IN4080 2021, Mandatory assignment 1, part A

Your answer should be delivered in devilry.ifi.uio.no no later than Friday, 24 September at 23:59

Overview of the assignment

Mandatory assignment 1 has two parts

- Part A Working with texts and frequencies
- Part B Text classification using scikit-learn

You should answer both parts. It is possible to get 70 points on part A and 30 points on part B, 100 points altogether. You are required to get at least 60 points to pass. It is more important that you try to answer each question than that you get everything correct.

General requirements

- We assume that you have read and are familiar with IFI's requirements and guidelines for mandatory assignments
  - https://www.uio.no/english/studies/examinations/compulsory-activities/mn-ifi-mandatory.html
  - https://www.uio.no/english/studies/examinations/compulsory-activities/mn-ifi-guidelines.html
- This is an individual assignment. You should not deliver joint submissions.
- You may redeliver in Devilry before the deadline, but include all files in the last delivery. Only the last delivery will be read!
- If you deliver more than one file, put them into a zip-archive.
- Name your submission <username>_in4080_mandatory_1

Delivery

Your delivery can take one of two forms

Alternative A:

- Deliver the code files.
- In addition, deliver a separate pdf file containing results from the runs together with answers to the text questions.

Alternative B:

- A Jupyter notebook for each of the two parts where you answer the text questions in markup.
- In addition, deliver a pdf-version of the notebooks where all the results of the runs are included.

Whether you use the first or second alternative, make sure that the code runs on the IFI machines or on your own machine in the environment we provided, see **https://www.uio.no/studier/emner/matnat/ifi/IN4080/h22/lab-setup/**

If you use any additional packages, they should be included in your delivery.

Exercise 1 – Conditional frequency distributions (35%)

(Hint: Complete exercises 3 and 4 on the first exercise set before you start on this one.)
The NLTK book, chapter 2, has an example in section 2.1, in the paragraph Brown Corpus, where they compare the frequency of modal verbs across different genres. We will conduct a similar experiment, We are in particular interested in to which degree the different genres use the masculine pronouns (*he, him*) or the feminine pronouns (*she, her*).

a. Conduct a similar experiment as the one mentioned above with the genres: *news, religion, government, fiction, romance* as conditions, and occurrences of the words: *he, she, her, him*, as events. Make a table of the conditional frequencies and deliver code and table.
(Hint: Have you considered case folding?)

b. Answer in words what you see. How does gender vary with the genres?

Maybe not so surprisingly, the masculine forms are more frequent than the feminine forms across all genres. However, we also observe another pattern. The relative frequency of *her* compared to *she* seems higher than the relative frequency of *him* compared to *he*. We want to explore this further and make a hypothesis, which we can test.

Ha: The relative frequency of the objective form, *her*, of the feminine personal pronoun (*she* or *her*) is higher than the relative frequency of the objective form, *him*, of the masculine personal pronoun, (*he* or *him*).

c. First, consider the complete Brown corpus. Construct a conditional frequency distribution, which uses gender as condition, and for each gender counts the occurrences of nominative forms (*he, she*) and objective forms (*him, her*). Report the results in a two by two table. Then calculate the relative frequency of *her* from *she* or *her*, and compare to the relative frequency of *him* from *he* or *him*. Report the numbers. Submit table, numbers and code you used.

It is tempting to conclude from this that the objective form of the feminine pronoun is relatively more frequent than the objective form of the male pronoun. Beware, however, *her* is not only the feminine equivalent of *him*, but also of *his.* So what can we do? We could do a similar calculation as in point (b), comparing the relative frequency of *her* –not to the relative frequency of *him* –but compare *her + hers* to *him + his.* That might give relevant information, but it does not check the hypothesis, Ha.

d. What could work is to use a tagged corpus, which separates between the two forms of *her*, i.e, if the corpus tags *her* as a personal pronoun differently from *her* as a possessive pronoun (determiner). The tagged Brown corpus does that. Use this to count the occurrences of *she, he, her, him* as personal pronouns and *her, his, hers* as possessive pronouns. See NLTK book, Ch. 5, Sec. 2, for the tagged Brown corpus. Report in a two-ways table.

   You could solve this by using the originally tagged Brown corpus with the original tags.

Alternatively, you may use the corpus with the universal pos tags. Both use diferent tags for the two functions of *her*. The name of the tags are different, though.

e. We can now correct the numbers from point (b) above. How large percentage of the feminine personal pronoun occurs in nominative form and in objective form? What are the comparable percentages for the masculine personal pronoun?

f. Illustrate the numbers from (d) with a bar chart.

g. Write a short essay (200-300 words) where you discuss the consequences of your findings. Consider in particular, why do you think the masculine pronoun is more frequent than the feminine pronoun? If you find that the there is a different distribution between nominative and objective forms for the masculine and the feminine pronouns, why do you think that is the case? Do you see any consequences for the development of language technology in general, and for language technology derived from example texts in particular? Make use of what you know about the Brown corpus.

Exercise 2 – Downloading texts and Zipf's law (35%)

In this exercise, we will consider Zipf's law, which is explained in exercise 23 in NLTK chapter 2, and more thoroughly in the Wikipedia article: *Zipf's law*, which you are advised to read. We will use the text *Tom Sawyer*.

a. First, you need to get hold of the text. You can download it from project Gutenberg as explained in section 1 in chapter 3 in the NLTK book. You find it here:
https://www.gutenberg.org/files/74/74-0.txt

b. Then you have to do some clean up. The downloaded text contains a preamble and a long appendix about the Gutenberg project and copyrights that should be removed.

c. You can then extract the words. We are interested in the words used in the book and their distribution. We are, e.g. not interested in punctuation marks. Should you case fold the text? Explain the steps you take here and in point (b) above.

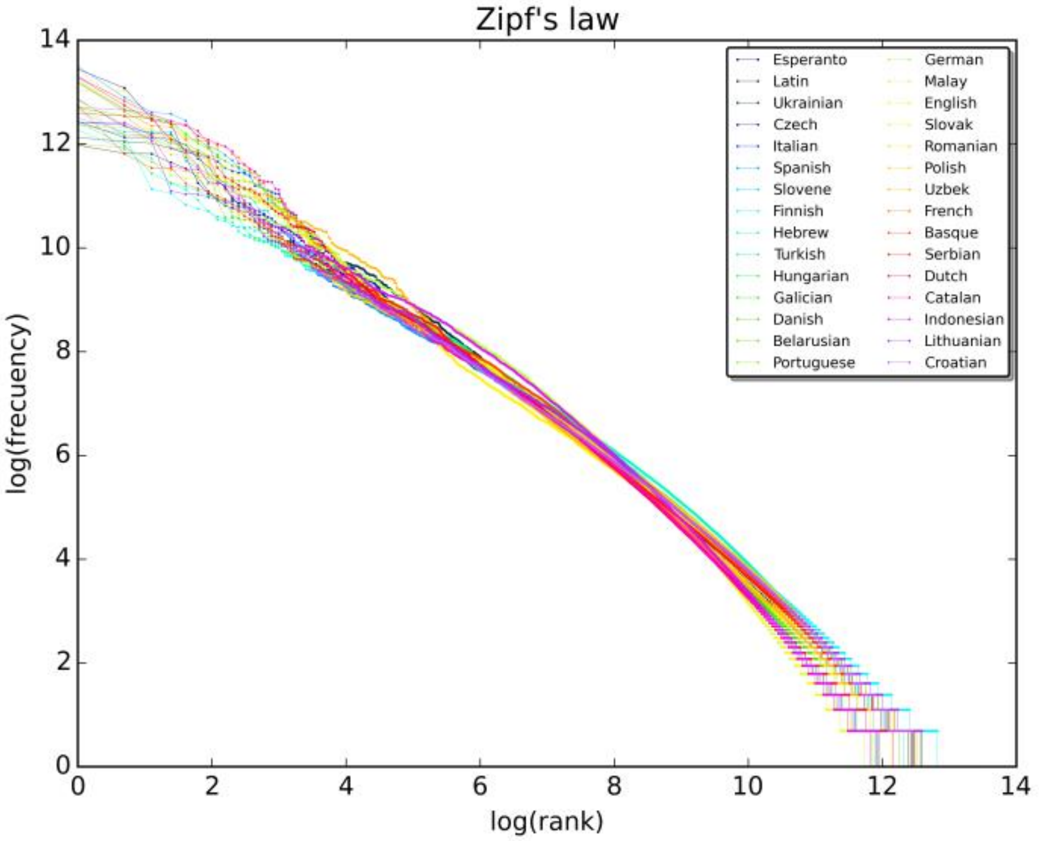
d. Use the nltk.FreqDist() to count the words. Report the 20 most frequent words in a table with their absolute frequencies.

e. Consider the frequencies of frequencies. How many words occur only 1 time? How many words occur n times, etc. for n = 1, 2, …, 10; how many words have between 11 and 50 occurrences; how many have 51-100 occurrences; and how many words have more than 100 occurrences? Report in a table!

f. We order the words by their frequencies, the most frequent word first. Let *r* be the frequency rank for each word and *n* its frequency. Hence, the most frequent word gets rank 1, the second most frequent word gets rank two, and so on. According to Zipf's law, r*n should be nearly constant. Calculate r*n for the 20 most frequent words and report in a table. How well does this fit Zipf's law? Answer in text.

g. Try to plot the rank against frequency for all words (not only the 20 most frequent ones). First, make a plot where you use linear scale on the axes. Then try to make a plot similarly to the Wikipedia figure below with logarithmic scales at both axes. Logarithms are available in numpy, using functions functions like *log()*, for the natural logarithm, and *log2()* for the base-2 logarithm. An alternative to using one of these functions is to explore *loglog()* plotting from *matplotlib.pyplot* directly.

Deliveries: Explanation of the steps you took in (b) and (c). The tables asked for in (d) and (e). The table in (f), the plots in (g) and answers to the question in (f) in words.



(source: Wikipedia)

End of part A, please proceed to part B