Audit Access Expires Mar 19, 2020
You lose all access to this course, including your progress, on Mar 19, 2020.

Assessment Part 2: Dates, Times, and Text Mining

In this part of the assessment, you will walk through a basic text mining and sentiment analysis task.

Project Gutenberg is a digital archive of public domain books. The R package **gutenbergr** facilitates the importation of these texts into R. We will combine this with the **tidyverse** and **tidytext** libraries to practice text mining.

Use these libraries and options:

```
library(tidyverse)
library(gutenbergr)
library(tidytext)
options(digits = 3)
```

You can see the books and documents available in **gutenbergr** like this:

```
gutenberg_metadata
```

Question 6

0/1 point (graded)

Use `str_detect` to find the ID of the novel *Pride and Prejudice*.

How many different ID numbers are returned?

0

✖ **Answer:** 6

\(\)

Answer code

```
gutenberg_metadata %>%
    filter(str_detect(title, "Pride and Prejudice"))
```

Submit

You have used 10 of 10 attempts

ℹ Answers are displayed within the problem

Question 7

0/1 point (graded)

Notice that there are several versions of the book. The `gutenberg_works` function filters this table to remove replicates and include only English language works. Use this function to find the ID for *Pride and Prejudice*.

What is the correct ID number?

Read the `gutenberg_works` documentation to learn how to use the function.

0

✖ **Answer:** 1342

\(\)

Answer code

```
gutenberg_works(title == "Pride and Prejudice")$gutenberg_id
```

Submit

You have used 10 of 10 attempts

ℹ Answers are displayed within the problem

Question 8

0/1 point (graded)

Use the `gutenberg_download` function to download the text for Pride and Prejudice. Use the **tidytext** package to create a tidy table with all the words in the text. Save this object as `words` .

How many words are present in the book?

0

✖ **Answer:** 122204

\(\)

Answer code

```
book <- gutenberg_download(1342)
words <- book %>%
  unnest_tokens(word, text)
nrow(words)
```

Submit

You have used 10 of 10 attempts

ℹ Answers are displayed within the problem

Question 9

0/1 point (graded)

Remove stop words from the `words` object. Recall that stop words are defined in the `stop_words` data frame from the **tidytext** package.

How many words remain?

0

✖ **Answer:** 37246

\(\)

Answer code

```
words <- words %>% anti_join(stop_words)
nrow(words)
```

Submit

You have used 10 of 10 attempts

ℹ Answers are displayed within the problem

Question 10

0/1 point (graded)

After removing stop words, detect and then filter out any token that contains a digit from `words`.

How many words remain?

0

✖ **Answer:** 37180

\(\)

Answer code

```
words <- words %>%
  filter(!str_detect(word, "\\d"))
nrow(words)
```

Submit

You have used 10 of 10 attempts

ℹ Answers are displayed within the problem

Question 11

0/3 points (graded)

Analyze the most frequent words in the novel after removing stop words and tokens with digits.

How many words appear more than 100 times in the book?

0

✖ **Answer:** 23

\(\)

Answer code

```
words %>%
    count(word) %>%
    filter(n > 100) %>%
    nrow()
```

What is the most common word in the book?

0

✖ **Answer:** elizabeth **or** Elizabeth

Answer code

```
words %>%
    count(word) %>%
    top_n(1, n) %>%
    pull(word)
```

How many times does that most common word appear?

0

✖ **Answer:** 597

\(\)

Answer code

```
words %>%
    count(word) %>%
    top_n(1, n) %>%
    pull(n)
```

Submit

You have used 10 of 10 attempts

ℹ Answers are displayed within the problem

Question 12

0/3 points (graded)

Define the `afinn` lexicon:

```
afinn <- get_sentiments("afinn")
```

Note that this command will trigger a question in the R Console asking if you want to download the AFINN lexicon. Press 1 to select "Yes" (if using RStudio, enter this in the Console tab).

Use this `afinn` lexicon to assign sentiment values to `words`. Keep only words that are present in both `words` and the `afinn` lexicon. Save this data frame as `afinn_sentiments`.

How many elements of `words` have sentiments in the `afinn` lexicon?

0

✖ **Answer:** 6065

\(\)

Answer code

```
afinn_sentiments <- inner_join(afinn, words)
nrow(afinn_sentiments)
```

What proportion of words in `afinn_sentiments` have a positive value?

0

✖ **Answer:** 0.563

\(\)

Answer code

```
mean(afinn_sentiments$value > 0)
```

How many elements of `afinn_sentiments` have a value of 4?

0

✖ **Answer:** 51

\(\)

Answer code

```
sum(afinn_sentiments$value == 4)
```

Submit

You have used 10 of 10 attempts

ℹ Answers are displayed within the problem