POL 574, Quantitative Analysis IV
Prof. Arthur Spirling
Assignment date: February 14, 2025

Homework 1

This homework must be turned in on Canvas by **2pm on Friday, February 28, 2025**. Late work will incur penalties of the equivalent of one third of a letter grade per day late.

It must be your own work, and your own work only—you must not copy anyone's work, or allow anyone to copy yours. This extends to writing code. You may consult with others, but when you write up, you must do so alone.

Your homework submission must be a PDF or HTML report, containing all written answers and code, generated from `RMarkdown`. **Raw `.R` or `.Rmd` files will not be accepted.**

Please remember the following:

- Each question part should be clearly labeled in your submission.
- Do not include written answers as code comments. We will not grade code comments.
- The code used to obtain the answer for each question part should accompany the written answer.
- **Your code must be included in full, such that your understanding of the problems can be assessed.**

There are plenty of resources to get started with `RMarkdown` online, see for instance here. You can also use the `template.Rmd` template in our GitHub repo to get started. Using this template is not required.

---

1. Let's begin by loading in data on United Nations General Assembly speeches.[1] Create a corpus of speeches by the United States, the United Kingdom, and Australia for the year 2017. For each country, use only the first speech in the data for 2017.

   (a) Calculate the type-token ratio and Guiraud's index of lexical richness for each of these speeches and report your findings.

   (b) Create a document feature matrix of the speeches, with no preprocessing other than to remove the punctuation–be sure to check the options on "dfm" and "tokens" in R as appropriate. Calculate the cosine similarity between the documents with `quanteda`. Report your findings.

[1] Data from Turco, Linnea R. "Speaking Volumes: Introducing the UNGA Speech Corpus." *International Studies Quarterly* 68.1 (2024).

2. Consider different preprocessing choices you could make. For each of the following parts of this question, you have three tasks: (i) make a theoretical argument for how it should affect the TTR of each document and the similarity of the documents (ii) re-do question (1a) with the preprocessing option indicated and (iii) re-do question (1b) with the preprocessing option indicated.

   To be clear, you must repeat tasks (i-iii) for each preprocessing option below. You should remove punctuation in each step.

   (a) Stem the words

   (b) Remove stop words

   (c) Convert all words to lowercase

   (d) Does tf-idf weighting make sense here? Calculate it and explain why or why not.

3. Recreate the tf-idf table from Table 2, Column 1 of Ballandonne and Cersosimo (2023)[2] for Adam Smith's *The Theory of Moral Sentiments*. You can find the specific table (and a discussion of the authors' process) in the lecture notes from Week 1. See if you can recreate their top tf-idf terms:

   (a) Load the texts as a corpus

   (b) Tokenize each text as a single document, following the preprocessing of the authors.

   (c) Generate a tf-idf document-feature matrix, following the approach of the authors. Report the top 10 features for *The Theory of Moral Sentiments*. Are they in agreement with Ballandonne and Cersosimo?

   ** Don't stress if you can't get each term to align exactly. Just do your best to follow the process of the authors.*

4. Take the following two headlines:

   ```
   "Trump's immigration crackdown sparks humanitarian crisis at the US border"

   "Trump's immigration reforms strengthen US national security and US economy"
   ```

   (a) Create a DFM of the two sentences. Make sure to remove punctuation and convert the sentences to lower case.

[2] Ballandonne, Matthieu, and Igor Cersosimo. ”Towards a “Text as Data” Approach in the History of Economics: An Application to Adam Smith's Classics.” *Journal of the History of Economic Thought* 45.1 (2023): 27-49.

(b) Calculate the Euclidean distance between these sentences **by hand—that is, you can use base R, but you can't use distance functions from** `quanteda` **or similar.** Report your findings.

(c) Calculate the Manhattan distance between these sentences by hand. Report your findings.

(d) Calculate the Jaccard similarity between these sentences by hand. Report your findings.

(e) Calculate the cosine similarity between these sentences by hand. Report your findings.

(f) Consider two sentences, one from American English, one from British English.

```
At the theatre, my neighbour wore her favourite jewellery
At the theater, my neighbor wore her favorite jewelry
```

What's the Levenshtein distance between these two? Show your working. Now, using `stringdist::stringdist()`, measure the distance between them using method = "dl" (using otherwise standard defaults). Is the distance larger or smaller? Why? That is, what does Damerau-Levenshtein distance allow you to do that Levenshtein distance does not? (hint: what does the `t=` part of the weight correspond to?)

5. For this question we will use the UK Political Party Manifestos data from `quanteda` (data corpus `data_corpus_ukmanifestos`).

   (a) First, extract and concatenate the entire text of the corpus, remove punctuation and set all characters to lower case. Use this text to produce a contingency table for the collocation "Northern Ireland" *[Hint: Regular expressions with look-aheads and look-behinds are useful here]*. Calculate the expected frequency of "Northern Ireland" under independence. Compare the observed and expected frequency. Based on this comparison, is "Northern Ireland" a meaningful multi-word expression in this corpus?

   (b) Now use `quanteda`'s `textstat_collocation` to inspect the same 2-gram "Northern Ireland". Report the λ and z values. How do these results relate to your conclusions in question 5(a)?

   (c) Finally, use `textstat_collocation` to inspect all 2-grams with `min_count = 5`. Report the 10 collocations with the largest λ value. Report the 10 collocations with the largest count. Discuss which set of n-grams are likely to be multi-word expressions.

6. Using *The Theory of Moral Sentiments*, make a graph demonstrating Zipf's Law (in log-log space). Lower case everything, and drop punctuation to do so. What are the five most common terms in the corpus?

7. Find the value of b that best fits *The Theory of Moral Sentiments* for Heaps' Law, fixing $k = 44$. Remove punctuation and lower case everything, as previously. Repeat this exercise for the situation where you don't lower case everything. How does b change? Why?

8. Let's focus on the UNGA speech data again. Pick two countries and consider the context in which speakers from these countries talk about *security* and *freedom*. Use `quanteda`'s `kwic` function to discuss the different context in which those words are used by the different countries.

9. Consider the bootstrapping of the texts we used to calculate the standard errors of the Flesch reading scores of Irish budget speeches in Precept 3.

   (a) Obtain all UNGA speeches given by the US from 2005-2015. Generate estimates of the FRE scores of these speeches over time (i.e. per year), using sentence-level bootstraps instead of the speech-level bootstraps used in lab. Include a graph of these estimates.

   (b) Report the means of the bootstrapped results and the means observed in the data. Discuss the contrast.

   (c) For the empirical values of each text, calculate the FRE score and the Gunning's-Fog Index score. Report the FRE and Gunning's-Fog Index scores and the correlation between them.

   Hint: After you split up each speech into sentences, some of the sentences will not be "sentences" at all (e.g. headings). Regular expressions are one way to remove this kind of text.

10. The US Declaration of Independence was not produced in one go. It was drafted and edited. The relevant versions are called `jefferson_draft` and `final_version` respectively.

    (a) Using `text.alignment::smith_waterman()` and the standard scoring defaults therein, report the local alignment score for the texts. Perform your analysis at the `words` (not `characters`) level.

    (b) Repeat the analysis, but be more aggressive about matches specifically. That is, increase the `match` score to 3, and reduce the `mismatch` penalty to zero. What is the local alignment score now? Does it imply more, or less, of the text overlaps? Why—that is, what is the mechanism for this change?

11. This question is about Benford's Law, and uses the `countypres_2000-2020.csv` data set.

    (a) Pull the county level data for 2020 in Pennsylvania. Replicate Groharing & McCune (2022) Figure 1 (the barplot, as shown in lecture). That is, give the frequency for the Democrats and Republicans of each leading digit in their county voting returns for PA. Also add what we would expect under Benford's Law for the first digit distribution.

(b) Produce a similar graphic for Texas in 2020.

(c) Produce a similar graphic for Vermont in 2020.

(d) This article "Fact check: Deviation from Benford's Law does not prove election fraud" suggests one should be quite skeptical about claims of fraud based on these plots. What is the problem for states such as Vermont? To answer this question, compare the (log) distribution of county vote totals in TX and VT, and what that might mean for the plots you produced above.