Gov 2018: Lab Text preprocessing

Your name:

March 29, 2022

I. Preprocessing a Corpus and working with DTM

1. Preprocessing a Corpus

You'll need the below packages:

```
rm(list=ls())
library(tm)
```

```
## Loading required package: NLP
```

```
library(qdapDictionaries)
library(dplyr) # Data preparation and pipes $>$
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2) # for plotting word frequencies
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:NLP':
##
##     annotate
```

```
library(SnowballC) # for stemming
```

A corpus is a collection of texts, usually stored electronically, and from which we perform our analysis. A corpus might be a collection of news articles from Reuters or the published works of Shakespeare.

Within each corpus we will have separate articles, stories, volumes, each treated as a separate entity or record. Each unit is called a "document."

For this first part of the lab, you'll be using a section of Machiavelli's Prince as our corpus. Since The Prince is a monograph, we have already "chunked" the text, so that each short paragraph or "chunk" is considered a "document."

1.1 Corpus Sources and Readers

The tm package supports a variety of sources and formats. Run the code below to see what it includes

```
getSources()
```

```
## [1] "DataframeSource" "DirSource"       "URISource"       "VectorSource"
## [5] "XMLSource"       "ZipSource"
```

```
getReaders()
```

```
##  [1] "readDataframe"           "readDOC"
##  [3] "readPDF"                 "readPlain"
##  [5] "readRCV1"                "readRCV1asPlain"
##  [7] "readReut21578XML"        "readReut21578XMLasPlain"
##  [9] "readTagged"              "readXML"
```

Reading documents from the mach.csv file. Each row is a document, and columns are text and metadata (information about each document). This is the easiest option if you have metadata.

```
docs.df <-read.csv("mach.csv", header=TRUE) #read in CSV file
docs <- Corpus(VectorSource(docs.df$text))
docs
```

```
## <<SimpleCorpus>>
## Metadata:  corpus specific: 1, document level (indexed): 0
## Content:  documents: 188
```

Once you have the corpus, inspect the documents using inspect()

```
# see the 16th document

inspect(docs)
```

```
## <<SimpleCorpus>>
## Metadata:  corpus specific: 1, document level (indexed): 0
## Content:  documents: 188
##
##   [1] DEDICATORY LETTER Niccolo Machiavelli to His Magnificence Lorenzo de' Medici  Those who wish t
##   [2]  But considerable problems arise if territories are annexed in a country that differs in langua
##   [3] them, saying that he could not fight well with them, and he therefore wanted to confront the e
##   [4] the Swiss, they are not confident of being able to win battles without them. The outcome is tha
##   [5] Empire; and all the vigour that was drained from it was received by the Goths.  I conclude, th
##   [6]  XIV: How a ruler should act concerning military matters  A ruler, then, should have no other
##   [7] citizens. For being unarmed (apart from other bad consequences) results in your being despised
##   [8] war. There are two ways of doing this: one is by going on exercises; the other is by study.  W
##   [9]  One of the reasons why historians have praised Philopoemen, the leader of the Achaean League,
##  [10] especially for the light they shed on the actions of eminent men: to find out how they waged wa
##  [11]  XV: The things for which men, and especially rulers, are praised or blamed  It remains now to
##  [12]  I shall set aside fantasies about rulers, then, and consider what happens in fact. I say that
##  [13] have more reason to be devoted to him if they intend to behave well, and to fear him if they do
##  [14] held to be good. But because it is not possible to have all of them, and because circumstances
##  [15] considered generous, it will harm you. If it is practised virtuously, and as it should be, it w
##  [16]  Therefore, since a ruler cannot both practise this virtue of generosity and be known to do so
##  [17] able to wage war. The present King of France has fought many wars without imposing any special
##  [18] those who sought power in Rome; but if after gaining power he had survived, and had not moderat
##  [19] it. It is only giving away what belongs to yourself that harms you.  There is nothing that is s
##  [20] one should take care not to be merciful in an inappropriate way. Cesare Borgia was considered
```

```
##  [21] dangers. Virgil makes Dido say: Res dura, et regni novitas me talia cogunt moliri, et late fine
##  [22] benefit them they are all devoted to you: they would shed their blood for you; they offer thei
##  [23]  However, when a ruler is with his army, and commands a large force, he must not worry about be
##  [24] therefore be reassured, and will be afraid of causing trouble, for fear that they will be dispo
##  [25] happened to Scipio, considered a most remarkable man not only in his own times but in all other
##  [26] harmful quality was not only concealed but contributed to his glory.  Returning to the matter o
##  [27]  You should know, then, that there are two ways of contending: one by using laws, the other, fo
##  [28] recognise traps, and a lion to frighten away wolves. Those who rely merely upon a lion's streng
##  [29] much dominated by immediate needs, that a skilful deceiver always finds plenty of people who wi
##  [30]  A ruler, then, should be very careful that everything he says is replete with the five above-m
##  [31] will always be judged to be honourable and be praised by everyone. For the common people are im
##  [32] censurable faults  will involve him in any danger.  What will make him hated, above all else, a
##  [33]  A ruler who succeeds in creating such an image of himself will enjoy a fine reputation; and it
##  [34]  But with regard to one's subjects, if there is no external threat, one's only fear must be tha
##  [35] this nuisance, and becomes hostile to the ruler. And they are dangerous enemies because, althou
##  [36] your plan to a malcontent, you enable him to become contented, because obviously he can now exp
##  [37]  Countless examples could be given on this subject; but I shall limit myself to one, which occu
##  [38] enough to rule.  I conclude, then, that rulers should worry little about being plotted against
##  [39] people hated the nobles because they were afraid of them, he wanted to protect the people. He o
##  [40]  It will perhaps seem to many people that, if the lives, careers and deaths of some of the Roma
##  [41] be made is that, whereas in other principalities only the ambition of the nobles and the insole
##  [42] men), when they realised how difficult it was to satisfy these two conflicting tendencies, trie
##  [43] died honoured by all, because he became emperor by hereditary right, and did not owe his power
##  [44] necessary for maintaining your power is corrupt, you are forced to indulge its proclivities in
##  [45]  Severus possessed so much ability that he was able to keep the soldiers friendly, and rule suc
##  [46] aided their invasion. What usually happens is that, as soon as a strong invader attacks a count
##  [47] to become emperor, he took his army towards Rome with such speed that he reached Italy before i
##  [48] true. But when Severus had defeated and killed Nigrinus, and the eastern part of the Empire was
##  [49] deeds.  His son Antoninus was also a man with many very fine qualities, who was greatly admired
##  [50] rash decision and likely to cause his downfall, as indeed happened.  But let us consider Commod
##  [51]  It remains to discuss the character of Maximinus. He was an extremely warlike man; and since t
##  [52] because they were afraid of his brutality, first Africa revolted, then the senate and all the p
##  [53] administering provinces, as the armies of the Roman Empire did. Hence, if it was then more nece
##  [54]  But let us return to our subject. I maintain that anyone who considers what I have written wil
##  [55]  XX: Whether building fortresses, and many other things that rulers frequently do, are useful o
##  [56] converted into firm adherents. Since it is not possible to provide all your subjects with arms
##  [57] conquered. They established colonies, they had friendly relations with the less powerful (thoug
##  [58] you against powerful enemies and hostile subjects.  As I have said, then, new rulers of new pri
##  [59] more easily. This policy may have been sound in the period when there was a certain equilibrium
##  [60] the Venetians of all their land empire. The use of such methods, then, indicates that a ruler i
##  [61] encourage hostile forces cunningly, so that when he crushes them his reputation and power will
##  [62]  Moreover, since this matter is important, I do not want to fail to remind any ruler who has re
##  [63] were disaffected.  Rulers have been accustomed to build fortresses to strengthen their power. T
##  [64] depends on the circumstances. Moreover, if they help you in some respects, they will be harmful
##  [65] circumstances were such that no foreign power was able to help the people. But fortresses were
##  [66] become the most famous and glorious king in Christendom. And if his achievements are examined,
##  [67]  It is also very beneficial for a ruler to perform very unusual deeds within his kingdom, such
##  [68] them. For if the first signs of trouble are perceived, it is easy to find a solution; but if ot
##  [69]  Antiochus invaded Greece, invited there by the Aetolians in order to drive out the Romans. An
##  [70] victorious: even if he is powerful and you are at his mercy, he is beholden to you and friendsh
##  [71] win, the ruler whom you help will be at your mercy. (And it is certainly to be expected that he
##  [72]  No government should ever believe that it is always possible to follow safe policies. Rather,
##  [73] careful to preserve the prestige of his office, for this is something that should never be dimi
##  [74] kinds of mind: the first grasps things unaided; the second when they are explained; the thit:d
```

```
##  [75] who governs a state' should never think about himself or his own affairs but always about the i
##  [76] rulers easily make mistakes, unless they are very shrewd and are skilful at choosing men. I rei
##  [77] decisions. He should so conduct himself with his advisers that they will all realise that the n
##  [78] carry out his plans, those in his court get to know about them, and then advise him to act diff
##  [79] were merely brewing, were always able to overcome them. They never allowed them to develop in c
##  [80] is shrewd, this is undoubtedly a mistaken view. For it is an infallible rule that a prince who
##  [81] its source in the shrewdness of the ruler; the ruler's shrewdness cannot derive from sound advi
##  [82] reliable allies and exemplary conduct. But a man who inherits a principality and loses it throu
##  [83] and knew how to win over the people and how to deal with the nobles, he was able to carry on th
##  [84]  XXV: How much power fortune has over human affairs, and how it should be resisted  I am not un
##  [85] half.  I compare fortune to one of those dangerous rivers that, when they become enraged, flood
##  [86]  Considering the matter in more detail, I would observe that one sees a ruler flourishing today
##  [87] acting cautiously and the other impetuously. The reason for these different outcomes is whether
##  [88] cautious man to act expeditiously, he does not know how to do it; this leads to his failure. Bu
##  [89] Kingdom of Naples. On the other hand, Julius involved the King of France: for that King saw the
##  [90] he followed any of the policies I have advocated. I shall discuss Louis, not Charles; since he
##  [91] failure. But if circumstances had changed so that it was imperative to act cautiously, he would
##  [92]  Bearing in mind all the matters previously discussed, I ask myself whether the present time is
##  [93] stability, beaten, despoiled, lacerated, overrun, in short, utterly devastated. And although re
##  [94] illustrious family which (because it is successful and talented, and favoured by God and by the
##  [95] cloud has shown you the way; water has flowed from the rock; manna has rained down here. Everyt
##  [96] to achieving greatness,  will make him revered and admired; and in Italy there is no lack of ma
##  [97] confirm this judgement.  If your illustrious family, then, wants to emulate those great men who
##  [98] against infantry that fight as strongly as they do themselves. Thus, it has been seen, and expe
##  [99]  This opportunity to provide Italy with a liberator, then, after such a long time, must not be
## [100] not committed other errors. When he had conquered Lombardy, then, the King at once regained the
## [101] Magnificence, I trust very much that your humanity will lead you to accept it, since it is not
## [102] Church, some of the Venetians), were forced to remain allied to him. And with their help he cou
## [103] Naples, he divided it with the King of Spain. And whereas previously Louis had been arbiter of
## [104] Venetians deserves to be excused, because it enabled him to gain a foothold in Italy, this othe
## [105] they would always have prevented the other powers from intervening in Lombardy; they would neve
## [106]  King Louis, then, lost Lombardy because he did not follow any of the policies followed by thos
## [107]  From this may be derived a generalisation, which is almost always valid: anyone who enables an
## [108] of two ways: either by one ruler, who is helped to govern the kingdom by others, who are in rea
## [109] Kingdom is divided into sanjaks, to which he sends various administrators, whom he changes and
## [110] the ruler will rebel, thus facilitating the invasion, for the reasons already mentioned. Since
## [111]  The opposite occurs in kingdoms ruled like France, because it is easy for you to make headway
## [112] low ground, in order to understand the character of the mountains and other high points, and cl
## [113] Alexander was forced to make a frontal assault and win a decisive victory; afterwards, since Da
## [114] authority he acquired within it. And because their old hereditary ruling families no longer exi
## [115] live under their own laws, exacting tribute and setting up an oligarchical government that will
## [116] cities is the only certain way of holding them. Anyone who becomes master of a city accustomed
## [117] obeying but lack their older ruler; they are unable to agree on making one of themselves ruler
## [118] behaviour. Since it is not always possible to follow in the footsteps of others, or to equal th
## [119] that one or other of these would, to some degree, mitigate many of the difficulties. Neverthele
## [120] from those of Moses, who had such a great master.  If their deeds and careers are examined, it
## [121] the Medes, and that the Medes should have been soft and weak because of the long peace.  And Th
## [122] the innovator, they do it with much vigour, whereas his supporters act only half-heartedly; so
## [123]  All the states, all the dominions that have held sway over men, have been either republics or
## [124]  If Moses, Cyrus, Theseus and Romulus had been unarmed, the new order which each of them estab
## [125] already discussed. But it certainly is worthy of mention in this context, so let it suffice for
## [126]  VII: New principalities acquired through the power of others and their favour  Private citizen
## [127]  To illustrate these two methods of becoming ruler, namely, through ability or through favour c
## [128]  If the whole career of the Duke is considered, then, it will be seen that he succeeded in layi
```

```
## [129] already under the protection of the Venetians). Apart from this problem, Alexander saw that the
## [130] Milan than the Pope received troops from him for his own campaign in the Romagna, which was mad
## [131] when, after he had captured the Duchy of Urbino, he attacked Tuscany, for the King made him aba
## [132] spelt their ruin, and called a diet at Magione, in the province of Perugia.  This meeting gave
## [133] good basis for his power, because he controlled all the Romagna, together with the Duchy of Urb
## [134] and maintained.  I say, then, that states which are hereditary, and accustomed to the rule of t
## [135] Remirro quickly restored order and peace, and acquired a very formidable reputation. Later, the
## [136]  But let me continue from where I left off. I say that the Duke was very powerful, and secure i
## [137] and seek to take away what Alexander had given to him. He decided to protect himself against th
## [138] Roman nobles, and most of the cardinals. As for annexing new territories, he had planned to bec
## [139]  However, five years after the Duke had taken up the sword, Alexander died.  He found himself i
## [140] Baglioni, Vitelli and Orsini came to Rome, they were unable to stir anyone up against him. More
## [141]  Hence, anyone who considers it necessary in his new principality to deal effectively with his
## [142] ensure that the man he favoured was made pope, he could have prevented certain other choices. A
## [143] who thinks that new benefits make important men forget old injuries is mistaken.  The Duke, the
## [144]  Agathocles the Sicilian, who became King of Syracuse, was not only an ordinary citizen, but or
## [145] continuity of his family's rule extinguishes the memories of the causes of innovations: for any
## [146] control of the city, and thereafter held it without any civil strife. Although he was twice def
## [147] action.  Yet it cannot be called virtue to kill one's fellow-citizens, to betray one's friends
## [148] to see him and his own city, and to inspect in some measure his own patrimony. Since achieving
## [149] invited Giovanni Fogliani and all the leading citizens of Fermo. After the banquet, and all the
## [150] head. And when he had killed all the malcontents who could have harmed him, he consolidated his
## [151] power by acting cruelly even in peaceful times let alone in times of war, which are always unco
## [152] them at once, so as not to have to inflict punishments every day. Thus he will be able, by his
## [153]  IX: The civil principality  I turn now to the other case, when a private citizen becomes ruler
## [154] the nobles, according to whether one or the other has the opportunity to act. As for the nobles
## [155] who becomes ruler through popular support finds himself standing alone, having around him noboc
## [156] whom you have injured in annexing a principality, yet you cannot retain the friendship of those
## [157]  To clarify this matter, let me say that two main considerations need to be borne in mind with
## [158] watch these nobles very carefully, and fear them as much as if they were declared enemies, beca
## [159]  Nabis, ruler of the Spartans, withstood a siege by all the other Greek powers and by a triumph
## [160] courageous, does not despair in difficult times, and maintains the morale of his people by his
## [161] him in such a crisis. And in difficult times, he will always lack men on whom he can depend.  I
## [162] There is another consideration that must be borne in mind when examining the strength of princi
## [163] about it. No advice can be given with regard to the second case, except to exhort such a ruler
## [164] people without public expense, they always have enough raw materials to keep the people engaged
## [165] reply that a strong and spirited ruler will always overcome such problems, by encouraging his s
## [166]  XI: Ecclesiastical principalities  It remains now only to discuss ecclesiastical principaliti
## [167] reconquered a second time, it is less likely to be lost, since the ruler, because of the rebell
## [168] since they are raised up and maintained by God, only a presumptuous and rash man would examine
## [169] secondly, that none of the other Italian powers should acquire more territory and power. Those
## [170] one pope almost succeeded in destroying the Colonna faction, the next pope would be hostile to
## [171] Church already powerful, possessing all the Romagna, the Roman barons reduced to impotence, and
## [172]  His Holiness Pope Leo,  then, has found the Papacy very powerful; and it is to be hoped that,
## [173] new, old or mixed) are good laws and good armies. Since it is impossible to have good laws if g
## [174] required to fight. In peaceful times you will be despoiled by them, in war by your enemies. The
## [175] to conquer Italy with a piece of chalk; and he who said that our sins were responsible spoke th
## [176]  For many centuries both Rome and Sparta were armed and independent. Today the Swiss are very w
## [177] forced to seek help from the King of Aragon.  And although the Venetians and the Florentines au
## [178]  I say, then, that the territories a conqueror annexes and joins to his own well-established st
## [179]  But let us turn to more recent events. The Florentines made Paulo Vitelli their general; he wa
## [180] expand their land empire, they had little reason to be afraid of their mercenary generals, beca
## [181] fear was losing, not the dangers arising from their being victorious, as indeed happened later
## [182] encouraged these revolts in order to increase its temporal power; and in many other cities rule
```

```
## [183] did not possess states of their own and lived by being mercenaries, small numbers of foot-soldi
## [184] have said, to avoid hardship and danger. The outcome of their activities is that Italy has beco
## [185] discuss this recent case of Pope Julius II. His decision can only be judged rash: to put himsel
## [186] infidels.  Therefore, anyone who wants to be unable to conquer should use such troops, because
## [187] not consider a victory that is gained by using foreign forces to be genuine.  I never hesitate
## [188] he was the complete master of his own forces.  I am reluctant to cite examples that are neither
```

And see the text using the `as.chracter`

```
 # see content for 16th document

as.character(docs[16])
```

```
## [1] " Therefore, since a ruler cannot both practise this virtue of generosity and be known to do so w
## [2] "list(language = \"en\")"
## [3] "list()"
```

1.2 Preprocessing functions

Many text analysis applications follow a similar 'recipe' for preprecessing, involving:

1. Tokenizing the text to unigrams (or bigrams, or trigrams)
2. Converting all characters to lowercase
3. Removing punctuation
4. Removing numbers
5. Removing Stop Words, inclugind custom stop words
6. "Stemming" words, or lemmitization. There are several stemming alogrithms. Porter is the most popular.
7. Creating a Document-Term Matrix
8. Weighting features
9. Removing Sparse Terms

See what transformations are available TM package.

```
getTransformations()
```

```
## [1] "removeNumbers"     "removePunctuation" "removeWords"
## [4] "stemDocument"      "stripWhitespace"
```

The function `tm_map()` is used to apply one of these transformations across all documents.

```
docs <- tm_map(docs, content_transformer(tolower)) # convert all text to lower case
```

```
## Warning in tm_map.SimpleCorpus(docs, content_transformer(tolower)):
## transformation drops documents
```

```
as.character(docs[[16]])
```

```
## [1] " therefore, since a ruler cannot both practise this virtue of generosity and be known to do so w
```

Using `tm_map`, apply the following transformations. You may have to look up the help files for these functions. 1. removePunctuation 2. removeNumbers 3. removeWords (see help file to remove stop words) 4. stripWhitespace 5. stemDocument

```
# remove Punctuation

docs1 <- tm_map(docs, removePunctuation)
```

```
## Warning in tm_map.SimpleCorpus(docs, removePunctuation): transformation drops
## documents
```

```
# remove Numbers

docs1 <- tm_map(docs1, removeNumbers)
```

```
## Warning in tm_map.SimpleCorpus(docs1, removeNumbers): transformation drops
## documents
```

```
# remove common words
docs1 <- tm_map(docs1, removeWords, stopwords("english"))
```

```
## Warning in tm_map.SimpleCorpus(docs1, removeWords, stopwords("english")):
## transformation drops documents
```

```
# remove own stop words (e.g. "prince")
docs1 <- tm_map(docs1, removeWords, words = "prince")
```

```
## Warning in tm_map.SimpleCorpus(docs1, removeWords, words = "prince"):
## transformation drops documents
```

```
# strip white space

docs1 <- tm_map(docs1, stripWhitespace)
```

```
## Warning in tm_map.SimpleCorpus(docs1, stripWhitespace): transformation drops
## documents
```

```
# stem the document

docs1 <- tm_map(docs1, stemDocument)
```

```
## Warning in tm_map.SimpleCorpus(docs1, stemDocument): transformation drops
## documents
```

1.3 Creating a DTM

A document term matrix is simply a matrix with documents as the rows and terms as the columns and a count of the frequency of words as the cells of the matrix. Use `DocumentTermMatrix()` to create the matrix and call it an object `dtm`.

```
dtm <-DocumentTermMatrix(docs1)
```

`tm` also lets us convert a corpus to a DTM while completing the pre-processing steps in one step.

```
dtm <- DocumentTermMatrix(docs,
           control = list(stopwords = TRUE,
                          tolower = TRUE,
                          removeNumbers = TRUE,
                          removePunctuation = TRUE,
                          stemming=TRUE))
```

1.4 Weighting

One common pre-processing step that some applicaitons may call for is applying tf-idf weights. The tf-idf, or term frequency-inverse document frequency, is a weight that ranks the importance of a term in its contextual document corpus. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to adjust for the fact that

some words appear more frequently in general. In other words, it places importance on terms frequent in the document but rare in the corpus.

```
dtm.weighted <- DocumentTermMatrix(docs,
          control = list(weighting =function(x) weightTfIdf(x, normalize = TRUE),
                         stopwords = TRUE,
                         tolower = TRUE,
                         removeNumbers = TRUE,
                         removePunctuation = TRUE,
                         stemming=TRUE))
```

```
## Warning in TermDocumentMatrix.SimpleCorpus(x, control): custom functions are
## ignored
```

Compare first 5 rows and 5 columns of the `dtm` and `dtm.weighted`. What do you notice?

hint: Use the `inspect` function and pass your subsetted dtm object.

```
inspect(dtm[1:5,1:5])
```

```
## <<DocumentTermMatrix (documents: 5, terms: 5)>>
## Non-/sparse entries: 3/22
## Sparsity           : 88%
## Maximal term length: 7
## Weighting          : term frequency (tf)
## Sample             :
##     Terms
## Docs abandon abil abject abl ablest
##    1       0    0      0   0      0
##    2       0    1      0   0      0
##    3       0    0      0   0      0
##    4       0    1      0   1      0
##    5       0    0      0   0      0
```

```
inspect(dtm.weighted[1:5,1:5])
```

```
## <<DocumentTermMatrix (documents: 5, terms: 5)>>
## Non-/sparse entries: 3/22
## Sparsity           : 88%
## Maximal term length: 7
## Weighting          : term frequency - inverse document frequency (normalized) (tf-idf)
## Sample             :
##     Terms
## Docs abandon       abil abject        abl ablest
##    1       0 0.00000000      0 0.00000000      0
##    2       0 0.04019928      0 0.00000000      0
##    3       0 0.00000000      0 0.00000000      0
##    4       0 0.03102336      0 0.02277345      0
##    5       0 0.00000000      0 0.00000000      0
```

2. Exploring the DTM

2.1 Dimensions

Look at the structure of our DTM. Print the dimensions of the DTM. How many documents do you have? How many terms?

```
# how many documents? how many terms?

mat <- as.matrix(dtm)

dim(mat)
```

```
## [1]  188 2368
```

```
# 188 documents, 2368 terms
```

2.2 Frequencies

Obtain the term frequencies as a vector by converting the document term matrix into a matrix and using `colSums` to sum the column counts:

```
# term frequencies as a vector

freq <- colSums(mat)
freq[1:10]
```

```
##   abandon      abil    abject       abl    ablest abovement  abovenam   absolut
##         4        35         1        61         1         5         1         2
##    absorb    accept
##         1         6
```

By ordering the frequencies you can list the most frequent terms and the least frequent terms. Print out the head of the least and most frequent terms.

```
# order

ord <- order(freq)

# Least frequent terms
freq[head(ord)]
```

```
##     abject     ablest   abovenam     absorb     access accomplish
##          1          1          1          1          1          1
```

```
# most frequent
freq[tail(ord)]
```

```
##   men peopl   one power  will ruler
##    95    98   168   169   251   280
```

2.3 Plotting frequencies

Make a plot that shows the frequency of frequencies for the terms. (For example, how many words are used only once? 5 times? 10 times?)

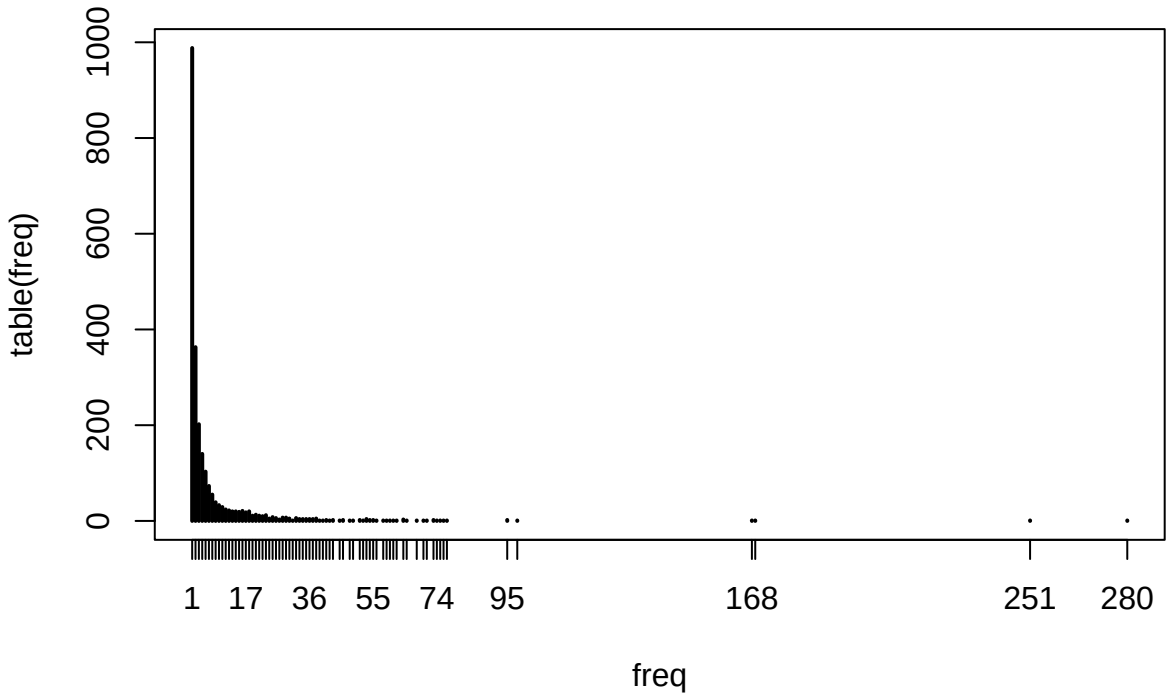
```
# frequency of frenquencies
head(table(freq),15)
```

```
## freq
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
## 988 363 202 140 103  73  55  39  33  29  24  22  20  20  19
```

```
tail(table(freq),15)
```

```
## freq
##  65  68  70  71  73  74  75  76  77  95  98 168 169 251 280
##   1   1   1   1   2   1   1   1   1   2   1   1   1   1   1
```

```
# plot
plot(table(freq))
```



Reorder columns of DTM to show most frequent terms first, and inspect the first five rows and first five columns.

```
dtm.ordered <- dtm[,order(freq, decreasing = T)]
inspect(dtm.ordered[1:5,1:5])
```

```
## <<DocumentTermMatrix (documents: 5, terms: 5)>>
## Non-/sparse entries: 10/15
## Sparsity           : 60%
## Maximal term length: 5
## Weighting          : term frequency (tf)
## Sample             :
##     Terms
## Docs one peopl power ruler will
##    1   0     0     0     1    1
##    2   3     0     0     1    3
##    3   0     0     0     0    0
##    4   0     0     0     1    1
##    5   3     0     0     1    1
```

2.4 Exploring word frequences

The TM package has lots of useful functions to help you explore common words and associations. Use `findFreqTerms` to find the words that appear at least 100x. Use `findAssoc` to find words that correlate with war (use as the third parameter 0.3).

```
# Have a look at common words
findFreqTerms(dtm, lowfreq=100)
```

```
## [1] "one"   "power" "ruler" "will"
```

```
# Which words correlate with "war"?
findAssocs(dtm, "war", 0.3)
```

```
## $war
##     wage    fight  antioch     argu     brew    induc      lip    maxim
##     0.73     0.52     0.45     0.45     0.45     0.45     0.45     0.45
##  relianc     sage    trifl  postpon     mere     evil    avoid     flee
##     0.45     0.45     0.45     0.41     0.35     0.34     0.32     0.32
##   occupi     glad glorious    heard     hunt ineffect     knew   produc
##     0.32     0.30     0.30     0.30     0.30     0.30     0.30     0.30
## temporis
##     0.30
```

Make wordclouds showing the most common terms:

```
# frequency of the words
freq <- sort(colSums(as.matrix(dtm)),decreasing=TRUE)
head(freq)
```

```
## ruler  will power   one peopl alway
##   280   251   169   168    98    95
```

```
# wordcoulds!
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
set.seed(123)
wordcloud(names(freq), freq, max.words=100, colors=brewer.pal(6,"Dark2"))
```

2.5 Remove sparse terms.

Sometimes we want to remove sparse terms and thus increase efficiency. Look up the help file for the function `removeSparseTerms`. Using this function, create an objected called `dtm.s` that contains only terms with <.9 sparsity (meaning they appear in more than 10% of documents).

```
dtm.s <- removeSparseTerms(dtm,.9)
dtm # 2365 terms
```

```
## <<DocumentTermMatrix (documents: 188, terms: 2368)>>
## Non-/sparse entries: 11754/433430
## Sparsity           : 97%
## Maximal term length: 15
## Weighting          : term frequency (tf)
```

```
dtm.s # 135 terms
```

```
## <<DocumentTermMatrix (documents: 188, terms: 136)>>
## Non-/sparse entries: 4353/21215
## Sparsity           : 83%
## Maximal term length: 12
## Weighting          : term frequency (tf)
```

```
dtm.s.matrix <- as.matrix(dtm.s)
colSums(dtm.s.matrix) / nrow(dtm.s.matrix)
```

```
##         abil          abl       achiev       acquir          act      alexand
##    0.1861702    0.3244681    0.1914894    0.1489362    0.3404255    0.2074468
##         also     although        alway        anyon          arm         armi
##    0.1170213    0.1702128    0.5053191    0.2021277    0.1595745    0.3297872
```

```
##       attack        avoid        becam        becom       better          can
##    0.2180851    0.1755319    0.1117021    0.3882979    0.1223404    0.3191489
##         care         caus        chang     circumst         citi      citizen
##    0.1010638    0.1223404    0.1542553    0.1329787    0.1861702    0.1648936
##       consid      control      countri       danger         deed       defend
##    0.3617021    0.1382979    0.2021277    0.1968085    0.1063830    0.1329787
##       depend    difficult   difficulti      discuss         duke       either
##    0.1170213    0.1542553    0.1595745    0.1808511    0.2180851    0.2872340
##        enemi    establish         even      everyon       exampl       favour
##    0.1702128    0.1276596    0.1542553    0.1329787    0.1063830    0.2553191
##         fear        fight         find        first       follow         forc
##    0.2287234    0.1489362    0.1382979    0.2446809    0.2021277    0.3085106
##      foreign        found         franc         gain         good       govern
##    0.1329787    0.1702128    0.1808511    0.1436170    0.1968085    0.1755319
##        great       happen         harm         hate         help         hold
##    0.3404255    0.1808511    0.1542553    0.1542553    0.2287234    0.1861702
##       hostil        itali         keep         kill         king      kingdom
##    0.1170213    0.2234043    0.1968085    0.1382979    0.3138298    0.1861702
##         lack          let         like        littl         live         long
##    0.1542553    0.1489362    0.1382979    0.1170213    0.1702128    0.1010638
##         luck         made     maintain         make          man         mani
##    0.1223404    0.1595745    0.2712766    0.2819149    0.2819149    0.3404255
##       matter          may         mean          men     militari       moreov
##    0.1914894    0.2021277    0.1223404    0.5053191    0.1595745    0.1223404
##         much         must    necessari         need        never nevertheless
##    0.2978723    0.2765957    0.1914894    0.1755319    0.2925532    0.1276596
##          new          now          one        order        other        peopl
##    0.4042553    0.1117021    0.8936170    0.2925532    0.3989362    0.5212766
##       permit        polici         pope        posit      possess      possibl
##    0.1170213    0.1382979    0.2127660    0.1489362    0.1808511    0.1223404
##        power      princip       realis       reason       remain        reput
##    0.8989362    0.2606383    0.1223404    0.2819149    0.1276596    0.1276596
##       result        roman         rule        ruler         said          say
##    0.1117021    0.1542553    0.1968085    1.4893617    0.1489362    0.1170213
##         secur         sinc      soldier        state       strong      subject
##    0.1489362    0.3936170    0.2393617    0.2819149    0.1063830    0.2872340
##         take     territori     therefor        thing         thus         time
##    0.1223404    0.1489362    0.2021277    0.2446809    0.1329787    0.4095745
##        troop        troubl          two          use         want          way
##    0.2712766    0.1436170    0.1914894    0.3723404    0.3882979    0.3776596
##         well      whether         will      without
##    0.3457447    0.1436170    1.3351064    0.1702128
```

(Optional Exercise) 3. Exporting the DTM

3.1

Convert a DTM to a matrix or data.frame in order to write to a csv, add meta data, etc.

First create an object that converts the `dtm` to a dataframe, and call it `dtm` again.

```
# coerce into dataframe
dtm <- as.data.frame(as.matrix(dtm))
names(docs)  # names of documents
```

```
##   [1] "1"   "2"   "3"   "4"   "5"   "6"   "7"   "8"   "9"   "10"  "11"  "12"
##  [13] "13"  "14"  "15"  "16"  "17"  "18"  "19"  "20"  "21"  "22"  "23"  "24"
##  [25] "25"  "26"  "27"  "28"  "29"  "30"  "31"  "32"  "33"  "34"  "35"  "36"
##  [37] "37"  "38"  "39"  "40"  "41"  "42"  "43"  "44"  "45"  "46"  "47"  "48"
##  [49] "49"  "50"  "51"  "52"  "53"  "54"  "55"  "56"  "57"  "58"  "59"  "60"
##  [61] "61"  "62"  "63"  "64"  "65"  "66"  "67"  "68"  "69"  "70"  "71"  "72"
##  [73] "73"  "74"  "75"  "76"  "77"  "78"  "79"  "80"  "81"  "82"  "83"  "84"
##  [85] "85"  "86"  "87"  "88"  "89"  "90"  "91"  "92"  "93"  "94"  "95"  "96"
##  [97] "97"  "98"  "99"  "100" "101" "102" "103" "104" "105" "106" "107" "108"
## [109] "109" "110" "111" "112" "113" "114" "115" "116" "117" "118" "119" "120"
## [121] "121" "122" "123" "124" "125" "126" "127" "128" "129" "130" "131" "132"
## [133] "133" "134" "135" "136" "137" "138" "139" "140" "141" "142" "143" "144"
## [145] "145" "146" "147" "148" "149" "150" "151" "152" "153" "154" "155" "156"
## [157] "157" "158" "159" "160" "161" "162" "163" "164" "165" "166" "167" "168"
## [169] "169" "170" "171" "172" "173" "174" "175" "176" "177" "178" "179" "180"
## [181] "181" "182" "183" "184" "185" "186" "187" "188"
```

3.2

Now add a column called `doc_section`. For the first 100 rows, the value of this column should be "Section 1". For documents 101-188, the section should be "Section 2".

```
# add fake column for section
dtm$doc_section <- "NA"
dtm$doc_section[1:100] <- "Section 1"
dtm$doc_section[101:188] <- "Section 2"
dtm$doc_section <- as.factor(dtm$doc_section)

summary(dtm$doc_section)
```

```
## Section 1 Section 2
##       100        88
```

3.3

Export the dataframe as a csv.

II. Sentiment Analysis with Thriller

In this section you'll conduct sentiment analysis on the lyrics of Michael Jackson's Thriller album.

1. Comparing Songs on the Thriller Album

Road the code below to get started.

```
rm(list=ls())
library(tm)
thriller <- read.csv("thriller.csv")
```

1.1

First preprocess the corpus. Create a document-term matrix from the `Lyrics` column of the `thriller` data frame. Complete the following preprocessing steps:

- convert to lower
- remove stop words
- remove numbers
- remove punctuation.

Think: Why is stemming inappropriate for this application?

```
# preprocess and create DTM
docs <- Corpus(VectorSource(thriller$Lyrics))

dtm <- DocumentTermMatrix(docs,
           control = list(tolower = TRUE,
                          removeNumbers = TRUE,
                          removePunctuation = TRUE,
                          stopwords = TRUE
                         ))

dtm <- as.data.frame(as.matrix(dtm))
```

2. Setting up the sentiment dictionary

2.1

We're going to use sentiment dictionaries from the `tidytext` package. Install and load the package.

```
library(tidytext)
```

2.2

Using the `get_sentiments` function, load the "bing" dictionary and store it in an object called `sent`. Take a look at the head of `sent`.

```
sent <- get_sentiments("bing")
head(sent)
```

```
## # A tibble: 6 x 2
##   word       sentiment
##   <chr>      <chr>
## 1 2-faces    negative
## 2 abnormal   negative
## 3 abolish    negative
## 4 abominable negative
## 5 abominably negative
## 6 abominate  negative
```

2.3

Add a column to `sent` called `score`. This column should hold a "1" for positive words and "-1" for negative words.

```
sent$score <- ifelse(sent$sentiment=="positive", 1, -1)
```

3. Scoring the Thriller album

3.1

Now you're ready to score each song.

(**NB**: There are probably many ways to program a script that performs this task. If you can think of a more elegant way, go for it!)

First, create a dataframe that holds all the words in our dtm along with their sentiment score.

```
# get all the words in our dtm and put it in a dataframe
words = data.frame(word = colnames(dtm))
head(words)
```

```
##      word
## 1  always
## 2    baby
## 3   babys
## 4  became
## 5 believe
## 6  billie
```

```
# get their sentiment scores
words <- merge(words, sent, all.x = T)
head(words)
```

```
##     word sentiment score
## 1 across      <NA>    NA
## 2 advice      <NA>    NA
## 3   aint      <NA>    NA
## 4    air      <NA>    NA
## 5  alien      <NA>    NA
## 6  alive      <NA>    NA
```

```
# replace NAs with 0s
words$score[is.na(words$score)] <- 0
head(words)
```

```
##     word sentiment score
## 1 across      <NA>     0
## 2 advice      <NA>     0
## 3   aint      <NA>     0
## 4    air      <NA>     0
## 5  alien      <NA>     0
## 6  alive      <NA>     0
```

3.2

Use matrix algebra to multiply the dtm by the scoring vector. This will return to us a score for each document (i.e., song).

Save the scores as a new column to `thriller`, called `sentiment`.

```r
# calculate documents scores with matrix algebra!
scores <- as.matrix(dtm) %*% words$score

# put it in the original documents data frame
thriller$sentiment <- scores
```

Which song is happiest? Go listen to the song and see if you agree.

4. Making a function

4.1

Using the code written above, make a function that accepts 1) a vector of texts, and 2) a sentiment dictionary (i.e. a data frame with words and scores), and returns a vector of sentiment scores for each text. Test it out!

```r
sentiment_score <- function(texts, sent_dict){

# preprocess texts
  docs <- Corpus(VectorSource(texts))
  dtm <- DocumentTermMatrix(docs,
           control = list(stopwords = T,
                          tolower = TRUE,
                          removeNumbers = TRUE,
                          removePunctuation = TRUE))
  dtm <- as.data.frame(as.matrix(dtm))


  # get all the words in our dtm and put it in a dataframe
  words = data.frame(word = colnames(dtm))

  # get their sentiment scores
  words <- merge(words, sent_dict, all.x = T)

  # replace NAs with 0s
  words$score[is.na(words$score)] <- 0

  # calculate documents scores with matrix algebra!
  scores <- as.matrix(dtm) %*% words$score

  return(scores)

}

# test it out!
sentiment_score(thriller$Lyrics, sent)
```

```
##    [,1]
## 1    -8
## 2    -4
## 3    -3
## 4    -1
## 5    28
## 6    -9
## 7    -2
```

```
## 8   -16
## 9    24
## 10    3
```

4.2

Using the function you wrote above, score the Thriller album with the "afinn" dictionary. Compare the scores across the two different dictionaries.

```
# # first load the dictionary
# library(textdata)
# afinn <- get_sentiments("afinn")
# head(afinn)
# afinn$score <- afinn$value
#
# # then run the function
# sentiment_score(thriller$Lyrics, afinn)
```