MA710 Assignment 3: Text Mining

Dr. David Oury

21 Mar 2017

The third assignment is a report in which you investigate New York Times articles, by applying text mining techniques. Your goal is to group these articles into a collection of topics. Code for the analysis is supplied in `20170321_TextMining_analysis.R`. Run all commands in `20170321_TextMining_functions.R` to define the functions used in the analysis file. The procedure is as follows:

1. Create a developer account at http://developer.nytimes.com and obtain an API key for the *Article Search API*. Store this key in variable `articlesearch.key`. See line 20 of the `20170321_TextMining_analysis.R` script.
2. Retrieve a set of articles from the New York Times article database using the `get.nyt.articles` function. *Choose* a query term and date range in your search. Modify these parameters until you find a satisfactory collection of articles. Use the `get.nyt.hits` function to determine the number of articles that match your search. You are limited to 1,000 API calls per day. One API call is required for every *page* of ten articles retreived.
3. *Choose* one of these four fields to analyze: `headline`, `snippet`, `lead_paragraph` or `abstract`. You will create a character vector from this field that I'll refer to below as "documents" or "document vector".
4. Clean the documents using the `clean.documents` function. This will remove numbers and punctuation from the documents. You may *choose* to modify this function.
5. Using the `modify.words` function, *choose*
   - whether to stem words (logical parameter `stem.words`)
   - what size n-grams to create (vector parameter `ngram.vector`)
   - which stop words to remove (vector parameter `stop.words`)
6. Create a document matrix using the `create_matrix` function. *Choose*:
   1. Binary weighting - use `tm::weightTfIdf` and create a "binary matrix"
   2. Term frequency weighting - use `tm::weightTf`, but do not create a "binary matrix"
   3. Term frequency-inverse document frequency weighting - use `tm::weightTfIdf`
7. Create a document-term matrix. You may *choose* to modify this matrix in two ways:
   1. Create a binary matrix (see binary weighting above)
   2. Keep only frequencies which meet a given threshold (see `reduce.dtm` function)
8. Create cluster groups after you *choose* `k`, which is the number of clusters to find.
9. Evaluate the cluster groups you have created with functions `table`, `check.clusters` and `view.cluster`. You may also consider using the `cluster.stats` function
10. Vary the options and parameters above and repeat these steps until you have found a good cluster group where as many as possible of the clusters consist of articles with a common topic. **Do not ask me how many times to repeat these steps.**

In the previous list the word "*choose*" indicates code or a parameters to modify.

To find "good" clusters modify the code and these parameters, and then evaluate the clusters and cluster groups you created.

Each code paragraph in `20170321_TextMining_analysis.R` is marked either:

- `OPTION`, which indicates that there is a choice to make when running this code (unless it is says otherwise) and that this code must be run (unless it is commented)
- `EVALUATE`, which indicates code you will use to evaluate your clusters
- `Check`, which indicates code you can use in checking your documents

Make your choices that determine what you modify and which options you use to create your cluster groups based on:

- Evaluations of the cluster groups. Can you identify topics for most clusters? Do all or most of the documents of a cluster seem to belong to that cluster?
- Error free completion of the code in a reasonable amount of time. Your code must complete in a reasonable amount of time and without errors to be useful.
- Your subjective assessment of the parameters and their effects. The number of parameter combinations is very large, certainly too large to examine the effects of each. You must choose which parameters to use and have reasons for these choices.

In summary, you have several objectives:

1. To find a set of options which create a cluster group which collects into clusters those documents with similar topics
2. To describe the options used to create the final cluster group
3. To describe the topics in the final clusters

Use `20170321_Assignment_3_template.Rmd` as a template for your report.

The final report is due on 6 Apr 2017 and should be emailed as a **PDF** to doury@bentley.edu in reply to an email I will send to you. The name of the file should be `A3-[shortname-list].pdf`. where `[shortname-list]` should be replaced by the list of Bentley shortnames for the members of your group where you place a dash "-" between each shortname. For instance, `A3-bobama-doury.pdf`.