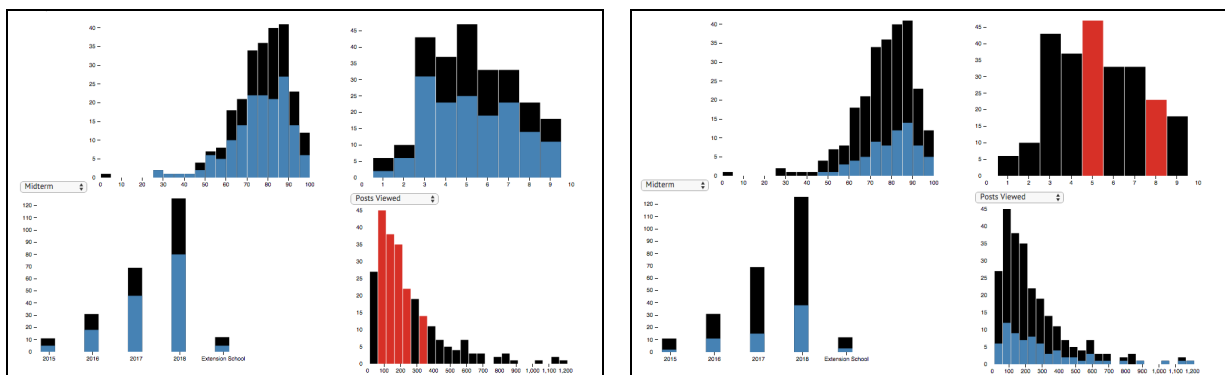


Figure 1. All charts are clickable

Further, each chart is given identical weight in the current interface. As such, it would seem strange if only one chart were clickable.

I also made the decision that it would not be worthwhile to allow multiple charts to be clickable. Though the added specificity could be useful in some cases, the combinations of these charts do not provide particularly interesting data.

Final Solution

5/3/2015

I made a few more important changes before arriving at my final visualization.

Clear All

When exploring the dataset, I often found myself clicking on a number of bars, and then wanting to unselect all of them. To facilitate this process, I created a “Clear All” button to improve the user experience.

2014 Data

I wrote an additional python script (a modification of the existing script) to import data from 2014. In the visualization, I created a toggle to switch between these two data sets. Incorporating the 2014 data into the existing classes for the visualization required minor generalizations to the code.

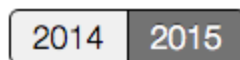


Figure 1. Year toggle

I decided not to create a way to view a combination of the two data sets. This could be a compelling area for future work, but I didn't want to spend my remaining time normalizing the differing grading schemes between the two years.

Styling

From axis labels and chart headings to general page layout, it was time for some styling.

I also added further visual encoding to show the user which chart was selected, and which charts were responding to changes in the selection.

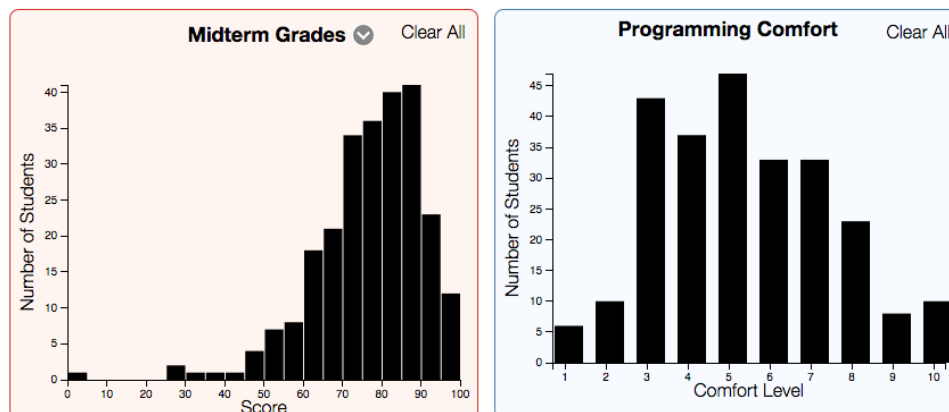


Figure 2. Styling

Tour

There were a few key pieces of information that I wanted the user to know when they encounter my visualization for the first time. I felt that it would be important to provide a little more information about where each data source came from (particularly the data pertaining to comfort and online forum usage), in addition to showing how to use the application.

I accomplished this through the use of an interactive tour that guides the user through a few aspects of the visualization.

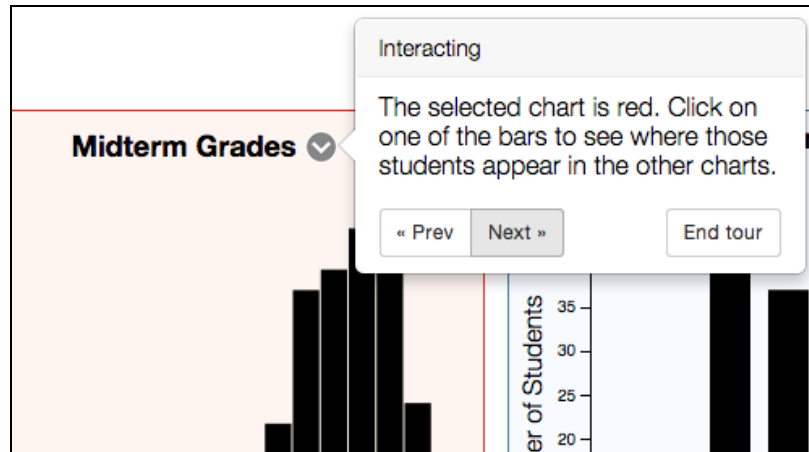


Figure 3. Interactive tour

Results and Conclusions

5/4/2015

Basic Data

Grades

In 2015, all problem sets were out of 75 points. Most students scored over 60. In 2014, each problem set had a different number of possible points, but a similar effect was observed.

Comfort

In both years, students reported similar comfort levels. The distribution of comfort levels was approximately bell-shaped, with the most frequent scores being between 3 and 8 (on a 1-10 scale).

Year

Both years, the number of freshman exceeds the number of sophomores, which exceeds the number of juniors, which exceeds the number of seniors.

Online Forum

In both years, students viewed between 0 and approximately 1500 posts on Piazza. The max number of contributions was about 200. There is slightly more data from the year of 2014 because the 2015 offering of the course has not yet finished.

Trends and Relationships

Upon initial inspection, there were fewer clear trends in the data than I was hoping. However, I was able to uncover some interesting relationships using my visualization.

Comfort Level

The self-reported comfort level of the students at the beginning of the semester had little impact on their success on the problem sets. The top performers on each problem set were evenly distributed amongst the comfort levels. There are a variety of potential explanations for this observation. One I would like to mention is that the course is taught in Ocaml, a language that almost no students have previously encountered. This is explicitly done to provide students with an even playing field to learn about abstraction and design, and the observed results could indicate that this goal is accomplished.

Discussion Forum

In 2015, none of the students that viewed more than 450 Piazza posts scored more than 1 standard deviation below the mean on the midterm. This effect persisted for almost every problem set. These students were evenly distributed amongst comfort levels and class years.

The students that make the most contributions on Piazza are evenly distributed across comfort levels.

Future Work

5/5/2015

There are a variety of large and small tasks that I would be interested in further pursuing for this visualization. I'll include a sampling of them below.

Additional Questions

There are a number of interesting comparative questions that this visualization could do a better job of answering. For instance, there is good way for the user to answer a question like, "how did the least comfortable students perform on the first problem set compared to the last problem set?". While the user could separately perform each of these queries, I could adjust the visualization to allow for this data to be more easily displayed.

Combining Years

As mentioned above, I provided users with the ability to view course data from 2014 or 2015, but not a combination of the two years. This was due to a lack of time to spend working out the differences (such as the different problem set grading practices) between the two years of the course. Providing this option would increase the size of the dataset and remove a distinction that is not necessary for all queries.

Interface Improvements

While I was able to make a functional interface, there are still a number of design improvements that could be made.

One useful feature would be the ability to select of the bars on the chart, reducing repetitive clicking when the user wants to select a large number of bars.

Multi-color Stacking

To more directly compare data within the chart, I could allow for a multi-color stacked bar graph. For instance, the user selects the top midterm performers and their demographic information appears on the other charts in blue (as happens currently). The user then clicks a button or holds down a modifier key and selects the lowest midterm performances. The demographic information of these users appears in green. This would better allow for direct comparisons of groups of students within a single chart.