DS-GA 1015, Text as Data
Prof. Andy Halterman
Assignment date: April 14, 2021

Homework 3

This homework must be turned in on NYU Brightspace by Thursday, **April 28, 2022, at 3pm**. Late work may be turned in up to **one** day late and will incur penalties of the equivalent of one half of a numeric point (e.g. $4.5 \rightarrow 4$).

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.**

Please consult "RMarkdown Basics" on the course GitHub for help with RMarkdown. You can also use the "Sample RMarkdown HW" template on the course GitHub to get started. Using this template is not required.

---

1. **LDA concepts**

   During lab, you've gone over the code that you need in order to fit LDA topic models. Both R and Python have a rich set of tools and tutorials for fitting topic models. In this problem, we'll instead focus on the concepts underlying topic models, specifically latent Dirichlet allocation, and the decisions you might make in fitting them.

   (a) Your boss has asked you to fit an LDA model on some proprietary text in your organization. The corpus of text is very large and he needs the model as soon as possible. Your LDA package gives you three options for fitting LDA: EM, variational inference, and Gibbs sampling. Which should you select, and why? (1-2 sentences)

   (b) Vanilla LDA requires you to specify k, the number of topics. Describe three approaches you could take to selecting the number of topics for your model. (3-4 sentences)

(c) Poking around in your organization's Github, you find two LDA models that were fit by a previous employee.

i. You examine the document-topic distribution (θ_i) for a few documents ($i = 1...N$) produced by Model 1, and find that the topics are very sparse, i.e., each document's topic distribution is highly concentrated in one or two topics, with little to no probability mass on other topics. Which hyperparameter is the likely cause of this and why? (1-2 sentences)

ii. You examine Model 2 and find that the topic-word distribution (β_k) is very flat (i.e., each topic assigns a similar probability to each word within the topic). Which hyperparameter is the likely cause of this and why? (1-2 sentences)

(d) Your boss's boss examines the output of your model and has a concern. She has a PhD in political science and insists on having full uncertainty estimates around all parameters in the model. She asks you to refit the model, telling you that runtime is not a concern. Which method for fitting do you select now, and why? (1-2 sentences)

(e) Finally, you have a model you're satisfied with. You present it to your group and someone asks you, "after we condition on the topic θ_i, are the words in the document independent of each other?" You pull up your slide with the LDA plate diagram and write the conditional probability of a word $p(w|????) =????$, answering your colleague's question. What is your answer and what expression do you write? (1 sentence with answer and explanation, plus one mathematical expression to justify it)

2. **STM: Topic Models with covariates:**

Note: Because topic models take a long time to run, you may save your fitted model objects or RStudio workspace to a .RData file after running the appropriate code to fit the models. When doing your final write-up, you may comment out the code that trains the model(s) and load the saved models from file. Please make sure the code to fit the models is clearly marked.

The Structural Topic Model (STM) is designed to incorporate document-level variables into a standard topic model. Since, we have information about both the vaccine they are discussing and the date of the articles, we can use an STM (from the `stm` package) to model the effects of these covariates directly.

For this exercise you will use a database of tweets about COVID19 vaccines used (https://www.kaggle.com/gpreda/all-covid19-vaccines-tweets). A sample of the data is available in the file `vaccinationtweets.csv`.

(a) Make sure that you have a variable that records the date the tweet was sent, then create a subset of `vaccinationtweets` that (1) only contains tweets from 2021-01-01 to 2021-04-30 (2) and only contains tweets that include hashtags with the names of the vaccines: "PfizerBioNTech" and "Covaxin" (use the contents of the variable `hashtags` to subset your data). Create a plot that shows how many tweets were associated with each vaccine on each day.

(b) Preprocessing decisions can have substantive impacts on the topics created by topic model algorithms. Make a brief argument for or against removing rare terms from a dfm on which you plan to fit a topic model. (2-4 sentences)

(c) Use what other preprocessing you believe to be appropriate for a topic modeling problem like this one? Discuss your preprocessing choices and apply them to the tweet corpus. (1-4 sentences)

(d) Construct a numeric binary variable from the "hashtags" variable in the restricted sample. Fit an STM model where the topic content varies according to this binary variable, and where the prevalence varies according to both this binary variable and the spline of the date variable that you've created earlier. Pick a value for k, which should be greater than 5. Keep in mind that this function is computationally demanding, so start with the minimum threshold document frequency threshold set to 10; if your computer takes an unreasonably long time to fit the STM model with this threshold, you can raise it to as high as 30.
Report the number of iterations completed before the model converged.

(e) Identify and name each of the 5 topics that occur in the highest proportion of documents using the following code:[1] `plot(stm_fit, type = "summary")`

(f) Next, refit your STM model using more topics, $k' > k$. Repeat the plotting exercise above. Are your topics substantively different? Which do you prefer? (1-3 sentences plus plot)

(g) Using the visualization commands in the `stm` package, discuss one of these top 5 topics. How does the content vary with the type of vaccine (the binary variable created) discussing that topic? How does the prevalence change over time? (2-4 sentences)

3. **Applications of Word Embeddings:**

(a) One of the limitations of dictionary-based methods is that dictionaries rarely include an exhaustive list of terms. Propose and justify a method for improving a dictionary using pretrained word embeddings. (2-3 sentences)

(b) Now imagine that you've implemented your method from the previous question and compared it to a standard dictionary-based method. Which accuracy measure (accuracy, F1, precision, recall...) would best indicate whether you've overcome the non-exhaustive dictionary problem? (1-2 sentences)

(c) In many models that we apply to bag-of-words representations, the size of the vocabulary is the greatest determinant of model size and runtime. Much of the early class covered ways to reduce the dimensionality of our text through stopword removal, stemming, lowercasing, etc. Describe and justify a procedure for using word embeddings to reduce the vocabulary size of a bag-of-words document representation. (N.B.: we still want our model to operate on a (reduced) bag of words, *not* one directly on word embeddings). (2-3 sentences)

(d) Your impatient boss from problem 1 has heard about word embeddings and wants you to use them as features in a supervised model to label customer emails with sentiment

[1] `stm_fit` represents the output of the STM model you fit in the preceding question.

scores. Should you train new embeddings from scratch or use existing, off-the-shelf embeddings? What are the factors in your decision? (2-3 sentences).

(e) Finally, briefly discuss the potential ethical issues of using pretrained word embeddings in your classifier. (2-3 sentences)