CS 3300 Project 2 Report

Joe Antonakakis - jma353

Yevhenia Stets - yvs4

Part A

A description of the data. Report where you got the data. Describe the variables. If you had to reformat the data or filter it in any way, provide enough details that someone could repeat your results. If you combined multiple datasets, specify how you integrated them. Mention any additional data that you used, such as shape files for maps. Editing is important! You are not required to use every part of the dataset. Selectively choosing a subset can improve usability. Describe any criteria you used for data selection. (10 pts)

Description

In order to compare universities and group them by metrics that people would care about, we looked through several sites for the most comprehensive lists of universities matched to SAT / ACT scores, admissions rates, and graduate salary reports. We wrote scrapers in Python to obtain various sources' info. Below is a list of the sites we used:

```
http://www.payscale.com/college-salary-report/bachelors?page=1 // Salaries

http://www.collegesimply.com/guides/1600-on-the-sat // SAT + admissions

https://developers.google.com/maps/documentation/geocoding/intro#Geocoding //
Geocoding

http://www.collegesimply.com/guides/low-acceptance-rate/?view=all // Not
currently used

https://www.powerscore.com/sat/help/average_test_scores.cfm // Not currently
used
```

The sources marked `// not currently used` were not used because they featured data that was present on the **College Simply** website, a source we found later in our search for data. We limited our actual college data acquisition to two sources because interesecting data sets for universities with several commonalities across names provided to be challenging and error-prone across >2 data sets. The **Payscale** site had an entire JSON *in-line* within their sites HTML, and this JSON provided us salary info and zip codes which we would use later. The

College Simply site provided us SAT scores and admissions rates for universities. Next, we signed up for an API key for Google's geocoding API, and ran our one JSON through this API to retrieve latitude and longitude coordinates. Finally, we intersected our two JSON lists based the names of universities. We checked these by splitting the names by spaces and checking each string. If a threshold number of similar strings were the same, the data was intersected and formatted for our final JSON. These threshold coefficients varied depending on what kind of name was presented (for example, a much higher threshold was required to match a name containing the words `University of` because of the sheer number of universities with this series of terms in their name). This method was completely accurate, as we both spot-checked our JSON afterwards. Our final JSON format, which was completely engineered by us, was the following (utilizing Stanford as an example):


```
{
    "location_info": {
        "lat": 37.4135757,
        "lng": -122.1689284,
        "state": "California"
    },
    "school": "Stanford University",
    "link": "/colleges/california/stanford-university/admission/",
    "salary_info": {
        "early_median_salary": 65900,
        "mid_career_median_salary": 123000,
        "rank": 7,
        "zip_code": "94305"
    },
    "score_info": {
        "sat_reading_low": 680,
        "sat_reading_high": 780,
        "sat_math_high": 790,
        "sat_math_low": 700
    },
    "admissions_info": {
        "applied": 36632,
        "acceptance_rate": 7
    }
}
```


Part B

A description of the mapping from data to visual elements. Describe the scales you used, such as position, color, or shape. Mention any transformations you performed, such as log scales. (10 pts)

Description

Our first challenge was mapping latitude and longitude values to the U.S. map used. We used a TopoJSON projection to project our latitude and longitude tuples onto the U.S. map and create a series of dots representing universities across the country. We then built functionality to "throw" these dots onto a xy-axis to the right of the map on user request of k-means clustering. The user could pick two parameters of the three we looked into (SAT score, admissions rate, and median mid-career salary of graduates) and a number of states with universities. The user would then click `Cluster` and new dots would form on top of the dots on the map and throw themselves onto the xy-plane with axes labeled according to the chosen paramters. This involved linear scaling of the axes, offset computation from the map to graph, and 5-means (we chose **5** as our **K**) clustering of the chosen states' universities. On hovering over the map dots *or* the clustering diagram dots, the user could see information about the university represented by the dot above our graph.

Our clustering was normalized given the data spread of the states chosen by the user. For each parameter, the difference between the max of the data set and the min of the dataset was computed for that parameter. For each data value of that parameter, the minimim was subtracted. The following arithmetic represented the value that was used for each data point in k-means clustering:

```
(data_val - min_of_set) / (max_of_set - min_of_set)
```

This was done to reduce the influence of particularly big numbers on the clustering. If this wasn't done, relatively small differences in, let's say, salary would cause data points that *should* be clustered to not be clustered.

In addition to clustering, we ran regressions on the 3 different possible subsets of the parameters we chose for all universities. These linear regressions featured the linear regression algorithm discussed in class and linear axes scaling. The regression was computed with the raw values of the data, but the coloring of the points was scaled. In general, these regressions aimed to show trend *and* accentuate outlier universities that didn't follow the general flow. Utilizing the same scaling as above, coefficients were computed, indicating how far off each data point was from the regression line. This was used to adjust the opacity of the point. One color was used, but opacities varied. If the point was off relative-25% or less, 0.25 opacity was used. If the point was off less than relative-50% but greater than relative-25%, then the opacity was 0.50, and so on. Labels with each univeristy, relevant info, and these regression discrepancy percents are presented on hovering over the dots for each regression plot.

Part C

The story. What does your visualization tell us? What was surprising about it? (5 pts)

Description

Our visualization allows the user to see the differences and similarities between US universities in different or same state based on the 3 variables: SAT score, acceptance rate, and mid-career median salary (this is the map and clustering). It also shows the general trends for universities when comparing 2 of the 3 variables(this is the regression graphs). In general, it is not surprising to find a positive linear relationship between acceptance rate and SAT score, or the positive linear relationship between salary and any of those 2 variables, since usually the more selective universities yield higher-paid graduates. What is of interest though, is the amount of variation in one variable of universities that are close to each other in the other variable. One of the most extreme examples of this would be College of the Ozarks, which has acceptance rate of only 9%(less than Cornell, close to Stanford University), but SAT of only 1034 and salary $71,800 (substantially less than Stanford's 1475 and $123000).

In general, we can observe universities like the one above, which is completely out on its own, as well as universities that are relatively "more efficient" in their paramaters. For example, Stevens Institute of Technology has a 40% acceptance rate, an SAT of 1300, but a median mid-career salary of $120,000, which is higher than Cornell's. Obviously Stevens is primarily Engineering-based, and with that profession comes a higher lifetime salary, but, if one was trying to get into a university and didn't have the credentials for an Ivy League or a similarly-ranked school, Stevens is one of the best with that acceptance rate and SAT. This can be discovered by looking at our clustering and regressions, as Stevens is in the categories of Ivy's for some clustering runs, and a clear outlier on two regression graphs.

On the opposite end of the spectrum, certain schools appear to be less efficient, requiring a higher SAT score or more harsh admission statistics, only to yield a lower median career salary. Schools like Wellesley College fit this category (30% acceptance rate, 1410 SAT score, $78,400 mid-career salary).

In general, our implementation is built as a **tool for discovery**, and it utilizes data-science techniques to present a user with university groupings in a way that has never been done before.

Outside Sources

Outside of our data sources, we referenced various sites for certain purposes:

Used for map : https://bl.ocks.org/mbostock/4090848

Used for mapping out universities based on latitude / longitude :
http://chimera.labs.oreilly.com/books/1230000000345/ch12.html#_adding_points

Used for dot plot inspiration : https://bl.ocks.org/mbostock/3887118

Used as a tool for button styling : http://css3buttongenerator.com/

moveToFront() and moveToBack() methods from :
http://stackoverflow.com/questions/14167863/how-can-i-bring-a-circle-to-the-front-with-d3

Used for general linear regression info :
https://en.wikipedia.org/wiki/Linear_regression

Sketch software (https://www.sketchapp.com/) for popover SVG generation