ECON 8600 Midterm
Elizabeth Zarrilli

** Note: For a more detailed description of how to run the individual python files to obtain, parse and clean data, see the github repository .README. This report will focus mainly on reporting and analyzing the data.

This project asked to scrape a list of Github user ids with accompanying data from a provided online database, parse and clean this list of duplicates and invalid ids, then scrape further information on these users from Github.com. Finally, we were to report summary statistics and scatterplots to visualize the collected data.

The code and instructions on how to use the code can be accessed on the github repository. This report will summarize the data collected in each step and provide further explanations/details of the data.

Part 1: Download github user data from charcoal website

The data from charcoalpaper.com is stored in three separate csv files: cleaned_dataset.csv (usernames from "GitHub Data", cleaned_bonus.csv (bonus usernames) and all_ids.csv.

These have a list of all of the unique ids (duplicate ids were removed) and corresponding information from the charcoalpapers.com website. Invalid ids and ids with missing data have not been removed from this data set; this will be done in part 2 while scraping user information from github.

Summary Statistics and Data Summary for Part 1:

- Sample Size of "Github Data" before removing any IDs: 746
- Number of unique IDs (excluding bonus ids): 680
- Number of unique, valid IDs: 526
- Number of invalid IDs/IDs with missing/invalid information: 154
- Number of unique bonus IDs scraped over 6 hours: 13

Here is a table of descriptive statistics for the two numerical variables: repository count and follower count and Member Since as an integer:

	Repo Count	Follower Count	Member Since
count	526.000000	526.000000	5.260000e+02
mean	144.551331	436.307985	2.010572e+07
std	524.701838	1271.325222	2.797537e+04
min	0.000000	0.000000	2.008012e+07
25%	15.250000	26.000000	2.008090e+07
50%	40.500000	72.000000	2.010041e+07
75%	107.750000	280.500000	2.012052e+07
max	8082.000000	12816.000000	2.021080e+07

From this table we can see that of the 526 valid user IDs, the mean number of followers for these users from 2022 was about 436 and the mean number of repositories was 144. The max number of repositories was 8082 and followers was 12816. From 'Member Since', we can see that the oldest account was created in January 2008 and the most recently created account was made in August, 2021. The average creation date of these accounts was around May, 2010.

Part 2: Download Github User Data from Github.com:

The data from Github.com is stored in github_data.csv. This does not include user information from the bonus IDs since there was no information from 2022 to compare to and I did not want this user data to only be included in data pulled from Github.com when comparing the data from 2022 and 2023.

However, if one wanted to pull the github data for the bonus user IDs and the static IDs, one could run 'run_github_requests.py' using "parsed_files/all_ids.csv" to create json files for ALL users that would be parsed in parse_github.py with no other changes. I chose to only run requests for user ids from the static list so I would not bias the comparison of the 2022 and 2023 data since the bonus IDs did not have data listed on the Charcoalpapers dataset.

The Github.com user data is stored in 'github_data.csv' which is a csv file with columns for:

- User id ('ghid')
- Avatar url ('avatar_url')
- Url
- Number of followers ('num_followers')
- Number of following ('num_following')
- User's full name ('full_name')
- Company
- Blog

- Location
- Email
- Hireable (as True = 1/False = 0)
- Bio
- Member since (‘start_time’, as integer yyyymmdd)
- Last update (‘update_time’, as integer yyyymmdd)
- Starred url
- Admin Status (‘admin’ as True = 1/False = 0)
- Number of starred repositories (‘num_starred’)
- Number of public repositories (‘num_repos’)

I chose to additionally download admin status, followers count and number of public repositories to see if these have any correlation with other data such as follower count or number of starred repositories.

Summary Statistics and Data Summary for Part 1:

- There is still 526 unique, valid IDs (not including bonus IDs for the sake of comparison)

Here is a table of descriptive statistics:

	num_followers	num_following	num_starred	hireable	admin
count	526.000000	526.000000	526.000000	526.000000	526.000000
mean	423.969582	3632.728137	24.446768	0.378327	0.007605
std	1271.038777	19174.203583	10.509498	0.485431	0.086955
min	0.000000	0.000000	0.000000	0.000000	0.000000
25%	21.000000	22.000000	27.250000	0.000000	0.000000
50%	67.500000	87.500000	30.000000	0.000000	0.000000
75%	256.000000	324.000000	30.000000	1.000000	0.000000
max	13952.000000	302922.000000	30.000000	1.000000	1.000000

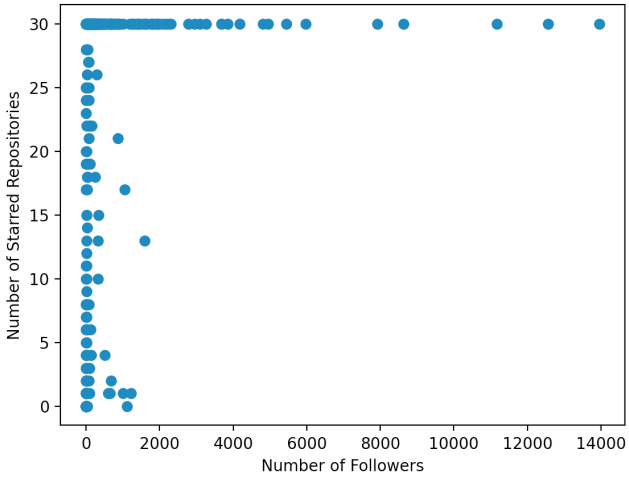
	start_time	update_time	num_starred	num_repos
count	5.260000e+02	5.260000e+02	526.000000	526.000000
mean	2.010798e+07	2.022530e+07	24.446768	181.157795
std	3.061442e+04	1.533614e+04	10.509498	956.987241
min	2.008013e+07	2.015041e+07	0.000000	0.000000
25%	2.008100e+07	2.023052e+07	27.250000	14.000000
50%	2.010053e+07	2.023092e+07	30.000000	40.000000
75%	2.012082e+07	2.023102e+07	30.000000	111.500000
max	2.023092e+07	2.023103e+07	30.000000	18264.000000

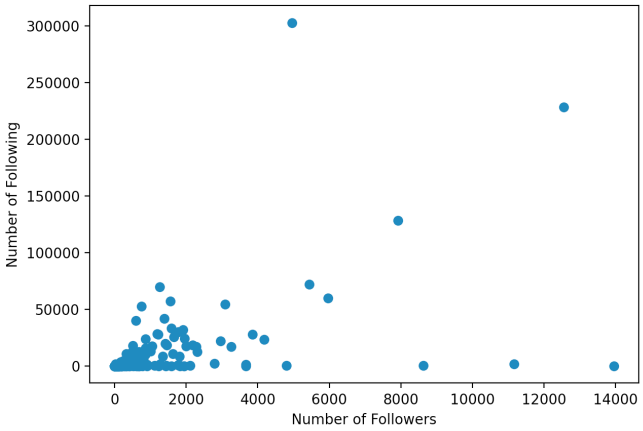
It is interesting to see that about 39% of our users are listed as hireable and less than 1% are admin. Because so few users were admin, I did not end up using this metric in my scatterplots.

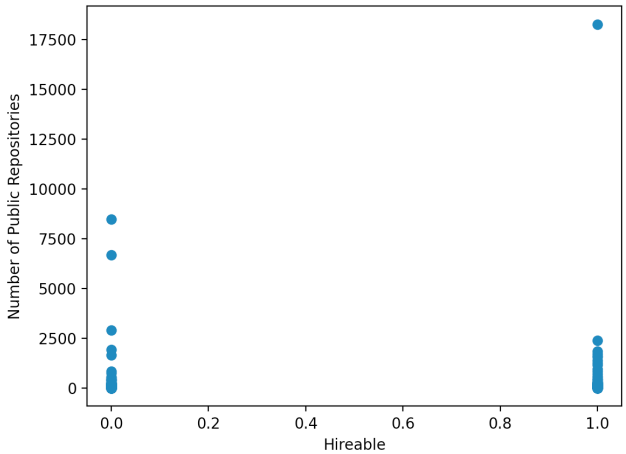
Some comparisons of the 2022 to 2023 data:

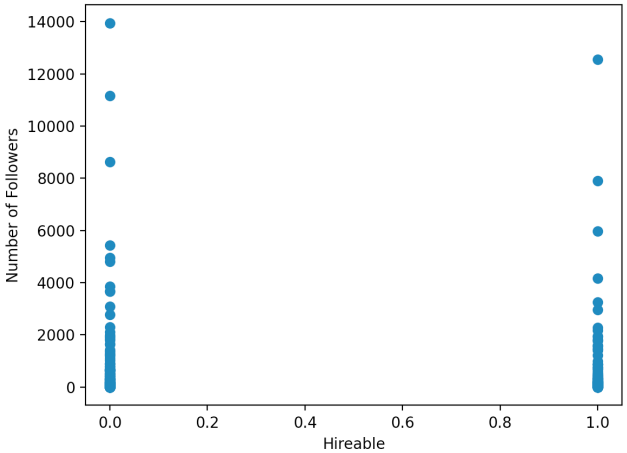
- To compare to the 2022 data, the current data for our set of 526 users has a new mean public repository count of 181 compared to just 144 in 2022. This mean on average our users have created about 40 more repositories since the prior data was collected.
- The new average follower count is only 424 compared to the mean in 2022 which was 436.
- A note on the number of starred data: using either an api request of the starred_url for each user or scraping each users individual repository page only returned at most 30 starred repositories. I could not figure out how to loop through multiple pages in a user's repository page to count all starred repositories. This means the max value for my num_starred variable is 30. However this variable is accurate for users with 30 or fewer starred repositories.

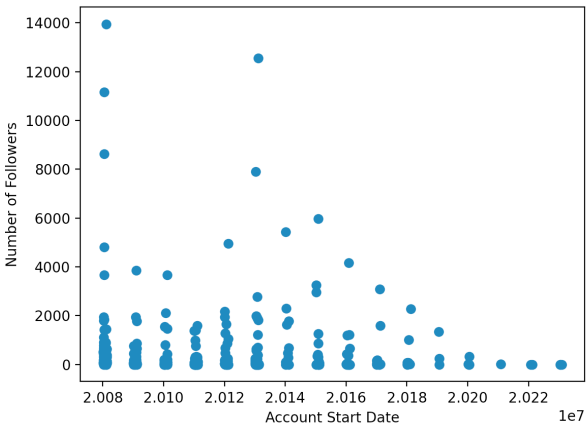
Part 3: Visualize and Describe the Data using Scatterplots:

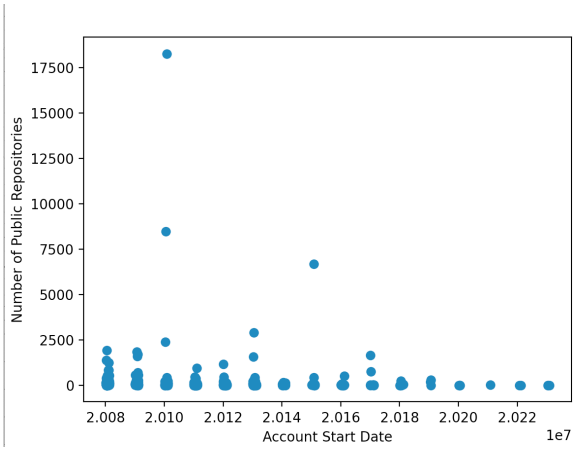












Hypotheses and Observations:

- I expected the number of following and followers to have a positive correlation which can be seen in the second scatterplot.
- I expected the number of followers to be positively correlated with the number of starred repositories. Although my data for number of starred is skewed on the top, it can clearly be seen that accounts with greater number of followers also have more starred repositories/are more likely to have greater than 30 starred repositories.
- I also thought plotting some variables against the dummy variable for “hireable” would be interesting. I chose to plot number of public repositories and number of followers. It is hard to tell without a jitter on the data but there does appear to be a positive correlation between hireable = 1 and number of public repositories. I would expect this because someone who lists themselves as hireable on github is likely using their account in their resume/job search. This means they likely have published repositories on their account that they would want to show their potential employers. There appears to be a much weaker relationship between hireable and follower count. This makes sense because there is not necessarily an incentive to have many followers just because someone is looking for a job.
- My favorite plots are the last two against the account creation date. There is a strong negative correlation between account creation date and number of followers. This is to be expected because a new account has had less time to gain followers. However, I am a little surprised that the correlation between account start date and number of public repositories is not stronger. There is definitely a negative relationship but it does not seem as pronounced as was with number of followers. It would make sense for the number of public repositories to increase as an account gets older.