Please do your homework using an R script. Homework is NOT collected in this class. However, on the day it is due, you may be asked to share your screen and run some of your code when we discuss this homework in class. This will count towards your participation grade.

1. Create a corpus in R from the PubMed file Covid19March1.txt. We are going to find and visualize the most upbeat document as well as the most depressing document in this corpus.  
   1. Create a term document matrix in R for this corpus. You may call it tdM. Note that we want the actual matrix.
   2. Create a data frame called tdMdf with all columns of tdM, but with an additional column called word which holds the word in the row names of tdM. Join this data frame with the afinn sentiment lexicon and overwrite tdMdf with the join. This should remove all rows that have no match with the lexicon, and it should also add a column called value. However, the row names will have disappeared, so recreate them using the column word, then remove that column (set it to NULL). What is the dimension of the matrix tdMdf now?
   3. Create a vector called docs that contains the indices of those documents that contain at least one of the words in the tdMdf data frame. **Hint:** Use the apply function to obtain the sum of the frequencies in tdMdf for each document, then find the indices for those documents that have a non-zero sum. How many documents like that did you find?
   4. Create a new data frame, tdMdf2, that contains the total sentiment for each word in tdMdf by each document in tdMdf. In other words, for each document you must multiply the frequency of each word by its sentiment. This data frame should have one column less than tdMdf since it will no longer have the value column.
   5. For the first document (with index equal to 1) show the frequencies and sentiment values for the words that do not have zero frequency. This information comes from tdMdf; also obtain the total nonzero sentiment for each word of document 1; this information comes from tdMdf2. Check that these numbers are the product of the numbers from tdMdf. Do the same for document 3. Note that the argument drop=F when subsetting data will display the output as a data frame, even if only one column is returned. This allows you to see the row names (the words, in this case).
   6. Create a vector called sumSent that has the sum of the sentiments for each document; this information comes from tdMdf2. Then create a scatterplot for all documents in the vector docs (created in c) where you plot the sentiment of the document against its index. Also plot a horizontal red line at the point where the sentiment is equal to zero.
   7. Create a histogram for sumSent, with a title and labels that make sense. What is the shape of this histogram? Use the skew.ratio function in the sur package to find out the skewness of the histogram. Find the mean and the median for this data set, and state what a typical value is for the sentiment of a document in this corpus.
   8. Find the indices of the document with the highest total sentiment value. What is the total sentiment value for this document? Show the words in this document that have nonzero frequency; show their frequencies as well as the sentiment values for those words. Then inspect this document in the corpus. Finally, do the same for the document with the lowest sentiment. **Hint:** use the which.max and which.min functions.
   9. For the two documents in h), create a term frequency vector for each (use termFreq). Then create two different types of word cloud for each document. Create the clouds any way you like but try to make them look good.