Part 3: Analyzing word and document frequency: tf-idf

- In the previous exercises we looked at *term frequency* (tf), meaning how frequently a word occurs in a document in a specific way.
- We removed stop words from our analysis, since they are highly frequent. But we can do better than that! We can actually keep the stop words, since they might be important as well.
    - We can then look at the *inverse document frequency* (idf), which decreases the weight for commonly used words and increases the weight for words that are not used very much in a collection of documents.
- This can be combined with term frequency to calculate a term's *tf-idf* (the two quantities multiplied together), *the frequency of a term adjusted for how rarely it is used*.

Term frequency in Austen’s novels I

- So, let’s try and find the important words Austen’s novels. We will use the statistic tf-idf to measure this. The inverse document frequency for any given term is defined as

$$idf(\text{term}) = \ln\left(\frac{n_{\text{documents}}}{n_{\text{documents containing term}}}\right)$$

- No worries, we have the skills to put this into code!

Term frequency in Austen’s novels II

Exercise 10

10a. Let’s start by looking at the novels of Austen and examine first term frequency, then tf-idf. We can start just by using dplyr verbs such as group_by() and join(). Can you fill in the blanks in the code below based on what you have learned so far and determine the most commonly used words in the novels? (Let’s also calculate the total words in each novel here, for later use)

```
library(dplyr)
library(janeaustenr)
library(tidytext)

book_words <- austen_books() %>%
  unnest_tokens(???,???) %>%
  count(book, word, sort = TRUE)

total_words <- book_words %>%
  group_by(???) %>%
  summarize(total = sum(n))

book_words <- left_join(book_words, total_words)
```

← Now have a look at your output. What strikes you in the data your tibble presents?

Term frequency in Austen’s novels III

10b. Now let’s plot the distribution of n/total = the number of times a word is used in a book/the total words in that book. Do you remember what package to call on to plot this distribution?

```
library(???)

ggplot(book_words, aes(n/total, fill = book)) +
  geom_histogram(show.legend = FALSE) +
  xlim(NA, 0.0009) +
  facet_wrap(~book, ncol = 2, scales = "free_y")
```

← Does the distribution of term frequency differ greatly per novel or is it quite similar? How would you interpret the plots?

Inverse document frequency in Austen's novels I

10c. Let's move on from term frequency to calculating tf-idf and attempt to find the words with high tf-idf (so high relative frequency). The bind_tf_idf() function in the tidytext package takes a tidy text dataset as input with one row per token (term), per document.

- The **word** column contains the terms/tokens
- The **book** column contains the documents
- The **n** column contains how many times each document contains each term

Based on those column headers, can you fill in the code below and calculate tf-idf?

```
book_tf_idf <- book_words %>%
  bind_tf_idf(???, ???, ???)

book_tf_idf %>%
  select(-total) %>%
  arrange(desc(tf_idf))
```

← What type of words have a high tf-idf score?

Inverse document frequency in Austen’s novels II

10d. Since everything in life is better with graphics, let’s visualize our tf-ifd findings! Run the code below to plot the highest tf-idf words in each of Austen’s novels. Can you make it so that you plot the scores per novel? And can you make sure that we see tf-idf for the tokens/terms we have been analyzing? Have a go…

```
library(forcats)

book_tf_idf %>%
  group_by(???) %>%
  slice_max(tf_idf, n = 15) %>%
  ungroup() %>%
  ggplot(aes(tf_idf, fct_reorder(???, tf_idf), fill = book)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~book, ncol = 2, scales = "free") +
  labs(x = "tf-idf", y = NULL)
```

← You have calculated and visualized what type of words distinguish Austen’s novels from each other. Another slam dunk! Time for one more? We have consistently looked at unigrams (single words) in our analyses so far, but now it’s time for a change…