Common text mining visuals

TEXT MINING WITH BAG-OF-WORDS IN R

Ted Kwartler
Instructor

Why make visuals?

- Good visuals lead to quick conclusions
- The brain efficiently processes visual information

Setting the scene

Setting the scene

Term Document Matrix (TDM)

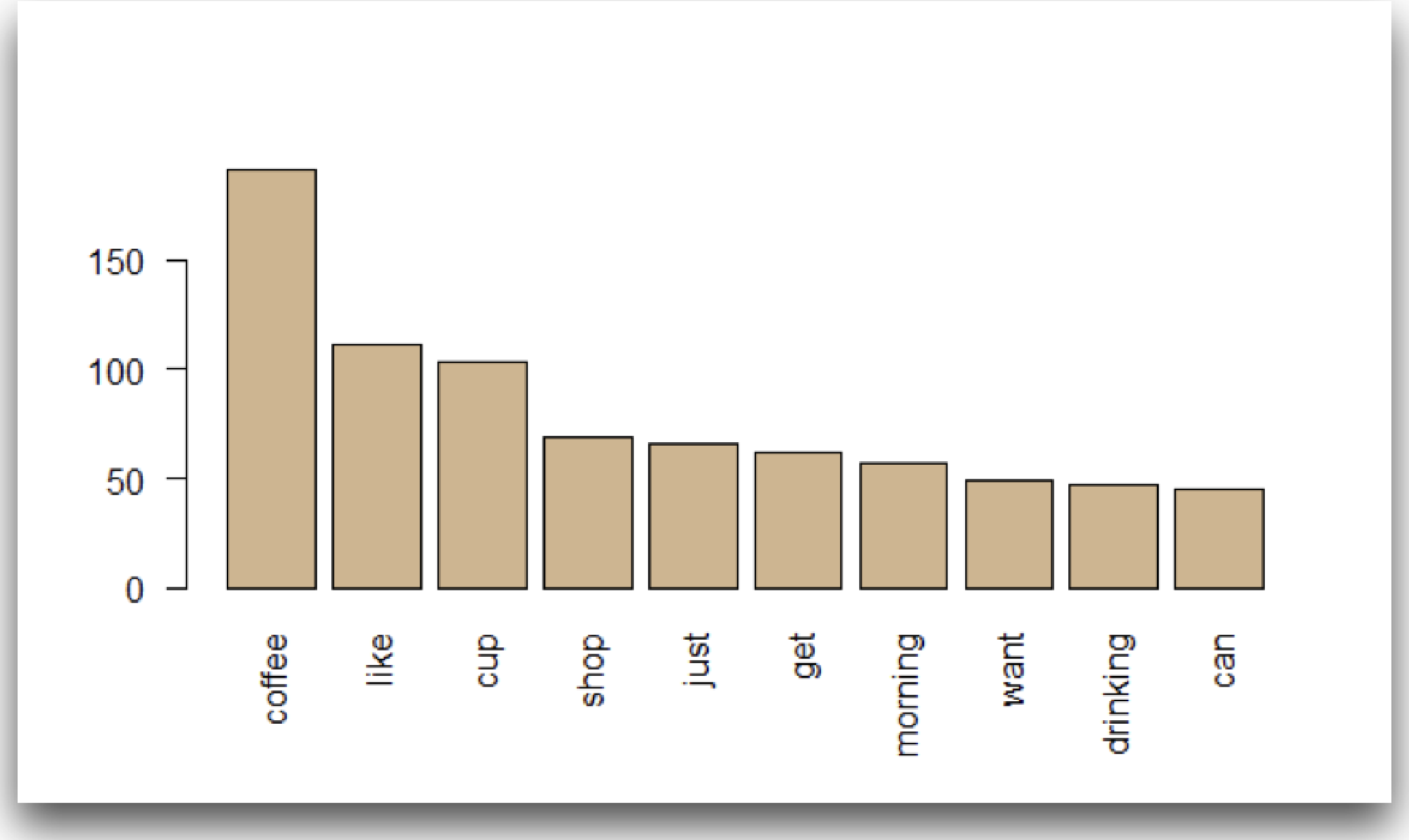
	Tweet1	Tweet2	Tweet3	…	Tweet_N
Term1	0	0	0	0	0
Term2	1	1	0	0	0
Term3	1	0	0	2	0
…	0	0	3	1	1
Term_N	0	0	1	1	0

Summed vector

Sum
0
2
3
5
2

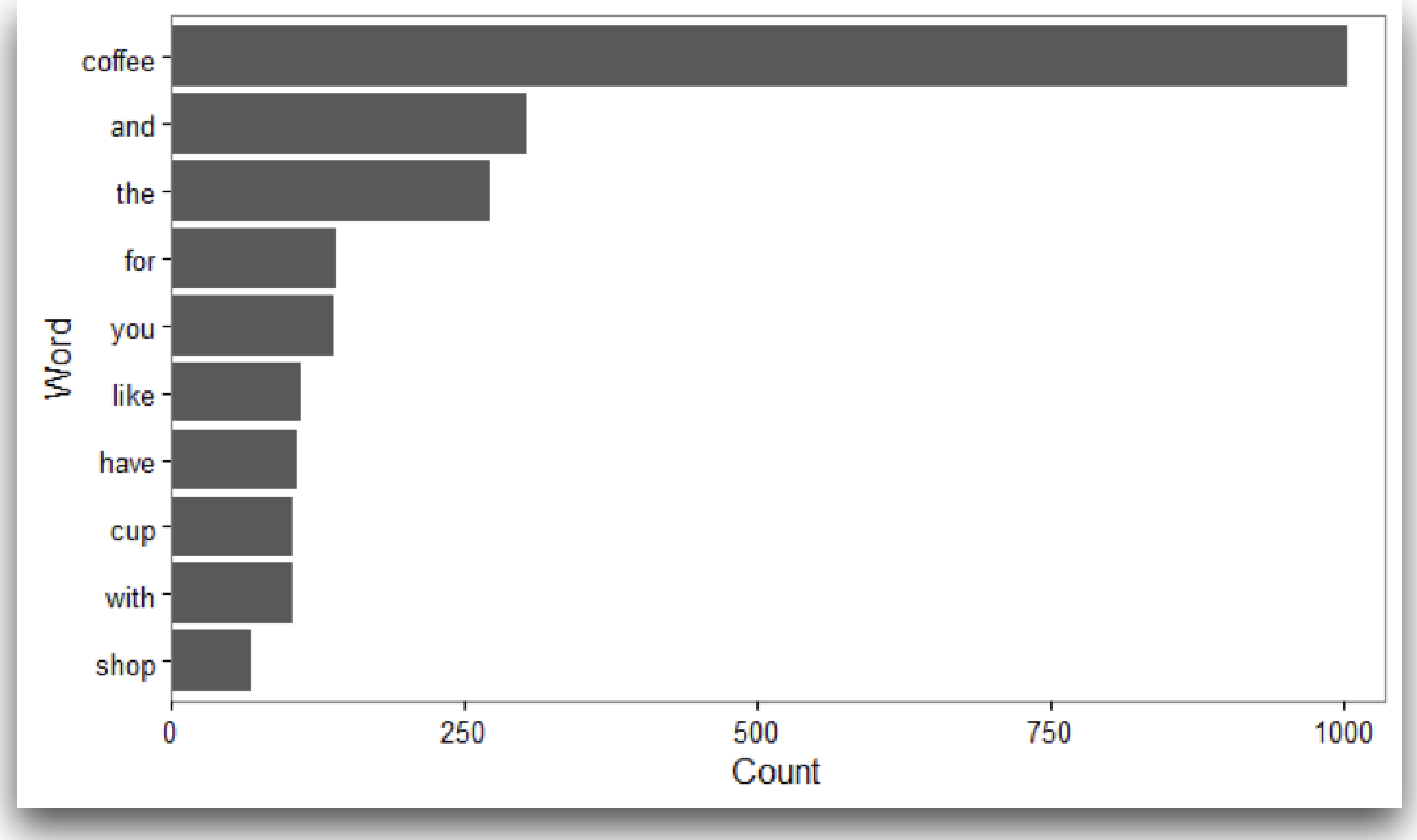
Term frequency plots with tm

```
# Convert TDM to matrix
coffee_m <- as.matrix(coffee_tdm)
# Sum rows and sort by frequency
term_frequency <- rowSums(coffee_m)
term_frequency <- sort(term_frequency,
                       decreasing = TRUE)
# Create a barplot
barplot(term_frequency[1:10],
        col = "tan",
        las = 2)
```



Term frequency plots with qdap

```
# Load qdap package
library(qdap)
# Find term frequencies
frequency <- freq_terms(
    tweets$text,
    top = 10,
    at.least = 3,
    stopwords = "Top200Words"
  )
# Plot term frequencies
plot(frequency)
```



Let's practice!

TEXT MINING WITH BAG-OF-WORDS IN R

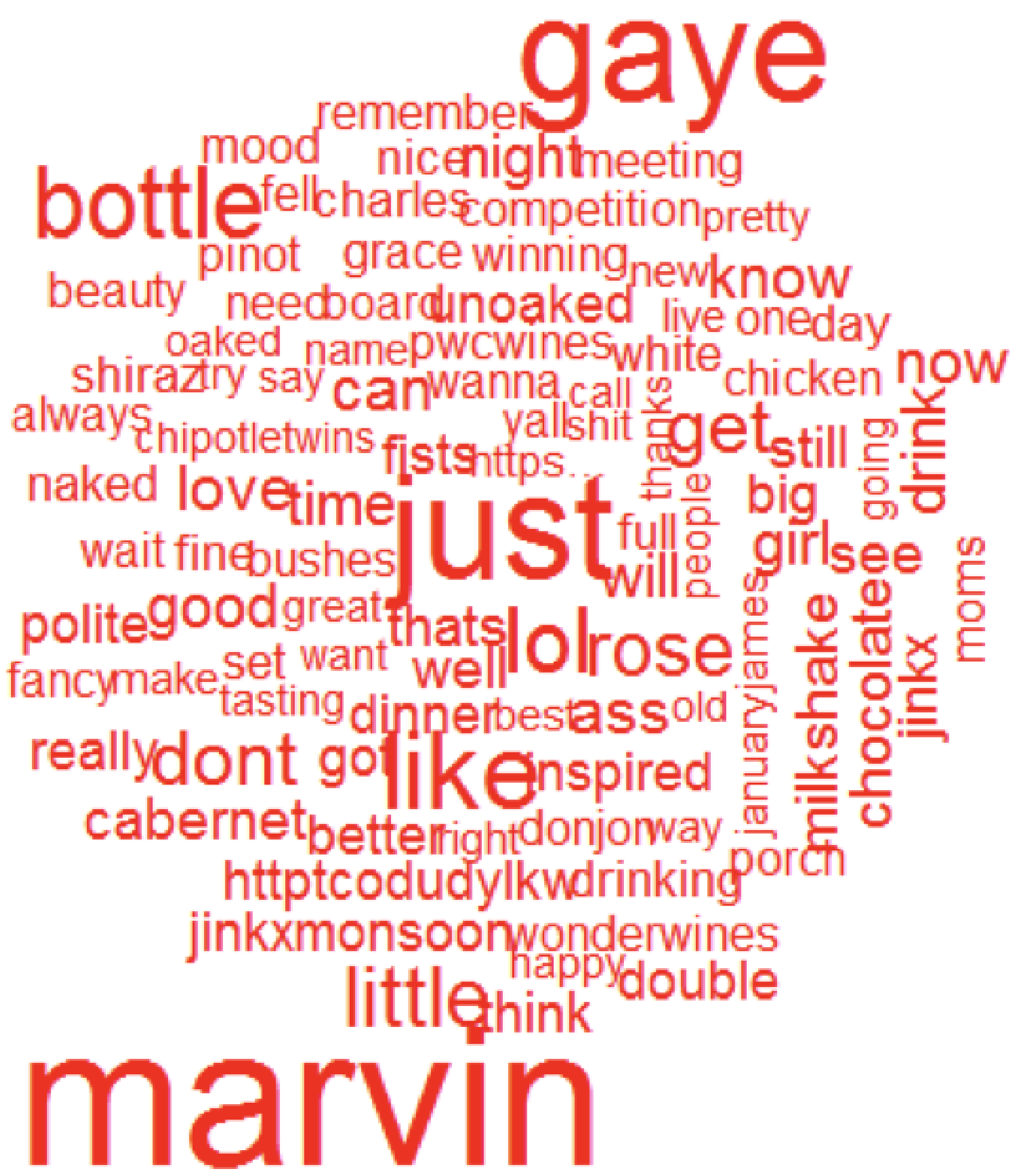
Intro to word clouds

TEXT MINING WITH BAG-OF-WORDS IN R

Ted Kwartler
Instructor

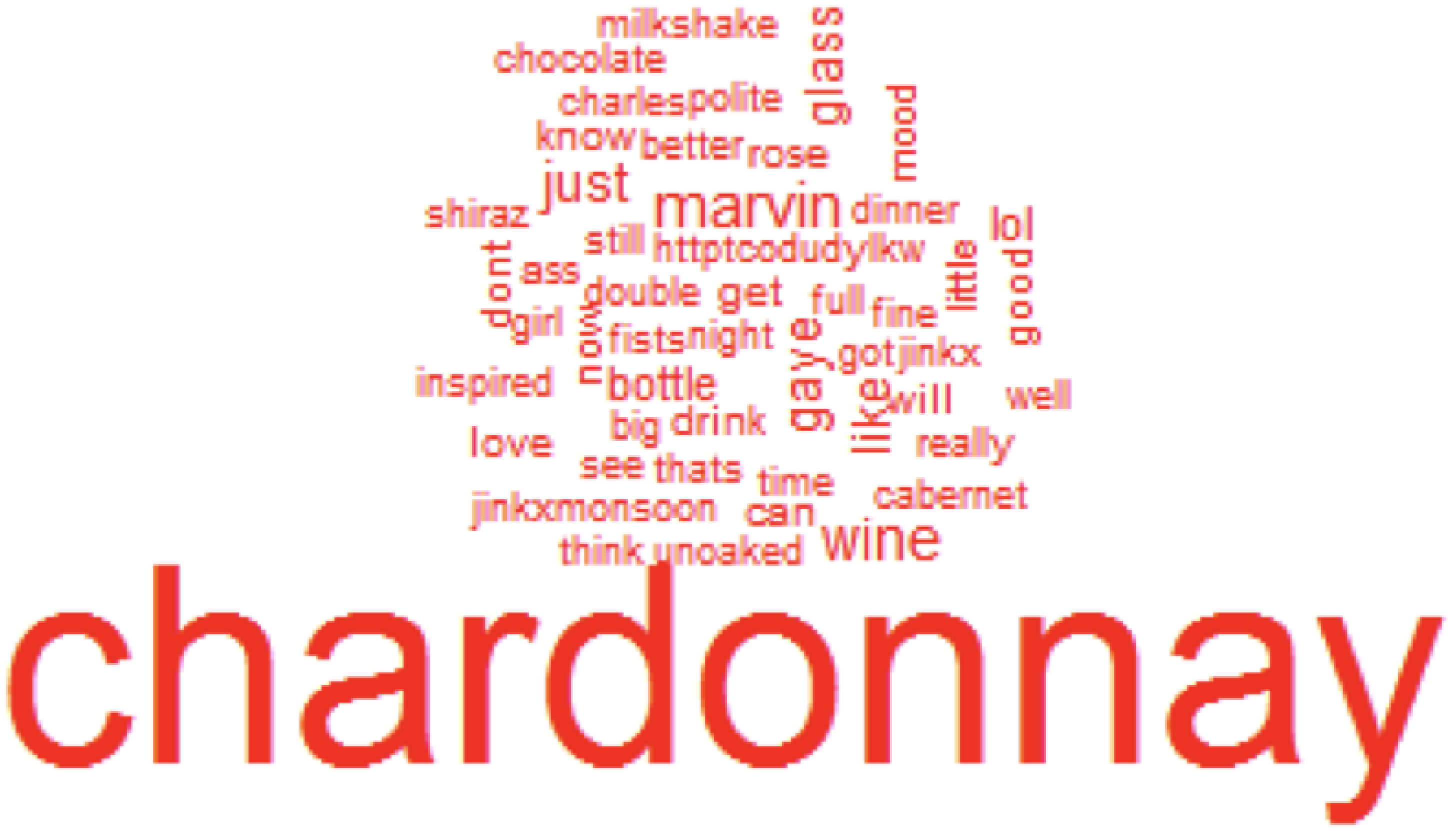
A simple word cloud

```
# Convert TDM to matrix
chardonnay_tdm <- TermDocumentMatrix(clean_chardonnay)
chardonnay_m <- as.matrix(chardonnay_tdm)

# Sum rows and sort by frequency
term_frequency <- rowSums(chardonnay_m)
word_freqs <- data.frame(term = names(term_frequency),
                         num = term_frequency)

# Make word cloud
wordcloud(word_freqs$term, word_freqs$num,
          max.words = 100, colors = "red")
```



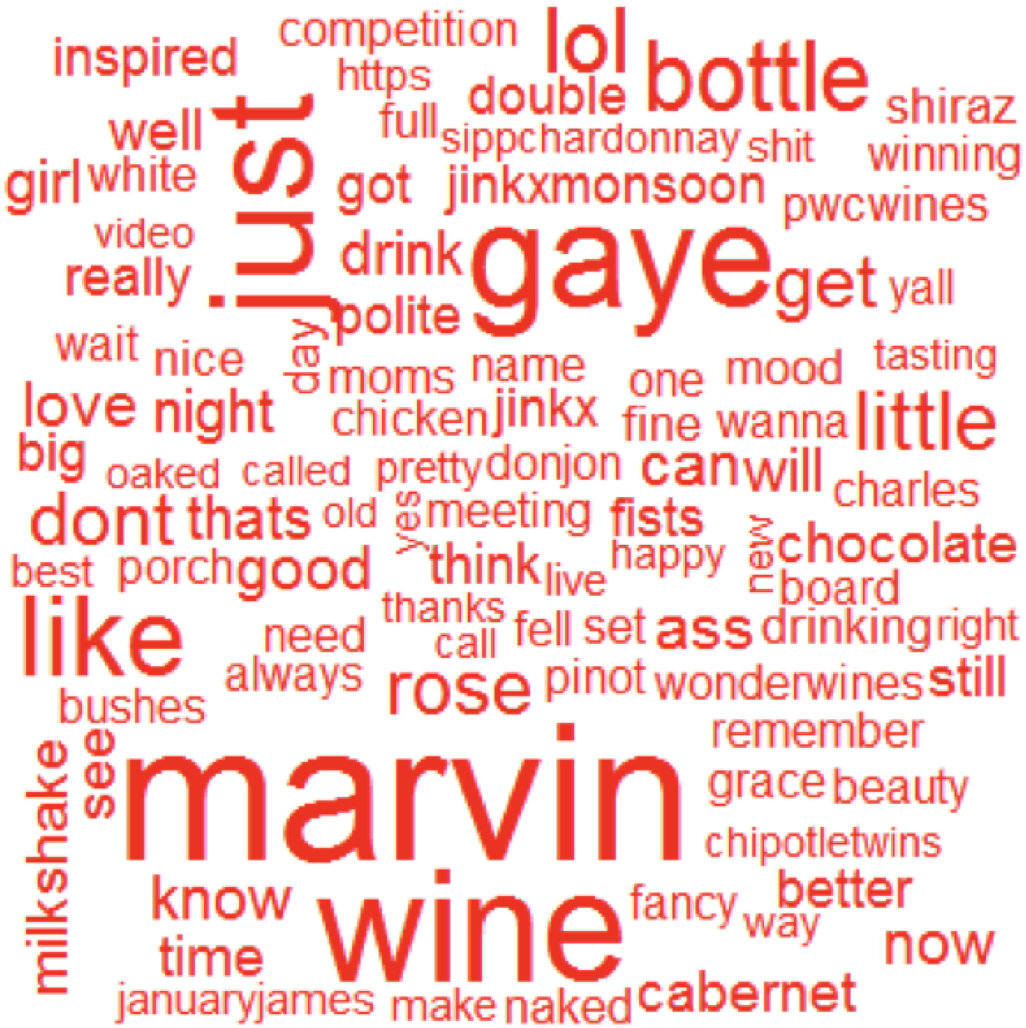
The impact of stop words

```r
clean_corpus <- function(corpus){
  corpus <- tm_map(corpus, removePunctuation)
  corpus <- tm_map(corpus, stripWhitespace)
  corpus <- tm_map(corpus, removeNumbers)
  corpus <- tm_map(corpus,
                   content_transformer(tolower))
  corpus <- tm_map(corpus, removeWords,
                   c(stopwords("en"), "amp"))
  return(corpus)
}
```



Removing uninformative words

```r
clean_corpus <- function(corpus){
  corpus <- tm_map(corpus, removePunctuation)
  corpus <- tm_map(corpus, stripWhitespace)
  corpus <- tm_map(corpus, removeNumbers)
  corpus <- tm_map(corpus,
                   content_transformer(tolower))
  corpus <- tm_map(corpus, removeWords,
                   c(stopwords("en"), "amp",
                    "chardonnay", "wine", "glass"))
  return(corpus)
}
```



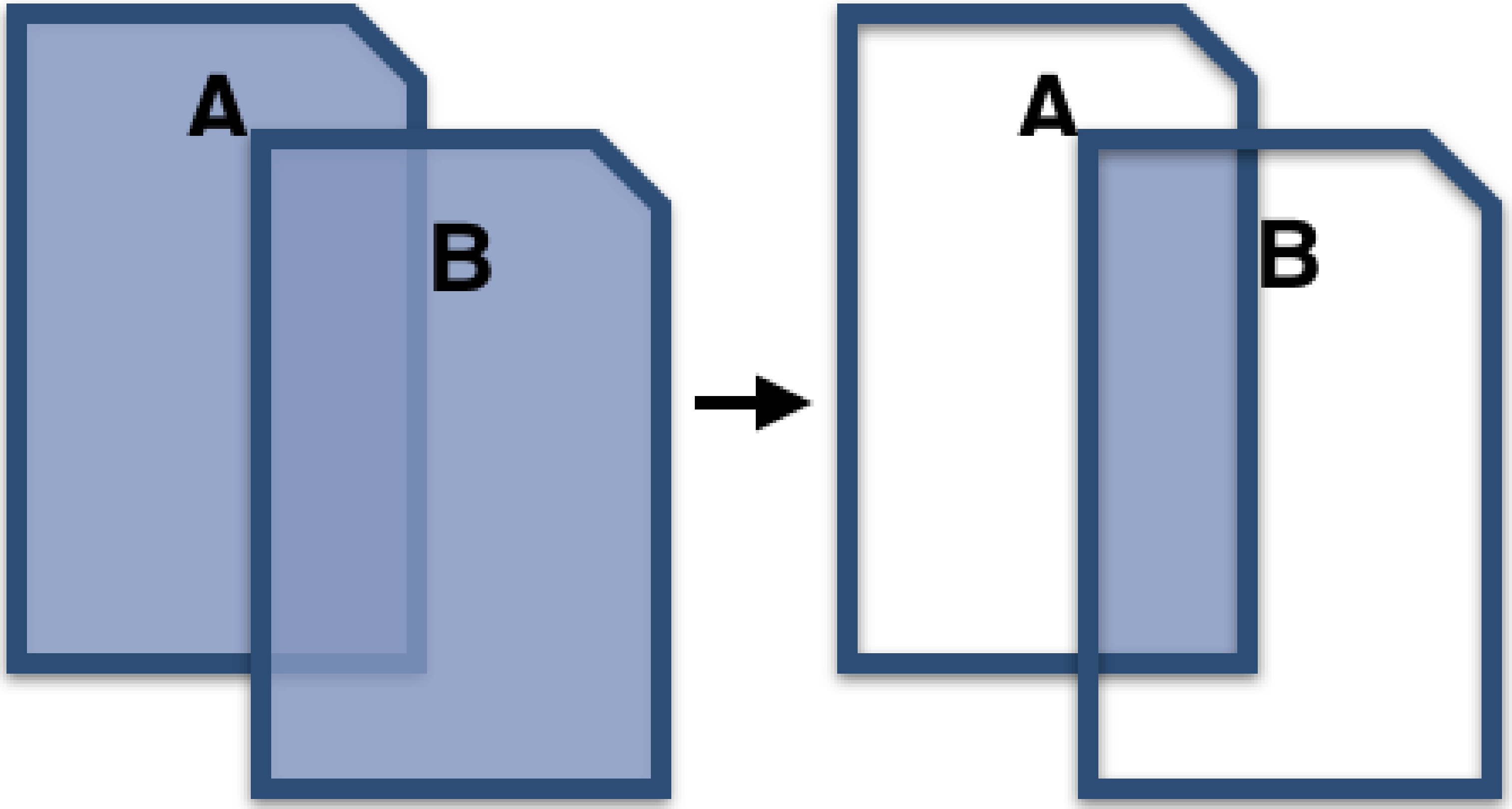
Let's practice!

Other word clouds and word networks

TEXT MINING WITH BAG-OF-WORDS IN R

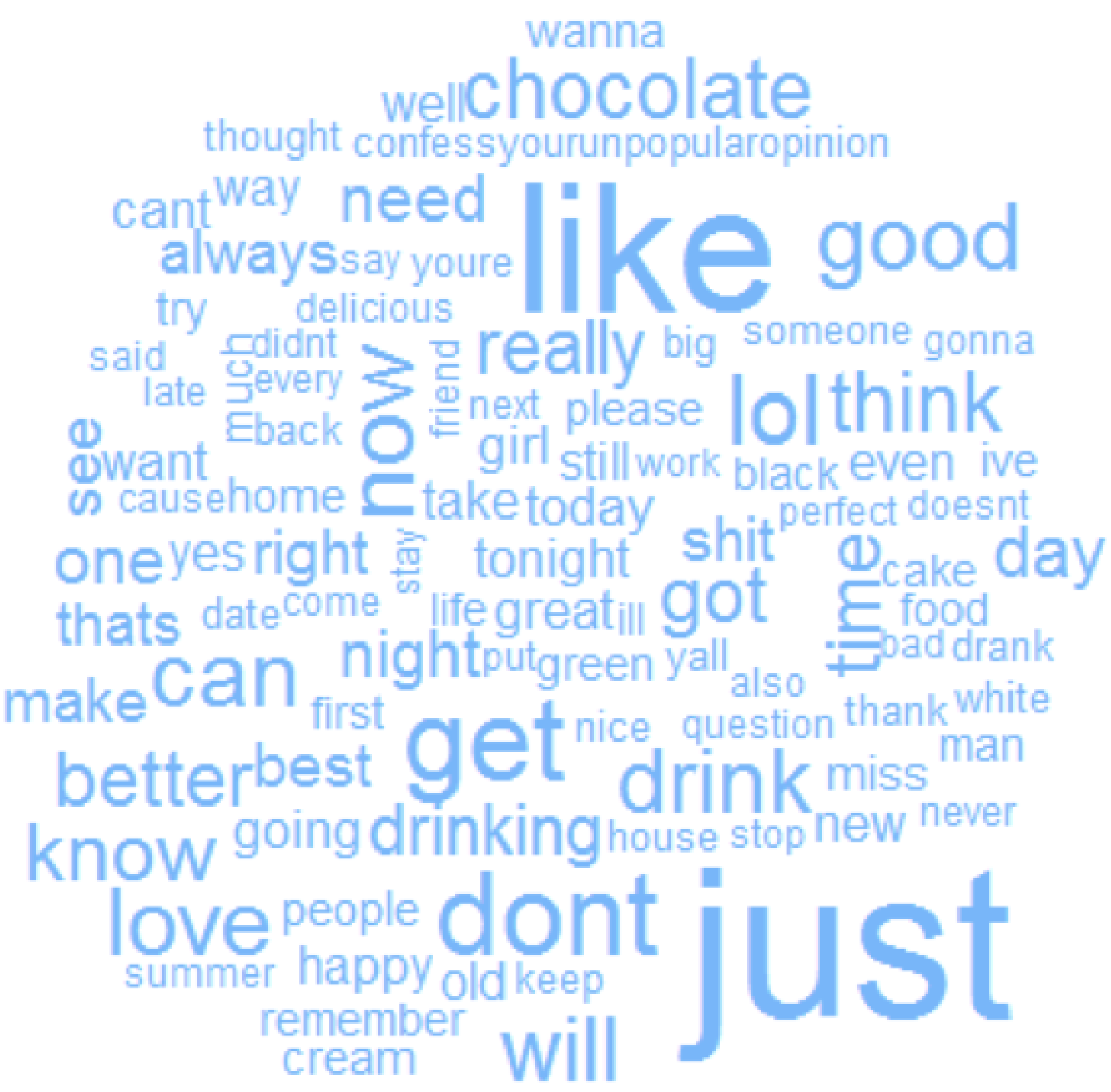
Ted Kwartler
Instructor

Commonality clouds

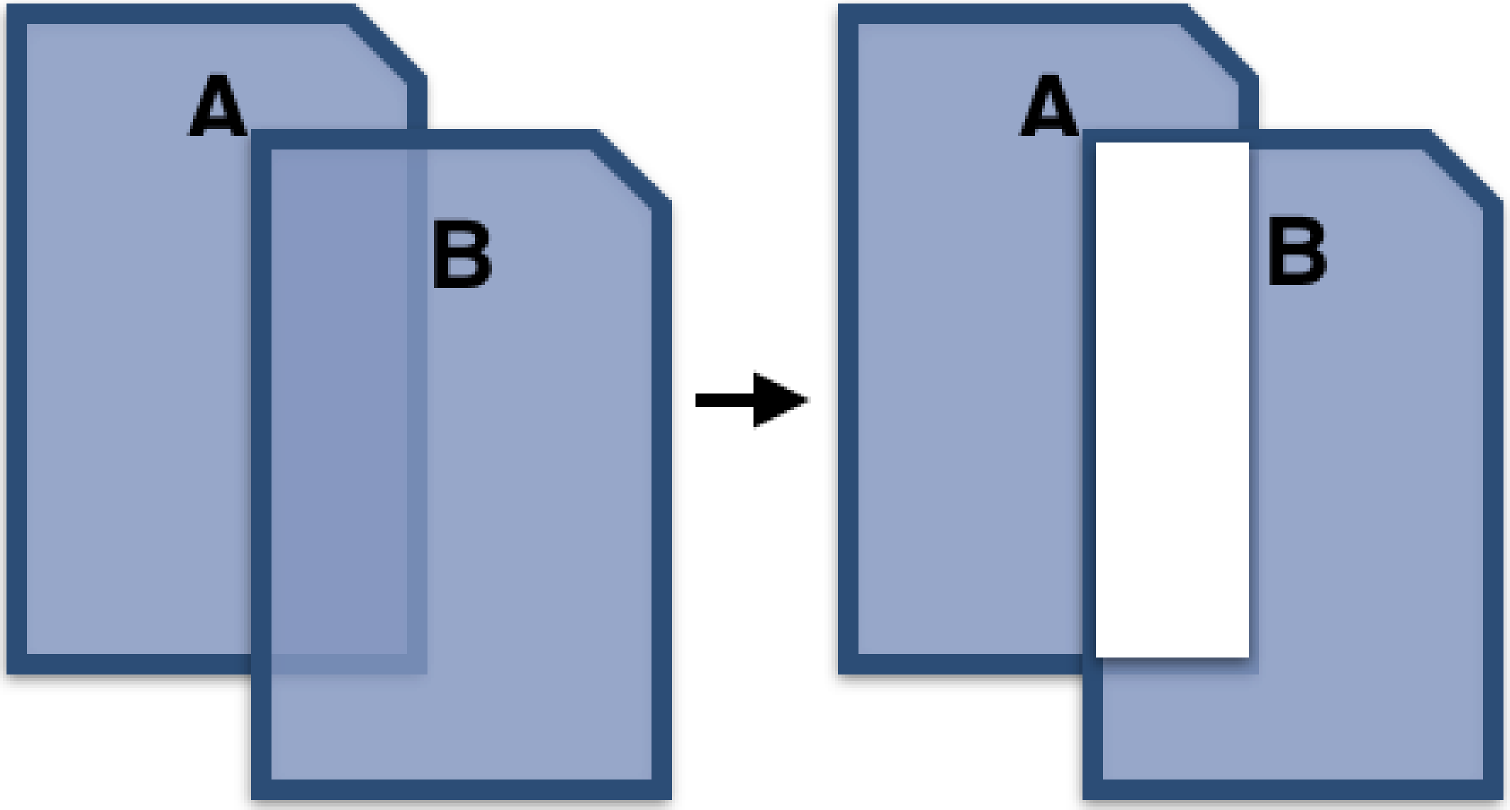


Commonality clouds

```
# Combine both corpora: all_tweets
all_coffee <- paste(coffee_tweets$text,
                    collapse = "")
all_chardonnay <- paste(chardonnay_tweets$text,
                        collapse = "")
all_tweets <- c(all_coffee, all_chardonnay)
# Clean all_tweets
all_tweets <- VectorSource(all_tweets)
all_corpus <- VCorpus(all_tweets)
all_clean <- clean_corpus(all_corpus)
all_dm <- TermDocumentMatrix(all_clean)
all_m <- as.matrix(all_tdm)
# Make commonality cloud
commonality.cloud(all_m, colors = "steelblue1",
                  max.words = 100)
```

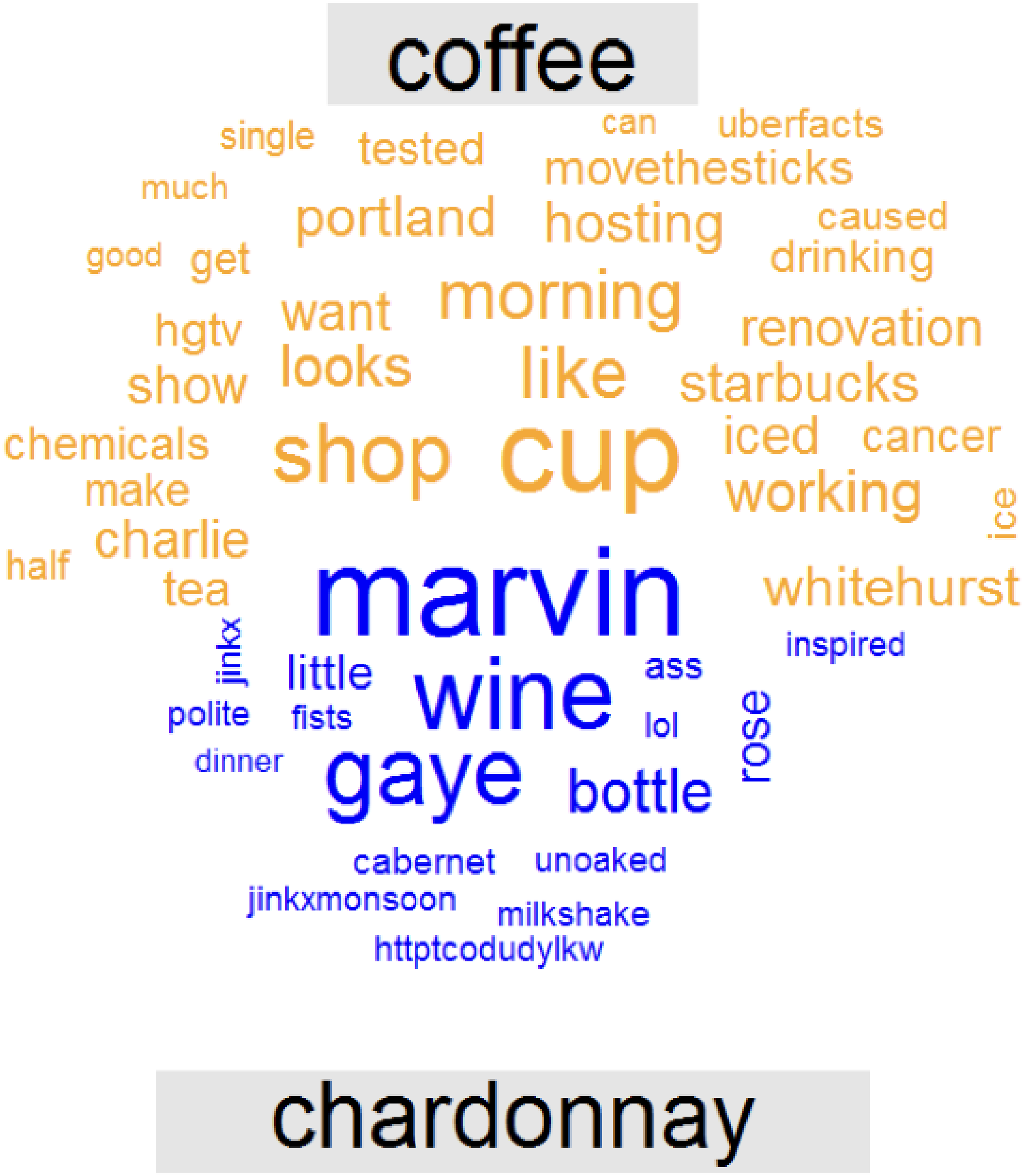


Comparison clouds



Comparison clouds

```
# Combine both corpora: all_tweets
all_coffee <- paste(coffee_tweets$text,
                    collapse = "")
all_chardonnay <- paste(chardonnay_tweets$text,
                       collapse = "")
all_tweets <- c(all_coffee, all_chardonnay)
# Clean all_tweets
all_tweets <- VectorSource(all_tweets)
all_corpus <- VCorpus(all_tweets)
all_clean <- clean_corpus(all_corpus)
all_tdm <- TermDocumentMatrix(all_clean)
colnames(all_tdm) <- c("coffee", "chardonnay")
all_m <- as.matrix(all_tdm)
# Make comparison cloud
comparison.cloud(all_m,
    colors = c("orange", "blue"), max.words = 50)
```

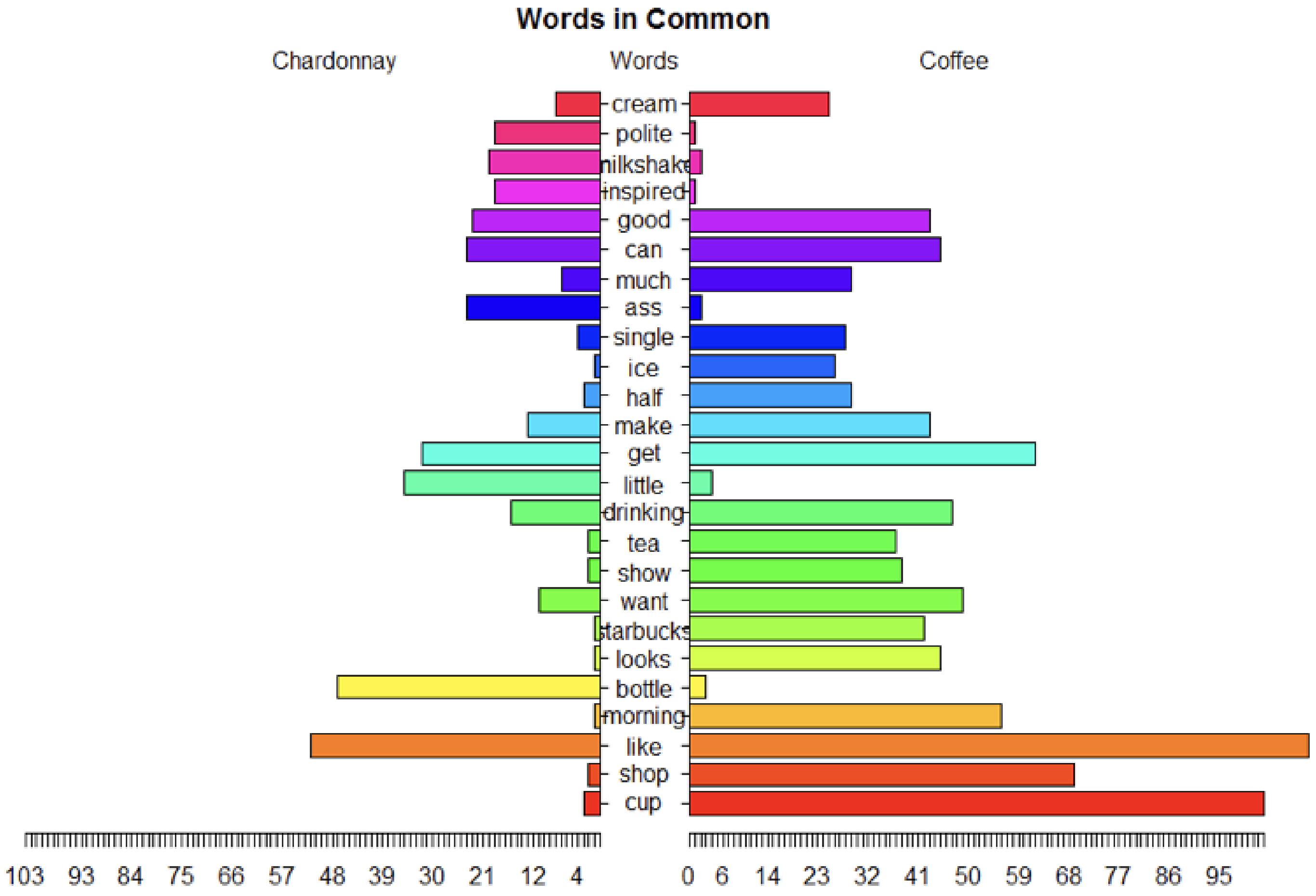


Pyramid plots

```r
# Identify terms shared by both documents
common_words <- subset(
  all_tdm_m,
  all_tdm_m[, 1] > 0 & all_tdm_m[, 2] > 0
)
# Find most commonly shared words
difference <- abs(common_words[, 1] - common_words[, 2])
common_words <- cbind(common_words, difference)
common_words <- common_words[order(common_words[, 3],
                             decreasing = TRUE), ]
top25_df <- data.frame(x = common_words[1:25, 1],
                       y = common_words[1:25, 2],
                       labels = rownames(common_words[1:25, ]))
```

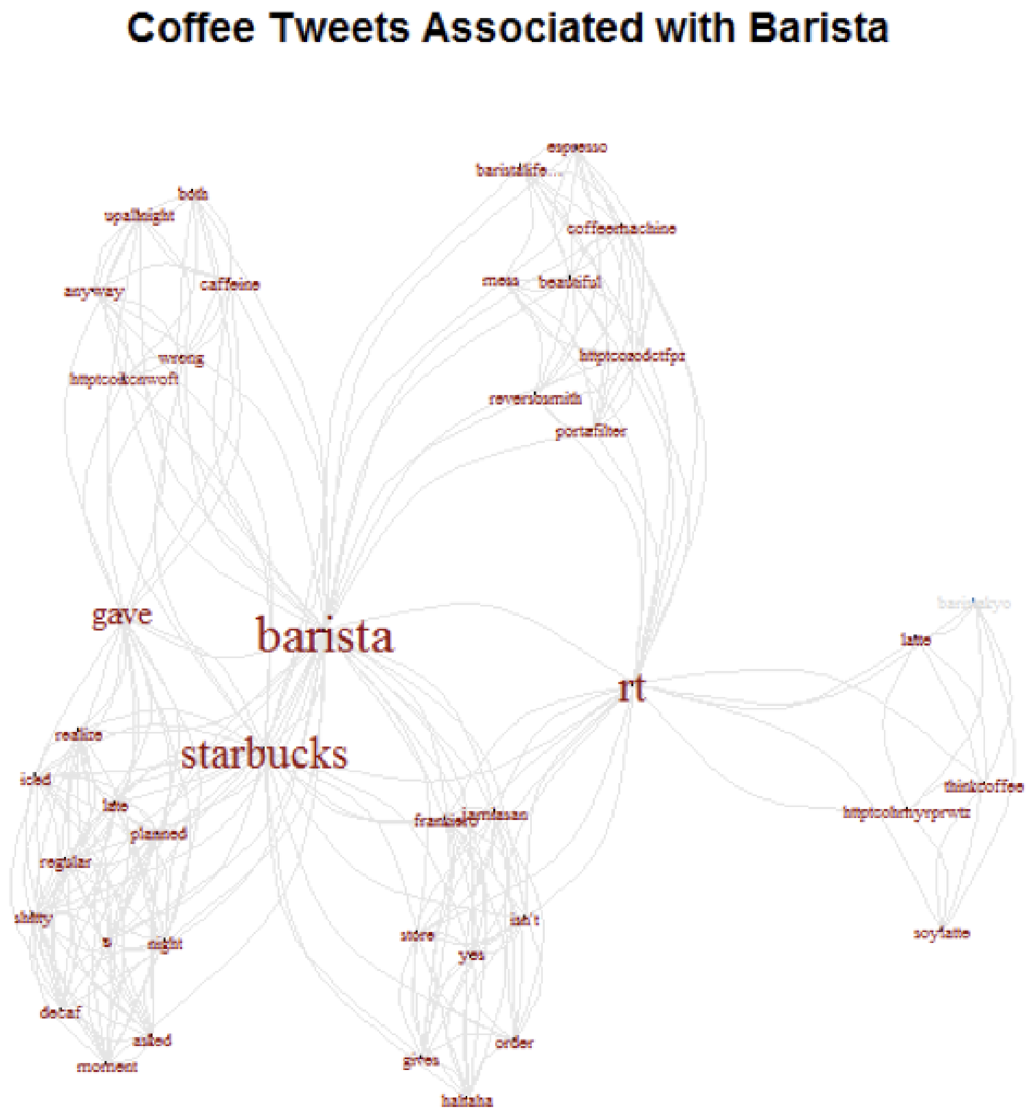
Pyramid plots

```r
# Make pyramid plot
pyramid.plot(top25_df$x, top25_df$y,
             labels = top25_df$labels,
             main = "Words in Common",
             gap = 8, laxly = NULL,
             raxlab = NULL, unit = NULL,
             top.labels = c("Chardonnay",
                            "Words",
                            "Coffee")
)
```



Word networks

```r
# Create word network
word_associate(coffee_tweets$text,
    match.string = c("barista"),
    stopwords = c(Top200Words, "coffee", "amp"),
    network.plot = TRUE,
    cloud.colors = c("gray85", "darkred"))

# Add title
title(main = "Barista Coffee Tweet Associations")
```


Coffee Tweets Associated with Barista

Let's practice!

TEXT MINING WITH BAG-OF-WORDS IN R