# Project 2 - ETL - OKCUPID

Tommy Quach

Martha Calderon

Ugoada Ofoezie

# Objective

Our objective was to scrape OkCupid personal essays by gender to see if there are any common words used by men and women.

# Program Outline

The program is designed to first login, then go to the “match” page which contained urls to profiles that we wanted to scrape data from. After collecting these urls, we then visit these individual pages one at a time. The program then scrapes the relevant data, and then heads to the next url for further scraping.

The number of urls available to us during the visit to the match page is 20, unless we scroll down, at which point more urls are presented to us. The functionality of splinter does not allow us to scroll down. To work around this limitation, we used the refresh functionality of splinter, at which point, we can collect another batch of urls.

This was very time consuming, opening the individual profiles one by one. Compounded to this delay is the fact that in order to properly pull the html, we had to set a “sleep” functionality so that the page can properly load. Finding a balance between waiting long enough so that page would properly load and not waiting so long so that the program takes forever to run. This time is also dependent on the machine/internet connection running the program.

After scraping the data, the information is stored in the database. If you do not empty the collection, then information is continually added to the collection.

Once we have populated the collection, we can perform the analysis in Pandas.

# Source Code

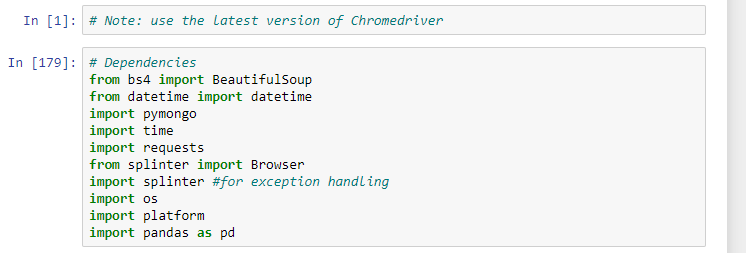


Figure 1: Contains our Dependencies. Note, this program requires the latest version of Chromedriver, otherwise, button clicks and filling in of data forms will not work

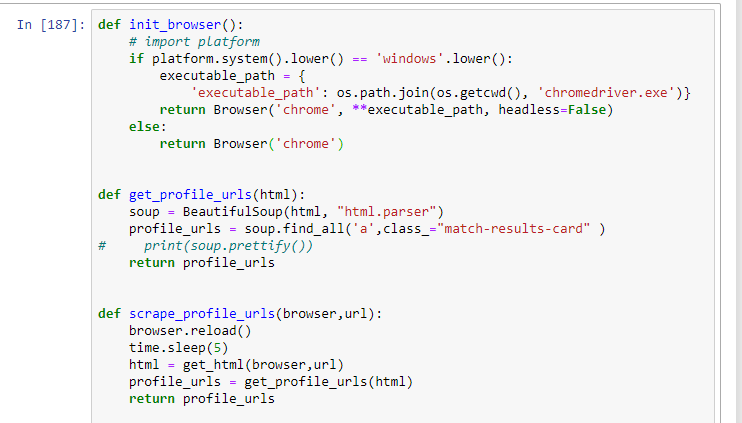


Figure 2: The init\_browser function initializes the browser, and works regardless if it’s a Windows or Mac. The scrape\_profile\_urls function takes the url of the “match” page and then returns all of the profile urls on the page, using the html of that page.



Figure 3: The login function allows the user to login. We had two profiles to grab male and female data. The get\_essays function allows you to grab the essays (the content that people manually wrote on their profiles).



Figure 4: The scrape\_essays\_and\_gender function is what pulls the pertinent information from the webpage.

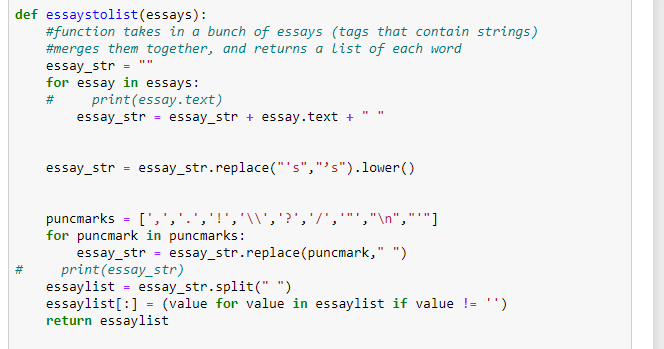


Figure 5: The essaystolist function takes in the essays, can concatenates all the essays into one string.

It then performing various string operations, such as removing punctuation marks, and making everything lowercase. The function then splits the string by the space character and returns a list to be entered into the database.



Figure 6: The scrape\_many function does several things. First, it controls the number of times the page is refreshed. Each time the page is refreshed, the okcupid website presents to us another set of profile urls. Inside the for loop, we visit each url, scrape what we want, and insert the words into our database. We also store the gender and profile url for future analysis.



Figure 7: The main\_login() function was separated from the main function because of difficulties logging in. This may have been a result of the program running to quickly and causing a weird login page to pop up.

Once the profiles are stored inside the database, we can perform the analysis in Pandas.

# Results

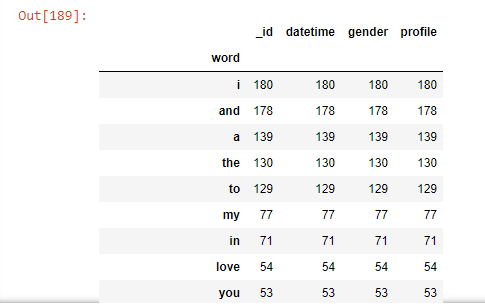


Figure 8: Above is a sample of the words found in the database after 5 refreshes for females looking for males. Unsurprisingly, the most common words are “i, and, a”. Going through the table though, words that of some interest include:

* Travel
* Movie
* Cooking
* Laugh
* World
* Fun
* Music
* Netflix
* Hamilton
* Mechanical
* Trivia
* Exploring
* Alabama

In the males, some of the common words included

* fun
* entertaining
* Chill
* Special
* cute
* Brooklyn
* Insta
* Adventurous
* experiences

Based on our dataset, the words special, fun and chill were in common, experiences, brooklyn, but “entertaining” and surprisingly “cute” was not in the female dataset.

# Limitations and Considerations

Our profiles only considered the viewpoint from a male and female in the age of 18-30, and within the NYC area. Our initial profile creation also limited us in what we can see, although we tried to be as inclusive as possible. We also could not let this program run indefinitely, as okcupid will definitely ban the account if you scrape too much.