Text mining in practice - Bag of Words - Common tm visuals

Digital Humanities DSC 105 Spring 2023

Marcus Birkenkrahe

March 30, 2023

README

- This lecture closely follows the DataCamp lesson "Text Mining with Bag-of-Words in R" by Ted Kwartler.
- Download and open the practice file `7_visuals_tm_practice.org` from GitHub to code along.
- In this lecture & practice:
  1. frequent terms with `tm` visualized using `barplot`
  2. frequent terms with `qdap` visualized using `plot`

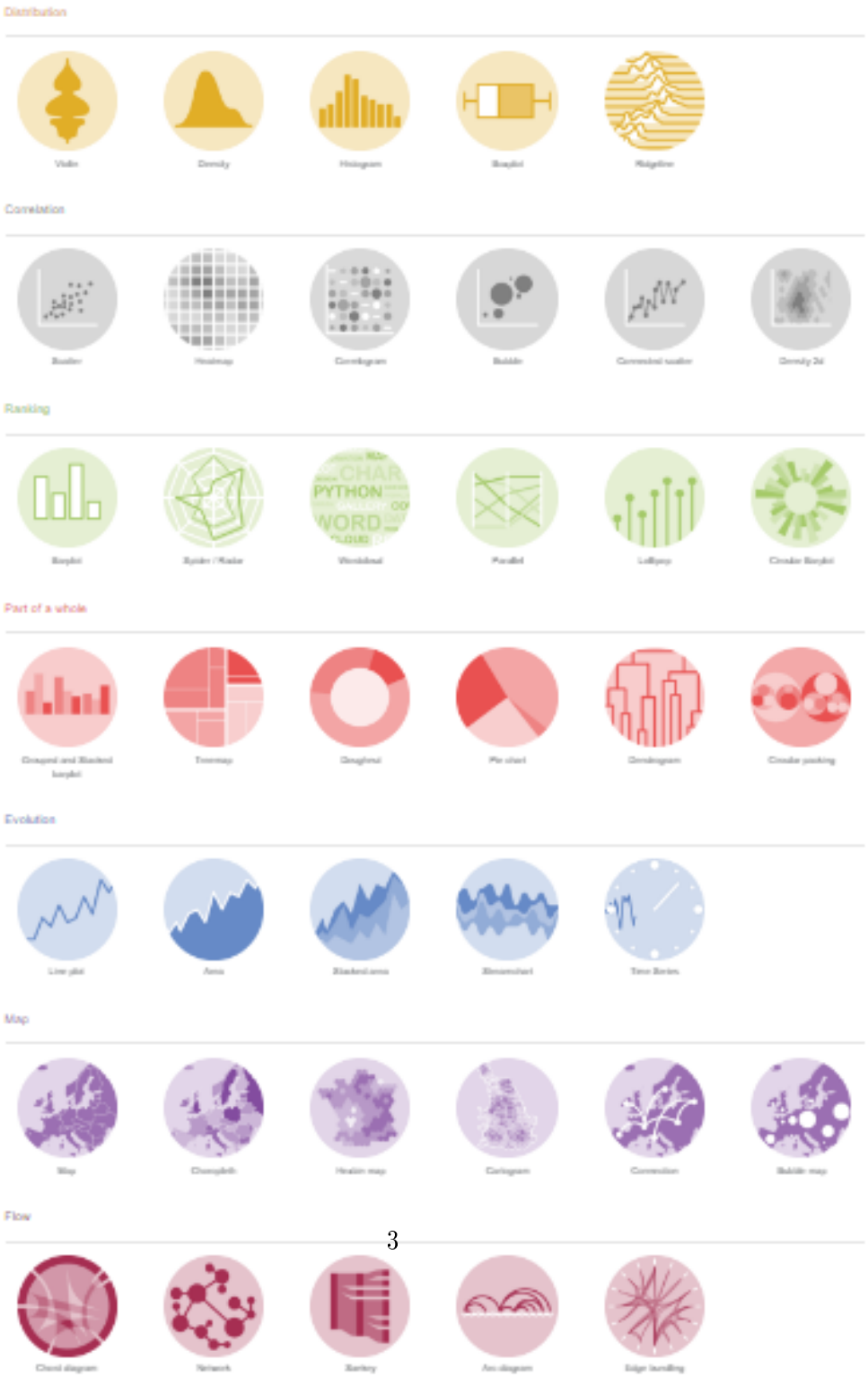
TODO Getting, loading, cleaning the corpus

- Download `corpora.org` from GitHub and run it: bit.ly/corpora_org
- Includes corpus creation, corpus cleaning and check printing

```
ls()
```

```
 [1] "api_key"                "ask_chatgpt"
 [3] "chardonnay_corpus"      "chardonnay_df"
 [5] "chardonnay_m"           "chardonnay_src"
 [7] "chardonnay_tdm"         "chardonnay_vec"
 [9] "clean_chardonnay"       "clean_chardonnay_corpus"
[11] "clean_coffee"           "clean_coffee_corpus"
[13] "coffee_corpus"          "coffee_df"
[15] "coffee_src"             "coffee_vec"
[17] "color_pal"              "load_packages"
[19] "M"                      "stops"
[21] "term_frequency"         "word_freq"
```

Importance of visuals



- Check out the R graph gallery, Kabacoff's Data visualization with R, visualization in base R by Crawford, or DataCamp's Intro to ggplot2
- For text mining, no need to invest too much into this at the start

Know your data

- Check that you have the coffee tweet corpus data loaded in your R session:

```
ls()
```

```
 [1] "api_key"                "ask_chatgpt"
 [3] "chardonnay_corpus"      "chardonnay_df"
 [5] "chardonnay_m"           "chardonnay_src"
 [7] "chardonnay_tdm"         "chardonnay_vec"
 [9] "clean_chardonnay"       "clean_chardonnay_corpus"
[11] "clean_coffee"           "clean_coffee_corpus"
[13] "coffee_corpus"          "coffee_df"
[15] "coffee_src"             "coffee_vec"
[17] "color_pal"              "load_packages"
[19] "M"                      "stops"
[21] "term_frequency"         "word_freq"
```

- The output should mean something to you - you should know how to:
  1. create these objects
  2. identify their nature (type), size and structure

```
:  [1] "chardonnay_corpus"       "chardonnay_df"
:  [3] "chardonnay_m"            "chardonnay_src"
:  [5] "chardonnay_vec"          "clean_chardonnay"
:  [7] "clean_chardonnay_corpus" "clean_coffee"
:  [9] "clean_coffee_corpus"     "coffee_corpus"
: [11] "coffee_df"               "coffee_src"
: [13] "coffee_vec"
```

- Most important for frequency computation is the stopwords cleaning step, because irrelevant terms will contaminate the desired data.

- If you don't remember, check the file `corpora.org` that contains the steps for creating and identifying the corpora and the data leading up to them.

Understand frequency terms

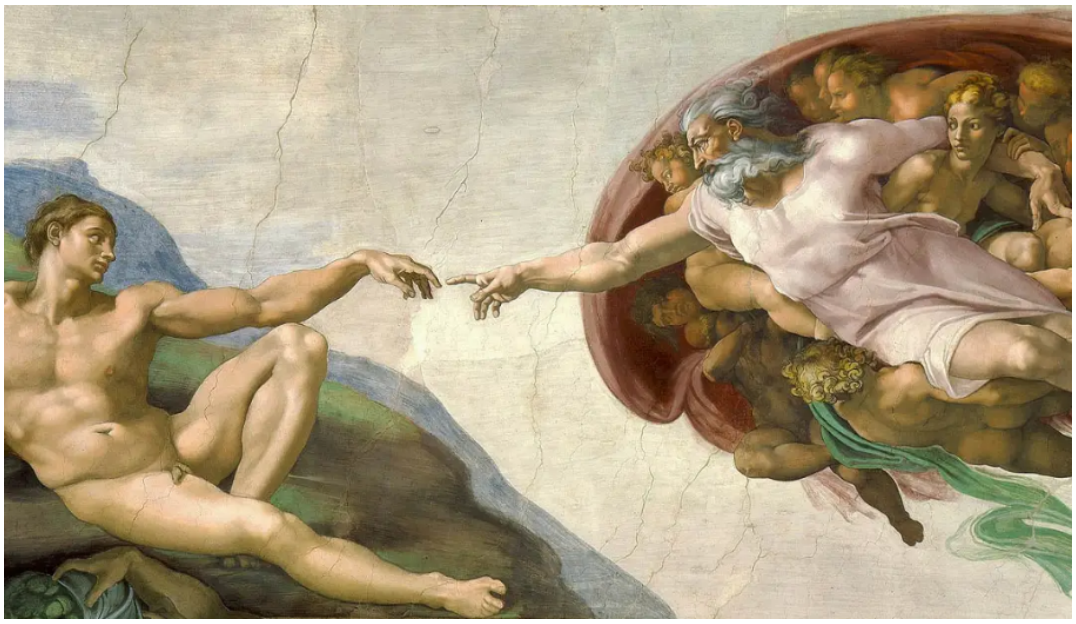


Figure 1: Michelangelo, Creation of Adam (1512)

- The purpose of frequency computation: to count certain words across the entire corpus, e.g. a collection of tweets.
- Different packages offer different methods to do this:
  1. the `tm` package relies on TDM analysis from a cleaned corpus
  2. the `qdap` package works straight from text
- Visualization offers a quick overview of the frequencies. Using the `wordcloud` package, we will explore:
  1. barcharts
  2. word clouds (size corresponds to frequency)

3. comparison clouds (compare word clouds)
4. pyramid plots (comparison bar charts)
5. word networks (association plots revealing connections)

- The interesting part of visualization (apart from the technical skill) is the interpretation and the **storytelling** that emerges
- Even Michelangelo's "Creation of Adam" (1512) can be seen as an example of text mining and word-based storytelling: can you see why?

Create frequent terms with `tm`

- Create the TDM for the coffee tweets, `coffee_tdm`, and display it:

```
coffee_tdm <- TermDocumentMatrix(clean_coffee)
coffee_tdm
```

```
<<TermDocumentMatrix (terms: 3035, documents: 1000)>>
Non-/sparse entries: 7304/3027696
Sparsity           : 100%
Maximal term length: 27
Weighting          : term frequency (tf)
```

- Convert the TDM to a matrix `coffee_m` and show its dimension and the terms 888 through 893, and documents 895 through 900.

```
coffee_m <- as.matrix(coffee_tdm)
dim(coffee_m)
coffee_m[888:893,895:900]
```

```
[1] 3035 1000
           Docs
Terms       895 896 897 898 899 900
  finds       0   0   0   0   0   0
  fine        0   0   0   0   0   0
  finger      0   0   0   0   0   0
  finish      0   0   0   0   0   0
  finished    0   0   0   0   0   0
  fireflies   0   0   0   0   0   0
```

- To get the frequency of each term (row) in all docs, we sum their occurrences across each row using `rowSums`:

```
m <- matrix(data=1:6, nrow=2, byrow=TRUE)
m
rowSums(m)
```

```
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[1]  6 15
```

- Run `rowSums` on `coffee_m` to get the `term_frequency`:

```
term_frequency <- rowSums(coffee_m)
```

Explore `term_frequency`

- Run the code again but this time add a check for the data structure with one of the `is.` functions, e.g. `is.matrix`, or `is.vector`.
- Pipe the result of the summing into `is.vector` and then into `print`:

```
rowSums(coffee_m) |> is.vector() |> print()
```

```
[1] TRUE
```

- Here is the nested version (without saving the result): it's `TRUE`!

```
is.vector(rowSums(coffee_m))
```

```
[1] TRUE
```

- Look at the first few items of the vector:

```
head(term_frequency)
term_frequency[1:6]
```

```
abasc abbslovesfed   abbycastro          abc    abccarpet        abdul
    1            1            1            1            1            1
abasc abbslovesfed   abbycastro          abc    abccarpet        abdul
    1            1            1            1            1            1
```

- Which term occurs most often and how many times in the tweets?
  1. check the `max`
  2. use `which` to get at the vector index and the name

```
max <- max(term_frequency)  # number of occurrences in 1000 tweets
max
idx <- which(term_frequency==max)  # returns the index
idx
term_frequency[idx]
names(term_frequency[1708])
```

```
[1] 111
like
1669
like
 111
[1] "look"
```

Order `term_frequency` values

- You can see that it needs to be sorted by frequency to be of any use so that the most frequent terms appear at the top:
  1. `sort` the `term_frequency`
  2. print the `head` of the result

```
sort(term_frequency) |> head()
head(sort(term_frequency))
```

```
abasc abbslovesfed   abbycastro          abc    abccarpet        abdul
    1            1            1            1            1            1
abasc abbslovesfed   abbycastro          abc    abccarpet        abdul
    1            1            1            1            1            1
```

- This didn't seem to have worked. What did we forget? Check `sort`:

```
sort(c(100,2,40,1000))
```

```
[1]    2   40  100 1000
```

- Check the arguments of `sort`:

```
args(sort)
```

```
function (x, decreasing = FALSE, ...)
NULL
```

- Now fix the `sort` of `term_frequency` and print the `head` again:

```
head(sort(term_frequency, decreasing = TRUE))
```

```
like     cup    shop    just     get morning
 111     103      69      66      62      57
```

- Overwrite `term_frequency` with its sorted version, and save the top 10 most common words to `term` using the index operator `[ ]`:

```
term_frequency <- sort(term_frequency, decreasing = TRUE)
terms <- term_frequency[1:10]
terms
names(terms)
```

```
   like      cup     shop     just      get  morning     want drinking
    111      103       69       66       62       57       49       47
    can    looks
     45       45
[1] "like"     "cup"      "shop"     "just"     "get"      "morning"
[7] "want"     "drinking" "can"      "looks"
```

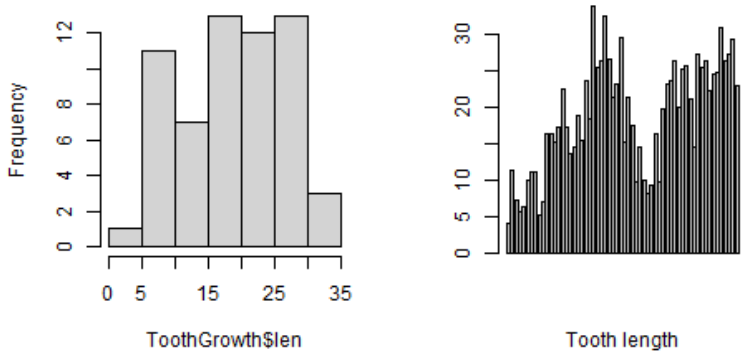
Make a barchart of 10 most frequent words

- To make a barchart of the top 10 most frequent words, we use `barplot`, a built-in base R function.
- Though `barplot` only needs one argument, `height`, the `help` reveals plenty of additional parameters:

```
barplot(height, width = 1, space = NULL,
        names.arg = NULL, legend.text = NULL, beside = FALSE,
        horiz = FALSE, density = NULL, angle = 45,
        col = NULL, border = par("fg"),
        main = NULL, sub = NULL, xlab = NULL, ylab = NULL,
        xlim = NULL, ylim = NULL, xpd = TRUE, log = "",
        axes = TRUE, axisnames = TRUE,
        cex.axis = par("cex.axis"), cex.names = par("cex.axis"),
        inside = TRUE, plot = TRUE, axis.lty = 0, offset = 0,
        add = FALSE, ann = !add && par("ann"), args.legend = NULL, ...)
```

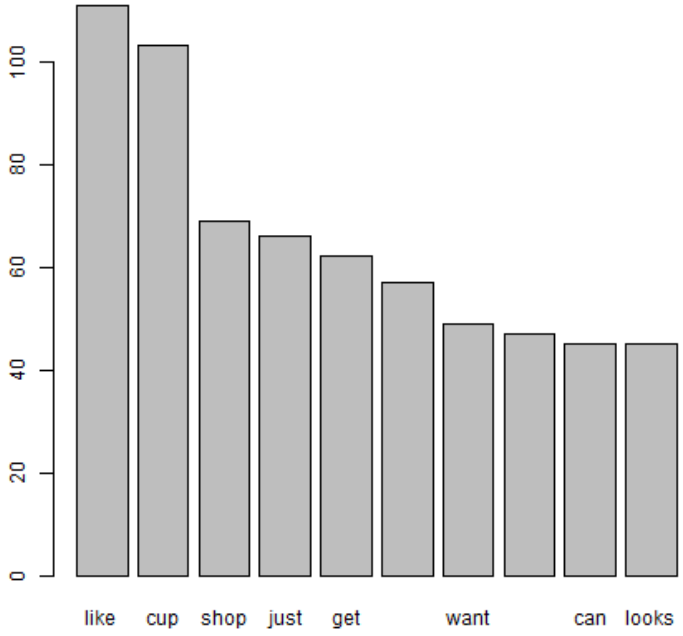
- The mandatory argument `height` is a vector or a matrix of values for the bars. If it's a vector, we get bars for each value, if it's a matrix, the values are stacked or dodged to account for the additional dimension.
- The `barplot` is similar to a histogram, the difference is that the histogram requires `numeric` x-values as input:

```
par(mfrow=c(1,2),pty='s')
hist(x=ToothGrowth$len,main="") ## histogram of the len variable
## barplot of the len variable
barplot(height=ToothGrowth$len, xlab="Tooth length")
```



- Plot a barchart of the 10 most common words `terms` with `barplot`:

```
barplot(height = terms)
```

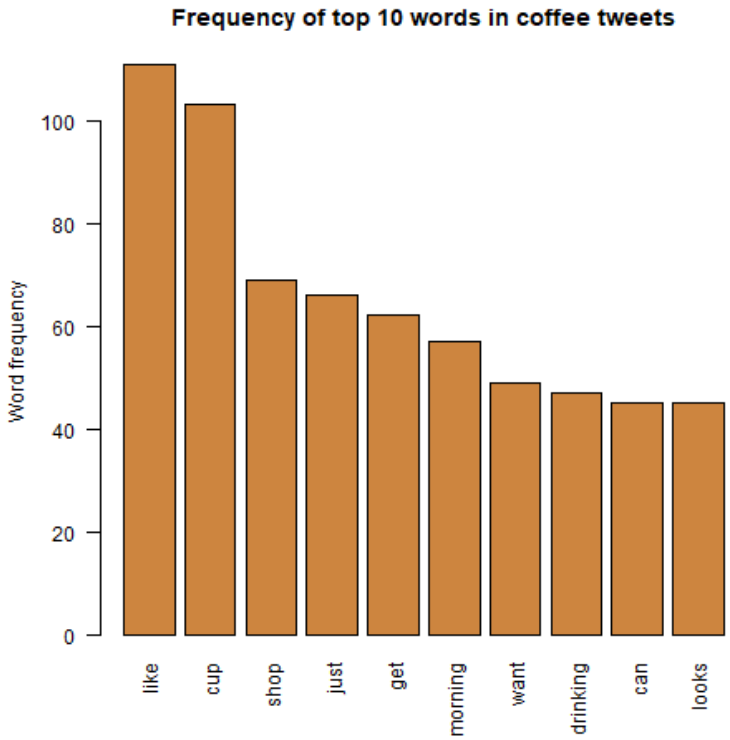


- The result is not wholly satisfying. Some labels don't show up because the words are too long. Let's customize a little:

  1. tilt the x-axis labels with `las=2`,
  2. add y-axis label title to explain the numbers
  3. add a title to the chart with `main`
  4. add a dash of color with `col` (e.g. "steelblue")

- Make these changes one after the other so that you can see the effects more clearly:

```
barplot(terms,las=2,
  main="Frequency of top 10 words in coffee tweets",
ylab="Word frequency",
col="peru")
```



- What are the available colors in R?

```
colors()  # available colors in a vector
```

```
  [1] "white"                "aliceblue"            "antiquewhite"
  [4] "antiquewhite1"        "antiquewhite2"        "antiquewhite3"
  [7] "antiquewhite4"        "aquamarine"           "aquamarine1"
 [10] "aquamarine2"          "aquamarine3"          "aquamarine4"
 [13] "azure"                "azure1"               "azure2"
 [16] "azure3"               "azure4"               "beige"
 [19] "bisque"               "bisque1"              "bisque2"
 [22] "bisque3"              "bisque4"              "black"
 [25] "blanchedalmond"       "blue"                 "blue1"
 [28] "blue2"                "blue3"                "blue4"
 [31] "blueviolet"           "brown"                "brown1"
 [34] "brown2"               "brown3"               "brown4"
```

```
 [37] "burlywood"           "burlywood1"          "burlywood2"
 [40] "burlywood3"          "burlywood4"          "cadetblue"
 [43] "cadetblue1"          "cadetblue2"          "cadetblue3"
 [46] "cadetblue4"          "chartreuse"          "chartreuse1"
 [49] "chartreuse2"         "chartreuse3"         "chartreuse4"
 [52] "chocolate"           "chocolate1"          "chocolate2"
 [55] "chocolate3"          "chocolate4"          "coral"
 [58] "coral1"              "coral2"              "coral3"
 [61] "coral4"              "cornflowerblue"      "cornsilk"
 [64] "cornsilk1"           "cornsilk2"           "cornsilk3"
 [67] "cornsilk4"           "cyan"                "cyan1"
 [70] "cyan2"               "cyan3"               "cyan4"
 [73] "darkblue"            "darkcyan"            "darkgoldenrod"
 [76] "darkgoldenrod1"      "darkgoldenrod2"      "darkgoldenrod3"
 [79] "darkgoldenrod4"      "darkgray"            "darkgreen"
 [82] "darkgrey"            "darkkhaki"           "darkmagenta"
 [85] "darkolivegreen"      "darkolivegreen1"     "darkolivegreen2"
 [88] "darkolivegreen3"     "darkolivegreen4"     "darkorange"
 [91] "darkorange1"         "darkorange2"         "darkorange3"
 [94] "darkorange4"         "darkorchid"          "darkorchid1"
 [97] "darkorchid2"         "darkorchid3"         "darkorchid4"
[100] "darkred"             "darksalmon"          "darkseagreen"
[103] "darkseagreen1"       "darkseagreen2"       "darkseagreen3"
[106] "darkseagreen4"       "darkslateblue"       "darkslategray"
[109] "darkslategray1"      "darkslategray2"      "darkslategray3"
[112] "darkslategray4"      "darkslategrey"       "darkturquoise"
[115] "darkviolet"          "deeppink"            "deeppink1"
[118] "deeppink2"           "deeppink3"           "deeppink4"
[121] "deepskyblue"         "deepskyblue1"        "deepskyblue2"
[124] "deepskyblue3"        "deepskyblue4"        "dimgray"
[127] "dimgrey"             "dodgerblue"          "dodgerblue1"
[130] "dodgerblue2"         "dodgerblue3"         "dodgerblue4"
[133] "firebrick"           "firebrick1"          "firebrick2"
[136] "firebrick3"          "firebrick4"          "floralwhite"
[139] "forestgreen"         "gainsboro"           "ghostwhite"
[142] "gold"                "gold1"               "gold2"
[145] "gold3"               "gold4"               "goldenrod"
[148] "goldenrod1"          "goldenrod2"          "goldenrod3"
[151] "goldenrod4"          "gray"                "gray0"
[154] "gray1"               "gray2"               "gray3"
```

```
[157] "gray4"          "gray5"          "gray6"
[160] "gray7"          "gray8"          "gray9"
[163] "gray10"         "gray11"         "gray12"
[166] "gray13"         "gray14"         "gray15"
[169] "gray16"         "gray17"         "gray18"
[172] "gray19"         "gray20"         "gray21"
[175] "gray22"         "gray23"         "gray24"
[178] "gray25"         "gray26"         "gray27"
[181] "gray28"         "gray29"         "gray30"
[184] "gray31"         "gray32"         "gray33"
[187] "gray34"         "gray35"         "gray36"
[190] "gray37"         "gray38"         "gray39"
[193] "gray40"         "gray41"         "gray42"
[196] "gray43"         "gray44"         "gray45"
[199] "gray46"         "gray47"         "gray48"
[202] "gray49"         "gray50"         "gray51"
[205] "gray52"         "gray53"         "gray54"
[208] "gray55"         "gray56"         "gray57"
[211] "gray58"         "gray59"         "gray60"
[214] "gray61"         "gray62"         "gray63"
[217] "gray64"         "gray65"         "gray66"
[220] "gray67"         "gray68"         "gray69"
[223] "gray70"         "gray71"         "gray72"
[226] "gray73"         "gray74"         "gray75"
[229] "gray76"         "gray77"         "gray78"
[232] "gray79"         "gray80"         "gray81"
[235] "gray82"         "gray83"         "gray84"
[238] "gray85"         "gray86"         "gray87"
[241] "gray88"         "gray89"         "gray90"
[244] "gray91"         "gray92"         "gray93"
[247] "gray94"         "gray95"         "gray96"
[250] "gray97"         "gray98"         "gray99"
[253] "gray100"        "green"          "green1"
[256] "green2"         "green3"         "green4"
[259] "greenyellow"    "grey"           "grey0"
[262] "grey1"          "grey2"          "grey3"
[265] "grey4"          "grey5"          "grey6"
[268] "grey7"          "grey8"          "grey9"
[271] "grey10"         "grey11"         "grey12"
[274] "grey13"         "grey14"         "grey15"
```

```
[277] "grey16"           "grey17"           "grey18"
[280] "grey19"           "grey20"           "grey21"
[283] "grey22"           "grey23"           "grey24"
[286] "grey25"           "grey26"           "grey27"
[289] "grey28"           "grey29"           "grey30"
[292] "grey31"           "grey32"           "grey33"
[295] "grey34"           "grey35"           "grey36"
[298] "grey37"           "grey38"           "grey39"
[301] "grey40"           "grey41"           "grey42"
[304] "grey43"           "grey44"           "grey45"
[307] "grey46"           "grey47"           "grey48"
[310] "grey49"           "grey50"           "grey51"
[313] "grey52"           "grey53"           "grey54"
[316] "grey55"           "grey56"           "grey57"
[319] "grey58"           "grey59"           "grey60"
[322] "grey61"           "grey62"           "grey63"
[325] "grey64"           "grey65"           "grey66"
[328] "grey67"           "grey68"           "grey69"
[331] "grey70"           "grey71"           "grey72"
[334] "grey73"           "grey74"           "grey75"
[337] "grey76"           "grey77"           "grey78"
[340] "grey79"           "grey80"           "grey81"
[343] "grey82"           "grey83"           "grey84"
[346] "grey85"           "grey86"           "grey87"
[349] "grey88"           "grey89"           "grey90"
[352] "grey91"           "grey92"           "grey93"
[355] "grey94"           "grey95"           "grey96"
[358] "grey97"           "grey98"           "grey99"
[361] "grey100"          "honeydew"         "honeydew1"
[364] "honeydew2"        "honeydew3"        "honeydew4"
[367] "hotpink"          "hotpink1"         "hotpink2"
[370] "hotpink3"         "hotpink4"         "indianred"
[373] "indianred1"       "indianred2"       "indianred3"
[376] "indianred4"       "ivory"            "ivory1"
[379] "ivory2"           "ivory3"           "ivory4"
[382] "khaki"            "khaki1"           "khaki2"
[385] "khaki3"           "khaki4"           "lavender"
[388] "lavenderblush"    "lavenderblush1"   "lavenderblush2"
[391] "lavenderblush3"   "lavenderblush4"   "lawngreen"
[394] "lemonchiffon"     "lemonchiffon1"    "lemonchiffon2"
```

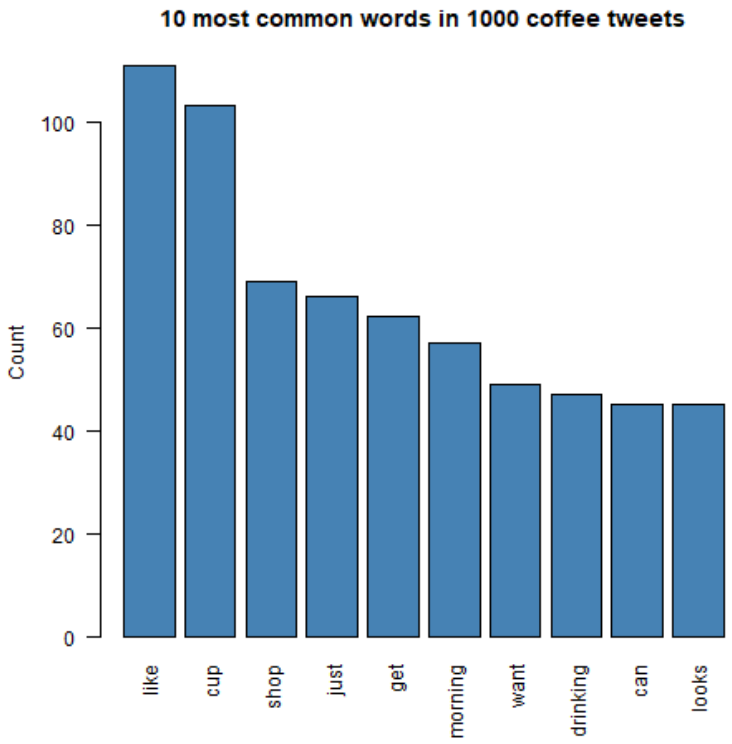
```
[397] "lemonchiffon3"        "lemonchiffon4"        "lightblue"
[400] "lightblue1"           "lightblue2"           "lightblue3"
[403] "lightblue4"           "lightcoral"           "lightcyan"
[406] "lightcyan1"           "lightcyan2"           "lightcyan3"
[409] "lightcyan4"           "lightgoldenrod"       "lightgoldenrod1"
[412] "lightgoldenrod2"      "lightgoldenrod3"      "lightgoldenrod4"
[415] "lightgoldenrodyellow" "lightgray"            "lightgreen"
[418] "lightgrey"            "lightpink"            "lightpink1"
[421] "lightpink2"           "lightpink3"           "lightpink4"
[424] "lightsalmon"          "lightsalmon1"         "lightsalmon2"
[427] "lightsalmon3"         "lightsalmon4"         "lightseagreen"
[430] "lightskyblue"         "lightskyblue1"        "lightskyblue2"
[433] "lightskyblue3"        "lightskyblue4"        "lightslateblue"
[436] "lightslategray"       "lightslategrey"       "lightsteelblue"
[439] "lightsteelblue1"      "lightsteelblue2"      "lightsteelblue3"
[442] "lightsteelblue4"      "lightyellow"          "lightyellow1"
[445] "lightyellow2"         "lightyellow3"         "lightyellow4"
[448] "limegreen"            "linen"                "magenta"
[451] "magenta1"             "magenta2"             "magenta3"
[454] "magenta4"             "maroon"               "maroon1"
[457] "maroon2"              "maroon3"              "maroon4"
[460] "mediumaquamarine"     "mediumblue"           "mediumorchid"
[463] "mediumorchid1"        "mediumorchid2"        "mediumorchid3"
[466] "mediumorchid4"        "mediumpurple"         "mediumpurple1"
[469] "mediumpurple2"        "mediumpurple3"        "mediumpurple4"
[472] "mediumseagreen"       "mediumslateblue"      "mediumspringgreen"
[475] "mediumturquoise"      "mediumvioletred"      "midnightblue"
[478] "mintcream"            "mistyrose"            "mistyrose1"
[481] "mistyrose2"           "mistyrose3"           "mistyrose4"
[484] "moccasin"             "navajowhite"          "navajowhite1"
[487] "navajowhite2"         "navajowhite3"         "navajowhite4"
[490] "navy"                 "navyblue"             "oldlace"
[493] "olivedrab"            "olivedrab1"           "olivedrab2"
[496] "olivedrab3"           "olivedrab4"           "orange"
[499] "orange1"              "orange2"              "orange3"
[502] "orange4"              "orangered"            "orangered1"
[505] "orangered2"           "orangered3"           "orangered4"
[508] "orchid"               "orchid1"              "orchid2"
[511] "orchid3"              "orchid4"              "palegoldenrod"
[514] "palegreen"            "palegreen1"           "palegreen2"
```

```
[517] "palegreen3"          "palegreen4"          "paleturquoise"
[520] "paleturquoise1"      "paleturquoise2"      "paleturquoise3"
[523] "paleturquoise4"      "palevioletred"       "palevioletred1"
[526] "palevioletred2"      "palevioletred3"      "palevioletred4"
[529] "papayawhip"          "peachpuff"           "peachpuff1"
[532] "peachpuff2"          "peachpuff3"          "peachpuff4"
[535] "peru"                "pink"                "pink1"
[538] "pink2"               "pink3"               "pink4"
[541] "plum"                "plum1"               "plum2"
[544] "plum3"               "plum4"               "powderblue"
[547] "purple"              "purple1"             "purple2"
[550] "purple3"             "purple4"             "red"
[553] "red1"                "red2"                "red3"
[556] "red4"                "rosybrown"           "rosybrown1"
[559] "rosybrown2"          "rosybrown3"          "rosybrown4"
[562] "royalblue"           "royalblue1"          "royalblue2"
[565] "royalblue3"          "royalblue4"          "saddlebrown"
[568] "salmon"              "salmon1"             "salmon2"
[571] "salmon3"             "salmon4"             "sandybrown"
[574] "seagreen"            "seagreen1"           "seagreen2"
[577] "seagreen3"           "seagreen4"           "seashell"
[580] "seashell1"           "seashell2"           "seashell3"
[583] "seashell4"           "sienna"              "sienna1"
[586] "sienna2"             "sienna3"             "sienna4"
[589] "skyblue"             "skyblue1"            "skyblue2"
[592] "skyblue3"            "skyblue4"            "slateblue"
[595] "slateblue1"          "slateblue2"          "slateblue3"
[598] "slateblue4"          "slategray"           "slategray1"
[601] "slategray2"          "slategray3"          "slategray4"
[604] "slategrey"           "snow"                "snow1"
[607] "snow2"               "snow3"               "snow4"
[610] "springgreen"         "springgreen1"        "springgreen2"
[613] "springgreen3"        "springgreen4"        "steelblue"
[616] "steelblue1"          "steelblue2"          "steelblue3"
[619] "steelblue4"          "tan"                 "tan1"
[622] "tan2"                "tan3"                "tan4"
[625] "thistle"             "thistle1"            "thistle2"
[628] "thistle3"            "thistle4"            "tomato"
[631] "tomato1"             "tomato2"             "tomato3"
[634] "tomato4"             "turquoise"           "turquoise1"
```

```
[637] "turquoise2"           "turquoise3"           "turquoise4"
[640] "violet"               "violetred"            "violetred1"
[643] "violetred2"           "violetred3"           "violetred4"
[646] "wheat"                "wheat1"               "wheat2"
[649] "wheat3"               "wheat4"               "whitesmoke"
[652] "yellow"               "yellow1"              "yellow2"
[655] "yellow3"              "yellow4"              "yellowgreen"
```

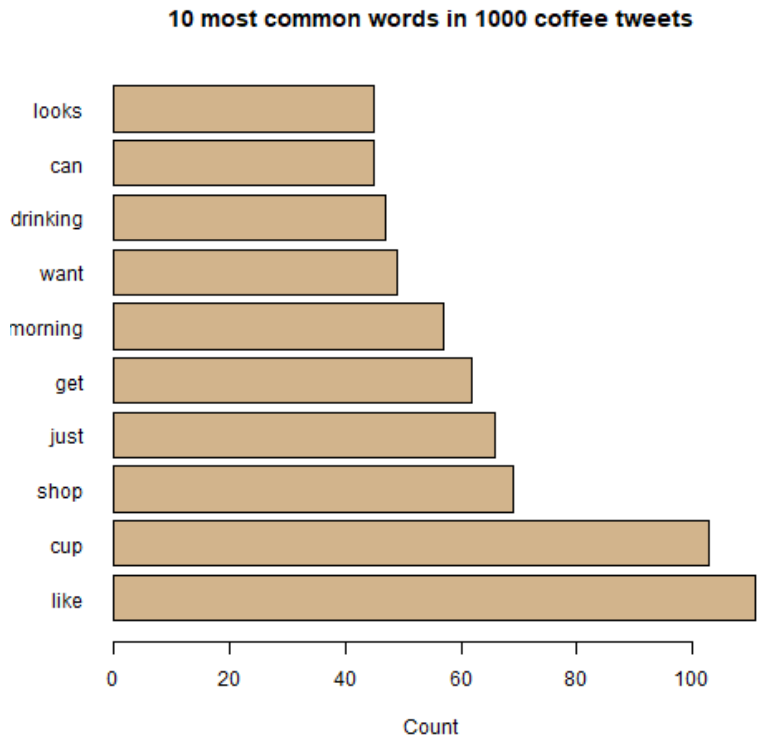
- Final result (for now):

```
barplot(height = terms,
        las=2,
        ylab="Count",
        main="10 most common words in 1000 coffee tweets",
        col="steelblue")
```



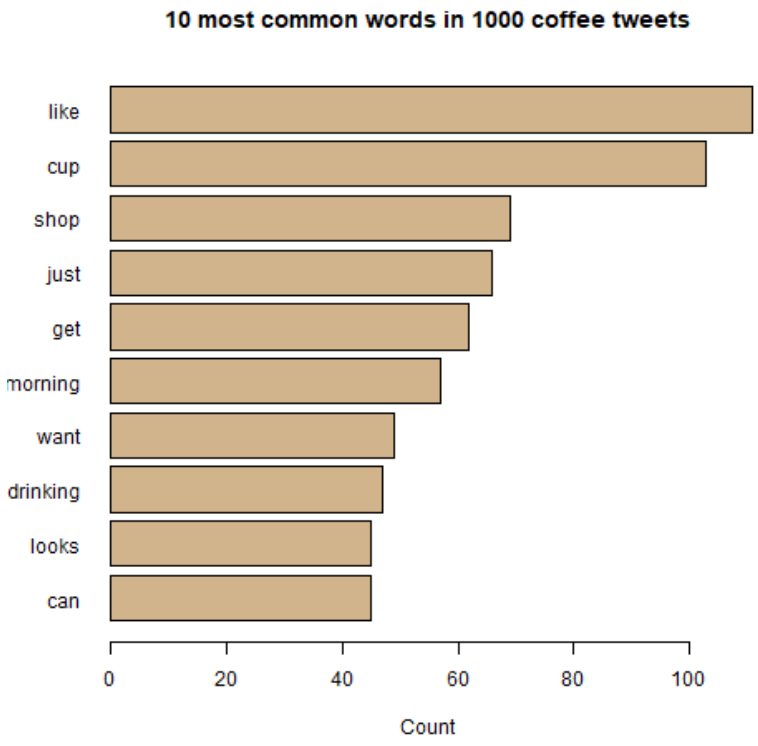
- To improve readability even more, tilt the graph to its side with `horiz=TRUE`, change `las` to `1` and `ylab` to `xlab`:

```
barplot(height = terms,
        las=1,   # tilt the count labels (x)
        xlab="Count",
        main="10 most common words in 1000 coffee tweets",
        col="tan",
        horiz=TRUE)
```



- Finally, reorder the y-axis values so that the most frequent term is at the top (and change `ylab` to :

```
barplot(height = sort(terms),
        las=1,
        xlab="Count",
        main="10 most common words in 1000 coffee tweets",
        col="tan",
        horiz=TRUE)
```


10 most common words in 1000 coffee tweets
like
cup
shop
just
get
morning
want
drinking
looks
can
0
20
40
60
80
100
Count