Exploring the text distribution of Nietzsche's *Antichrist*

G. Palafox

September 22, 2020

Abstract

An elementary text analysis of Nietzsche's *The Antichrist* [3] is performed. The distribution of some of the obtained data is estimated.

1 Introduction

A text analysis of Nietzsche's *The Antichrist* [3] is performed. We find the frequency of common words, and explore the lengths of sentences in the text. We try to find the distribution of this data, or simulate it if we cannot find a fitting common distribution. Additionally, we compare a network representation of the book with a random network having the same degree distribution.

2 Text analysis and distributions

The text extraction and analysis is performed in a Jupyter [1] notebook running R [4] version 4.0.0[1]. The book is downloaded directly from Project Gutenberg's site using R's `gutenbergr` library. The book downloaded starts with an introduction by the translator, which we omitted from the analysis, as the intention was to study the author's words.

First we obtain two elements of the text: the length of its sentences, and the frequency of the words appearing in it. We discard so called stop-words, and keep only the most frequent words (those appearing over eight times). We create a histogram of the sentences' length, that can be seen in Figure 1a. From our knowledge of discrete distributions, we infer these data has a geometric distribution. We use R's function `fitdistr` from the `MASS` library to try and fit the data to a geometric distribution, obtaining a parameter $p = 0.36$. We generate a thousand pseudo-random numbers with geometric distribution and $p = 0.36$, and plot a histogram of these values, as seen in Figure 1b. Observing these two histograms side-by-side in Figure 1, we conclude the length of the sentences has a geometric distribution with $p = 0.36$.

In the case of the word frequency, which is plotted in Figure 2a, the function `fitdistr` does not give as good of a match. In this case, we pseudo-randomly select words from the text, aiming to obtain a distribution similar to the actual one. To do this, we do the following. We interpret the bar plot as a probability mass function (prior to normalizing the area to be one). Then, we partition the words in groups according to the frequency of which they occur. We calculate the area of each of these groups, relative to the total area, to partition get a partition $p_0 = 0, p_1, \ldots, p_8, p_9 = 1$ of the unit interval . Finally, we generate a pseudo-randomly generated number p, and select a word pseudo-randomly from group k if $p_{k-1} \leq p \leq p_k$. A bar plot showing the frequency of the randomly generated words is shown in Figure 2b. It can be compared to the actual frequency of words in the text in Figure 2. Additionally, Table 1 shows the most frequent words from both the text and our random selection, and Figure 3 shows a boxplot comparing these two data sets.

2.1 Network representation

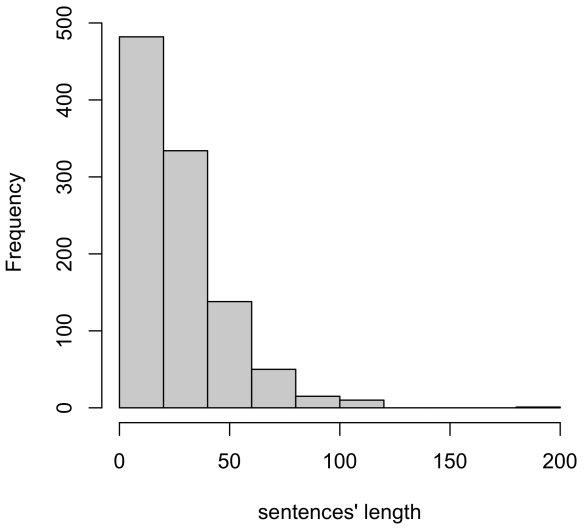
In a previous text analysis[2], a network representation of our book was made, as is seen in Figure 4a. Using R's library `igraph`, function `sample_degseq`, we create a random network which has the same degrees as our book network. This is done based on the configuration model [2], discarding graphs with multiple-edges or self-loops. We cannot say much about how these two compare, but decided to show them here nonetheless. The networks are shown side by side in Figure 4.

3 Conclusion

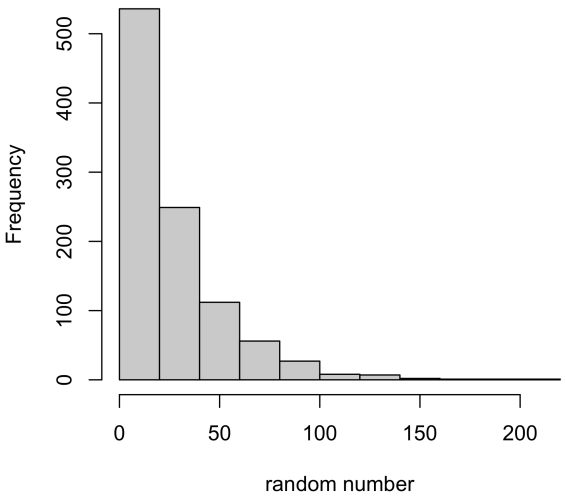
Overall, our exposition is very elementary. Further statistical analysis can be done to determine whether the geometric distribution really is a good fit for the sentences' length distribution. A deeper study of discrete probability distributions

[1] The script and a Jupyter [1] notebook detailing our analysis and graphics creation, can be found at `https://github.com/palafox794/AppliedProbabilityModels/tree/master/Assignment3`

[2] It can be found at `https://github.com/palafox794/AppliedProbabilityModels/tree/master/Assignment2`

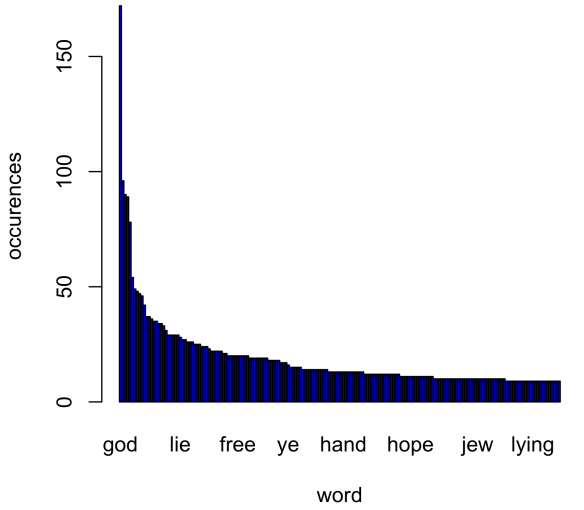


(a) Histogram of sentences' length.

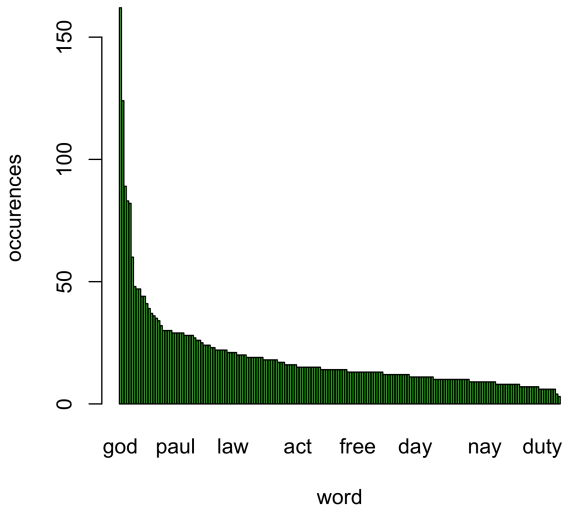


(b) Histogram of 1000 words with distribution Geo(0.36).

Figure 1: Sentences' length distribution.



(a) Word occurrence in the book.



(b) Word occurrence of words pseudo-randomly selected from the book.

Figure 2: Barplots of word occurrences.

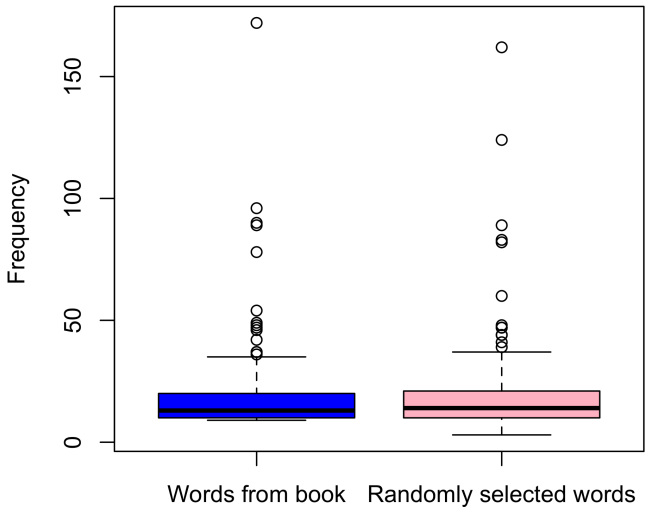


Figure 3: Comparison of word frequency.

(a) Book network. Words are vertices and edges joining words appear as a bigram in the text.

(b) Random network with same degree distribution as book network.

Figure 4: Network representation.

Table 1: Word frequency

(a) Frequency of words in the book

words	freq
god	172
life	96
christian	90
christianity	89
world	78
truth	54

(b) Frequency of pseudo-randomly selected words from book

rand_words	Freq
god	162
world	124
life	89
christian	83
christianity	82
sort	60

may help find a good fit for the distribution of how frequent words distribute in the text. More sophisticated network analysis techniques could be deployed, aiming to understand the differences and similarities of our book network and a random network with the same degree sequence.

References

[1] Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., et al. Jupyter notebooks—a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas: Proceedings of the 20th International Conference on Electronic Publishing* (2016), IOS Press, p. 87.

[2] Newman, M. *Networks*, vol. 1. Oxford University Press, Oct 2018.

[3] Nietzsche, F. *The Antichrist*. Project Gutenberg, Sept. 2006. `http://www.gutenberg.org/files/19322/19322.txt`.

[4] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020.