Assignment 03 - APIs and Text Mining

Due Date

March 22 (parts 1 and 2), 2024 by 11:59pm. The learning objectives are to collect data from an API and perform text mining.

APIs (Part 1)

The aim is to get more practice working with APIs. We will use and API maintained by NASA for retrieving information about near earth objects (NEOs), asteroids that pass close to Earth. The documentation is available at https://api.nasa.gov/ (scroll down to the API titled Asteroids NeoWs). First you will need an API key for accessing the service. You can get one at https://api.nasa.gov/ where you will need to supply an email address, which can be either your UofT email or a personal email address.

1. Use `start_date` and `end_date` along with the API key to submit an HTTP GET request to the NASA NeoWs Feed API. Pick a range of a few days.
2. There should be 3 elements in your retrieved query: a list of urls, the number of near earth objects that had their nearest approach during the time spanned by `start_date` and `end_date`, and an object whose attributes are the dates (represented by strings of the form YYYY-MM-DD) in the time spanned by `start_date` and `end_date`. Each such date attribute has an array of values, each of which represents a near Earth object. Summarize how many near earth objects you retrieved and some of the parameters that are available for them.
3. Extract information on `estimated_diameter`, `is_potentially_hazardous_asteroid`, and `relative_velocity`, and create a table with this information for all of the NEOs pulled in your date range.
4. Explore is how the number of near Earth objects per day changes from day to day. Is this number correlated from one day to the next?
5. Explore associations between the three variables extracted in 3.

Text Mining (Part 2)

The Consumer Financial Protection Bureau maintains a database of customer complaints. We will use text mining to see if there are any trends in the reported consumer's narrative description of the issue/complaint. The data can be acquired here: https://www.consumerfinance.gov/data-research/consumer-complaints/.

1. Using data.table (the file is very large), subset the data to include only the last 2 years (use Date received). Summarize the dataset dimensions and variables. Bonus points if you use the API to acquire the data!
2. Tokenize the complaints (consumer complaint narrative) and count the number of each token. Do you see anything interesting? Does removing stop words change what tokens appear as the most frequent? What are the 5 most common tokens for the complaints after removing stopwords?
3. Tokenize the complaints into bigrams. Find the 10 most common bigrams and visualize them with ggplot2.
4. Calculate the TF-IDF value for the complaints in the top 3 issues. Here, the issue is like the document. What are the 5 tokens from each issue with the highest TF-IDF value? How are the results different from the answers you got in question 1?