**Weekly Update**

**Oct 19 - Oct 23**

This week, based upon the assumptions we agreed with Cary from Greenwich during last week’s meeting, we started to work on reformulating the dataset and preparing the X variables that we need for future modeling. We each worked on different X predictors this week and planned to join our tables based on the ‘Company’ and ‘Date’ columns into one table next week.

Binqi was responsible for calculating the change in monthly average salary (in percentage) for each company ticker based on Greenwich's job data. She was also responsible for performing data analysis on the financial data scraped by Isaac and calculating stock price change (in percentage) as well as the monthly return for each company. Before entering into the calculation stage, Binqi first created a separate table from the original dataset, removed observations where the Date is after Jan. 2020 (as discussed previously with Cary), and filtered the data to only contain the relevant columns for her part of analysis. The ‘master’ table she worked on has over 16,000,000 rows and 26 columns originally. This first step definitely saved a lot of processing time and memory usage for further analysis. The processing time and efficiency of code is definitely something we should take into consideration in the future modeling stage.

Congda was responsible for calculating the job posting change (in percentage) and average job posting duration days. During the process, he encountered several issues and managed to find the solutions. One of the issues he discovered was that there were missing values in the remove\_date column, which take up around 1% of the total data. This issue can drastically influence the active job postings for each month and potentially introduce errors. Therefore, we as a group decided to communicate this issue with Cary. Cary suggested us to keep these entries and explained that some of the remove\_date are missing because they are still active. Another possible explanation for missing remove\_date is that these job postings might already expire without a specific date for removal. We should keep in mind that these entries should not be counted in the average duration days calculation.

Matt was responsible for engineering the percent change in important roles and training roles for companies. The issue he had mainly dealt with pgadmin4 not being able to handle relatively complex queries, especially with 4 or 5 joins needed for these features. Currently, is dealing with the issue by downloading the entire database. Additionally, issues dealing with inconsistent job tags and titles that contained strings such as “Training Manager” needed to be filtered out while “Training available” should be kept.

Isaac was responsible for getting additional information out of their job post locations. We wanted to test if the number of job postings in urban/rural (populated/not populated) areas and whether posting new job postings in new areas (proxy for expansion) could be predicting variables. Isaac decided that the most reliable address information out of the data was its zip codes, but because zip codes only contain small areas (for example, Evanston’s zip code could not be associated with Chicago), the data needed to an outside source to aggregate zip codes into higher level. Isaac decided to use US Census’s Core-Based Statistical Areas definitions to aggregate zip codes. He also appended zip code population information, so that every record had CBSA code, population. Using  pandas functions, Isaac obtained CBSA population information, and he calculated population percentiles for every zip code and CBSA. And by using groupby and minimum function, he was able to categorize if the record was a new posting into a new CBSA for that company.

Another issue that came to our notice was that, when we tried to calculate change in percentage from month to month, there are cases when the value in the previous month is zero and the value in the following month is non-zero. In this way, the change in percentage cannot be directly calculated by ‘next / previous - 1’ due to the division error. We researched online for multiple possible solutions and one of them seems quite reasonable, which is to add a very small number to the previous value, or add 1 to both previous and current. We are aware of the possible bias this method can introduce, therefore we are planning to ask Diego for advice.

Responding to Diego’s Comments for clarification:

3900 Normalized Companies: Cary mentioned that when data is scraped from various sources with the company’s proprietary software, names of the companies are sometimes misidentified, and therefore cannot link company and sector data to those posts. The unclean data goes through algorithms to be cleaned and identified (what Cary calls “normalized”). He currently has 3,900 unique companies that can be identified from the unclean source, and his team is working to expand those numbers.

Geographic Data: After talking with Cary, we decided to explore geographic trends, as described above in Isaac’s section.

Single Entry: Each row will consist of a company’s feature information for one month’s period, where many features are based on the change from the previous month’s information.