Alan Jiang, Srishti Tyagi, Emma Wang, Chelsea Choi
INFO3300 Project #2

Final Report

Data Description:

Main Data

-https://www.kaggle.com/winston56/fortune-500-data-2021 (Fortune 1000)

-https://www.kaggle.com/jackogozaly/data-science-and-stem-salaries (Levels.fyi)

- https://worldpopulationreview.com/state-rankings/cost-of-living-index-by-state (Cost of Living)

Map Data

-us.json

-map.json

-us-state-names.tsv

Our data consists of three main datasets from Kaggle. The Fortune 1000 dataset is a list of Fortune 1000 companies, the top 1000 corporate companies in America. For each of these companies, the characteristics of rank, revenue, past rank, number of employees, location of headquarters (city, state), sector, and more are available. The Levels.fyi dataset contains the salaries of data science and STEM jobs for companies all over the world. Each value in the dataset has characteristics like company, years of experience, level, title, yearly compensation, years at the company, base salary, and more.

The Cost of Living dataset, or COL, employs a metric that compares the city to NYC as its baseline. In calculating this cost of living, the dataset combines multiple categories such as rent, groceries, etc to arrive at this number.

To clean the data, first the Fortune 1000 dataset was filtered to contain only the technology sector. The cleaned Fortune 1000 dataset was remapped into a dictionary where the key is the company name and the value is all other attributes of that company including rank, revenue, past rank, etc. The Levels.fyi dataset was then filtered to only contain companies with the same names and city location as those found in the Fortune 1000 dataset.

Extensive data cleaning and manipulation was done with our data and a brief summary is below. To get the longitude and latitude of each city the companies reside in. First, every unique city was obtained from the Levels.fyi dataset using a filter. Then, the cities were looped through to be mapped to their corresponding latitude and longitude from the city.json file. Since some cities were not present in the

city.json file, a few additional latitudes and longitudes had to be hardcoded and pushed into the file. A new array was also generated and added to each city, so that each city will have information on all the companies in that city.

We wanted to create a choropleth map so we needed to create a new dictionary. First, filtering through the Levels.fyi dataset, every unique company was used as a key and the value was another dictionary with two keys - state (with a string of the state abbreviation) and an array of salaries for that company as ints. Looping through this dictionary, we created another dictionary that mapped state abbreviations to the average salary across all the companies of that state. This was done using a helper function that calculated the sum of every company salary list through the loop that recalculates the average salary each time a new company contributes to a state. Using this dictionary, we were able to create a colorScale and an anonymous function that maps the correct color to the state using us-state-names.tsv.

For our tooltip and plotting of cities that have company headquarters, we combined the three data sets by calculating and adding the average salary, cost of living, and associated list of company headquarters in that city. We used the new city array from earlier as the dataset to hold all of the information. The COL column in the cost of living data was joined with the new city array through the city column. In order to do so, the values in that column were casted as ints. Next, we calculated and added each company and their respective average total compensation as a dictionary to the new city array. This value was calculated through iterating through the levels fyi data and maintaining two dictionaries. One for the number of entries for that headquarters, the second for their total compensation added up. After adding the companies and average salaries, we were able to finally have the information we required to display on the tooltip when a city was clicked on.

Overview of Design Rationale (Static)

We used a map of the USA to give users a visual representation of the location of the cities in out dataset. We used a mesh to outline the borders of each state to give users an idea of how the states are separated. The choropleth map is the foundation of our project and many trade-offs were considered with sacrifices made in this scenario. During the data cleaning process, we noticed that there were a limited number of states that met the criteria of being in the Levels.fyi dataset and in the Fortune 1000 dataset. The data was mostly distributed on the coast with some companies sprinkled in the midwest region. Because of this bias, we decided that it would be beneficial to see where high-revenue company headquarters are rather than just focusing on one state. Therefore, we opted to keep the map. For the color

scale of the states, we used the average salary of the state across all companies. A color theme was chosen from colorbrewer to make sure that luminosity and hues were intuitive for users to understand the salary ranges represented by the colors.

We also plotted the headquarters of each Fortune 1000 that was chosen. Although Level.fyi provided non-headquarter locations, we opted to only select headquarters as HQ because they are usually in more highly coveted locations in metropolitan areas. Therefore, this would provide a more realistic view of where students looking for jobs might want to work. Instead of representing every single headquarter location (even if they were in the same city), we utilized a dictionary, as described above, so that a singular point on the map could represent one city that contains multiple companies. We opted for a neutral black color for the point, as we did not want to mislead users into thinking the color of the point had a meaning. To make the city points stand out better, a soft pink outline was used to make the circles pop. We tried different colors like red and blue, but those colors were too jarring against the background.

We also added a legend below the map to signify the salary ranges of each state. Since a quantize scale was used to visualize the average salaries per state in the choropleth map, the legend was made using code created by Prof. Rz so that the boundaries of each quantized “bin” could be visualized for their colors. The title and bottom border were given a bold blue color to encapsulate the visualization and give the map a sense of space rather than being open on a white page. A description about how to use the visualization was also included below the title so that users know how to navigate/interact with the map. A short title was also added to the legend so the coloring/meaning was more obvious for the viewer.

Overview of Interactive Elements and Design Rationale

To make our map more usable, we added a zoom feature and a zoom out button to our map. This follows the HCI design principle of consistency, as other interactive maps often have this feature as well. This can make the map more usable as there is more familiarity in the functionality of our visualization. Another benefit of the zoom feature is that it can allow the user to more easily focus on points and regions that they are more interested in. We also allowed the user to be able to zoom into the map just by clicking on a state. This is because when exploring a map, people usually look at one region at a time rather than jumping around and looking at New York and then California, for instance. This is especially true in our case, since our data is about the relationship location and salary data and the types of people looking at the visualization — such as Cornell graduates looking for jobs and places to live — are those who may want to live in a particular region, such as the East Coast. We also added a zoom out button so that the user can easily navigate to other parts of the map without having to do a lot of panning and zooming. We

chose not to add a zoom-in button because the level of detail in our map doesn't warrant a zoom in button and going back to our potential user audience, our users are likely to be more interested in particular regions and a zoom-in button would not allow them to be able to pinpoint specific locations of interest the same way that clicking on a state can.

After combining the three datasets, we utilized the information from these data sets in creating the tooltip and circles. When users click on a point on the map, the point will turn white and a tooltip will pop up near the state. This tooltip contains the companies, average salaries, and city. By doing so, users can click on cities of interest and find information about the company headquarters that exist there and their average income. Moreover, this allows users to directly see how the average salary of a company located on the coast compares to the midwest or a southern state. This feature was also useful particularly for densely clustered cities in a state like California, where without this feature, it would be very difficult to tell which city the user clicked on without a clear indicator like color. The user can make the tooltip information disappear by simply clicking on a state, but outside of the data points.

Design Iterations

We considered other ways of displaying the name of the city at each point.. In an earlier iteration, when users zoomed in on the map, they would see the city names. However, we found that areas with a high density of cities with headquarters would not allow clean labeling of the circles at the same time. We also tried to allow users to hover to retrieve city names. This would allow users to quickly find and focus on the circle representing the city they are interested in learning more about. However, we found that this collided with our tooltip when the user would hover and click on the city name and the name of the city would be covered by the tool tip, making the visualization look cluttered. Therefore, we directly printed the city inside the tooltip so that the city name would only show when the circle was clicked. Although this makes it harder to find a particular city, since intuitively the user would not know which point matched to which city, this method allowed us to convey the name of the city to the user without cluttering our visualization.

Originally, we had also considered using a slider to allow the user to filter which cities are displayed by selecting particular attributes on a slider. After filtering and data cleaning, we realized that the benefits of a slider would be very minimal and not that useful for a user, since relatively few cities are plotted on the map.

Alan Jiang, Srishti Tyagi, Emma Wang, Chelsea Choi
INFO3300 Project #2

The Story

Our target audience is people looking for careers in the technology industry, such as Cornell CIS graduates. More technology-based jobs are centered in clusters, e.g Silicon valley. Because of this trend, most tech jobs require relocation. In the levels.fyi dataset, the salaries and locations are available but the dataset doesn't provide a full picture of what life might be like in that city. When job seekers consider a position, they must think about COL(cost of living) as well, since even if they make a high salary, the cost of living in that region may be high. Our visualization gives our audience an idea of the pros and cons of moving to a different city for a particular job, through information about average salaries in a particular state, the cost of living in that state, as well as companies' salaries. Moreover, our visualization provides users with an intuitive understanding of which states have a higher average salary at the tech company headquarters within that state. The colors of the states instantly provide the user with a hierarchy of how the salaries rank in each state. By doing so, users can gain an understanding of two critical factors that must be considered during relocation, distance and salary.

During the course of creating this visualization, we came across many surprises from the data. One of which was the lack of companies everywhere except California. New York City, Seattle, and Chicago are all well known "tech hubs", but in this visualization the number of cities in California that have tech companies overpowers all the other cities well-known for this fact. This imbalance could be due to other tech companies in these "tech hubs" being mostly startups. In our map, we used the fortune 1000 dataset to filter the companies and therefore cities. Therefore, a future iteration of this visualization could look very different if younger companies were plotted instead. It was also surprising that some states did not have any Fortune 1000 companies at all. This means that for example, people living in the Midwest would likely have to completely relocate to work at the headquarters of a major tech company. We also found it interesting to see the distribution of these headquarters across the United States. We didn't expect to see the headquarters of high-revenue companies in places like Georgia and Idaho. Recent articles hint that tech companies are expanding eastwards from Silicon valley to deepen the pool of talent - which is not necessarily in east coast cities. A possible interesting future extension of this project would be to study future expansion of tech overtime.

Alan Jiang, Srishti Tyagi, Emma Wang, Chelsea Choi
INFO3300 Project #2

Team Contributions:

- Alan: minor Data Cleaning and Manipulation for easy data access, Static portion of map (Chloropleth), Data, Static Design, and Story in the Report
- Srishti: Major Data Cleaning, Static portion of map (Legend, circles, description, color), Report Editing/Overall contributions, Designing/CSS of final visualization
- Emma: Data Cleaning, Made map, Static portion of map, Interactive portion of map (zoom), Interactive portion of map (tooltip) , Interactive on Report, Code cleaning
- Chelsea: Data cleaning for combining data sets, Interactive portion of map (tooltip), Interactive on Report, Code cleaning, editing report

Approximate Hourly Breakdown:

- Foundational (picking a viable dataset and figuring out where to start)
    - (7 hours)
- Data Cleaning and Data Manipulation
    - (20 hours)
- Design how to represent data and EDA (exploratory data analysis)
    - (15 hours)
- Color palette/ finalize graphs (Figma)
    - (4 hours)
- Coding and plotting of graphs
    - (20 hours)
- Data Explanation/most of the writing in Final Report
    - (5 hours)
- Double-checking/ finalize the project
    - (4 hours)

MOST TIME SPENT: CODING and PLOTTING & DATA CLEANING and MANIPULATION