Assignment 16

Allison Tessman

2024-04-18

## Question.

1. Create a text data set about two different topics and:

* has a document column taking values from 1 to 5.
* has a texts column containing text data that has the topics belonging to one of the two chosen topic.
* has a source column citing the source of the text data.

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.3 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.5.0 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(tidytext)  
library(tm)

## Loading required package: NLP  
##   
## Attaching package: 'NLP'  
##   
## The following object is masked from 'package:ggplot2':  
##   
## annotate

library(wordcloud)

## Loading required package: RColorBrewer

#1. https://www.ncaa.com/news/softball/article/2024-02-05/6-storylines-note-ahead-2024-college-softball-season  
#2. https://www.flosoftball.com/articles/5043839-what-its-like-being-a-college-softball-player  
#3. #https://www.wbur.org/news/2024/04/15/boston-marathon-128th-race-runners-spectators-live-updates  
#4. https://theathletic.com/5417718/2024/04/15/boston-marathon-hellen-obiri-sisay-lemma/  
#5. https://d1softball.com/inside-the-numbers-on-secs-toughest-out-ace-up-dukes-sleeve-and-pitching-roads-to-okc/  
  
df <- read\_csv("~/Applied Analystics SAS Prog/mymath475/assign16topics.csv")

## Rows: 5 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (2): texts, links  
## dbl (1): document  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

State your topics. - Topic 1: College Softball - Topic 2: Boston Marathon

1. Perform text model on the dataset using LDA with the number of topics being 2.

* Plot the bar charts showing the terms with the highest probabilities for each topic
* Plot the word cloud of the terms for each topic.
* Does the bar charts and the word clouds identify the two chosen topics?
* Plot the distribution of topics for each document. Does the plot correctly identify the topic for each document?

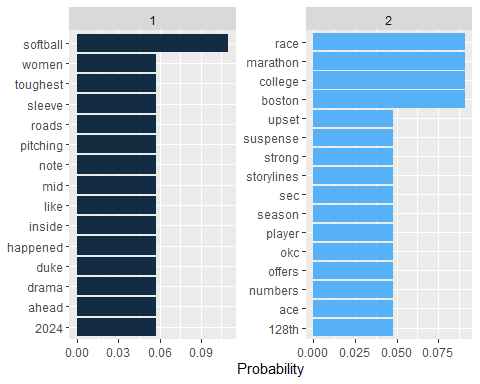
#Create doc- term matrix (DTM)  
# create the DTM  
df\_tm <- df %>%   
 unnest\_tokens(output = word, input = texts) %>%   
 anti\_join(get\_stopwords()) %>%   
 anti\_join(tibble(word = c(letters, LETTERS, "oh", 'just', 've', as.character(c(1:100)))))

## Joining with `by = join\_by(word)`  
## Joining with `by = join\_by(word)`

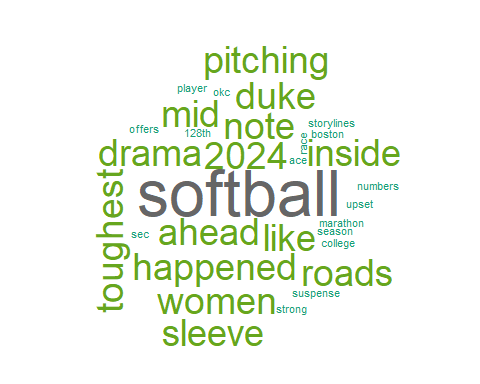
word\_freq <- df\_tm %>%   
 group\_by(document) %>% count(word, sort = TRUE)  
  
df\_dtm <- word\_freq %>%   
 cast\_dtm(document = document, term = word, n)  
  
tm::inspect(df\_dtm)

## <<DocumentTermMatrix (documents: 5, terms: 31)>>  
## Non-/sparse entries: 35/120  
## Sparsity : 77%  
## Maximal term length: 10  
## Weighting : term frequency (tf)  
## Sample :  
## Terms  
## Docs 2024 ahead boston college marathon note race season softball storylines  
## 1 1 1 0 1 0 1 0 1 1 1  
## 2 0 0 0 1 0 0 0 0 1 0  
## 3 0 0 1 0 1 0 0 0 0 0  
## 4 0 0 1 0 1 0 2 0 0 0  
## 5 0 0 0 0 0 0 0 0 0 0

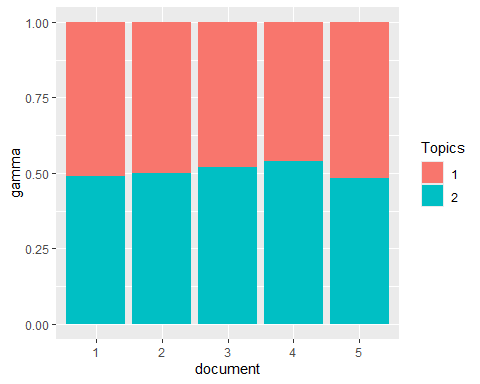
#Topic Modeling  
library(topicmodels)  
  
# Perform Topic Modeling  
  
n\_topics = 2 # set the number of topics  
  
lda\_out <-  
 LDA(df\_dtm, k = n\_topics, method = 'Gibbs',   
 control = list(seed = 1111))  
  
#Present results  
lda\_topics <- lda\_out %>%  
 tidy(matrix = "beta")   
  
word\_probs <- lda\_topics %>%  
 group\_by(topic) %>%  
 slice\_max(order\_by = beta, n = 10) %>%  
 ungroup() %>%  
 mutate(term = fct\_reorder(term, beta))  
  
# bar chart  
word\_probs %>%   
 ggplot(aes(beta, term, fill = topic)) +  
 geom\_col(show.legend = FALSE) +  
 facet\_wrap(~ as.factor(topic), scales = "free") +  
 labs(x = "Probability",  
 y = NULL)+  
 scale\_y\_reordered()



# word cloud  
 library(wordcloud)   
pal <- brewer.pal(8,"Dark2")  
  
for (i in c(1:n\_topics))  
{  
 topic <- lda\_topics %>%  
 group\_by(topic) %>%   
 filter(topic==i)  
  
topic %>%  
 with(wordcloud(term, beta, random.order = FALSE,   
 max.words = 50, colors=pal))  
}



# topic distribution for each documents  
lda\_documents = lda\_out %>%  
 tidy(matrix = "gamma")   
  
lda\_documents %>%   
 ggplot() +  
 geom\_col(aes(x = document, y = gamma, fill = factor(topic)))+  
 labs(fill = 'Topics')



1. Adding more rows and topics to the dataset and comment on the performance of LDA.

library(tidyverse)  
library(tidytext)  
library(tm)  
library(wordcloud)  
  
#1. https://www.ncaa.com/news/softball/article/2024-02-05/6-storylines-note-ahead-2024-college-softball-season  
#2. https://www.flosoftball.com/articles/5043839-what-its-like-being-a-college-softball-player  
#3. #https://www.wbur.org/news/2024/04/15/boston-marathon-128th-race-runners-spectators-live-updates  
#4. https://theathletic.com/5417718/2024/04/15/boston-marathon-hellen-obiri-sisay-lemma/  
#5. https://d1softball.com/inside-the-numbers-on-secs-toughest-out-ace-up-dukes-sleeve-and-pitching-roads-to-okc/  
#6. https://www.nytimes.com/2022/05/12/well/dog-behavior.html  
#7. https://www.medicalnewstoday.com/articles/322868  
#8. https://www.nationalgeographic.com/animals/article/150720-dogs-animals-science-pets-evolution-intelligence  
  
  
df <- read\_csv("~/Applied Analystics SAS Prog/mymath475/assign16topics2.csv")

## Rows: 8 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (2): texts, links  
## dbl (1): document  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

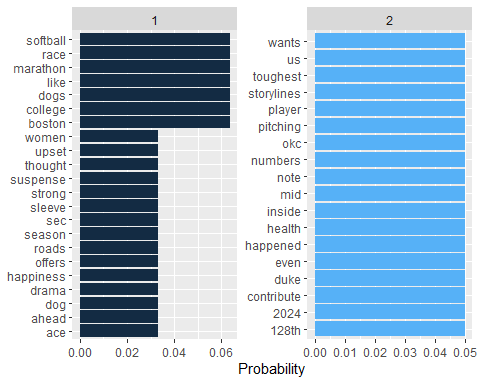
#Create doc- term matrix (DTM)  
# create the DTM  
df\_tm <- df %>%   
 unnest\_tokens(output = word, input = texts) %>%   
 anti\_join(get\_stopwords()) %>%   
 anti\_join(tibble(word = c(letters, LETTERS, "oh", 'just', 've', as.character(c(1:100)))))

## Joining with `by = join\_by(word)`  
## Joining with `by = join\_by(word)`

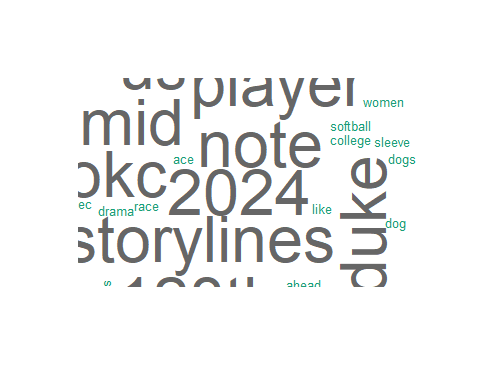
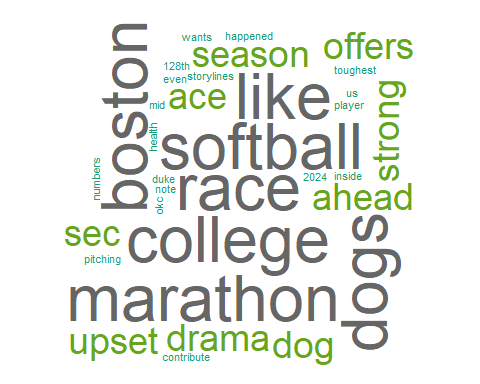
word\_freq <- df\_tm %>%   
 group\_by(document) %>% count(word, sort = TRUE)  
  
df\_dtm <- word\_freq %>%   
 cast\_dtm(document = document, term = word, n)  
  
tm::inspect(df\_dtm)

## <<DocumentTermMatrix (documents: 8, terms: 40)>>  
## Non-/sparse entries: 46/274  
## Sparsity : 86%  
## Maximal term length: 10  
## Weighting : term frequency (tf)  
## Sample :  
## Terms  
## Docs 2024 ahead boston college dogs like marathon note race softball  
## 1 1 1 0 1 0 0 0 1 0 1  
## 2 0 0 0 1 0 1 0 0 0 1  
## 3 0 0 1 0 0 0 1 0 0 0  
## 4 0 0 1 0 0 0 1 0 2 0  
## 5 0 0 0 0 0 0 0 0 0 0  
## 6 0 0 0 0 0 0 0 0 0 0  
## 7 0 0 0 0 1 0 0 0 0 0  
## 8 0 0 0 0 1 1 0 0 0 0

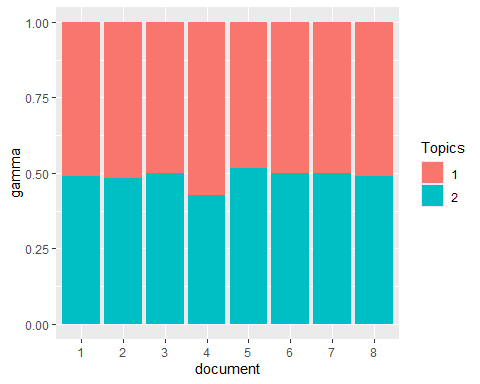
#Topic Modeling  
library(topicmodels)  
  
# Perform Topic Modeling  
  
n\_topics = 2 # set the number of topics  
  
lda\_out <-  
 LDA(df\_dtm, k = n\_topics, method = 'Gibbs',   
 control = list(seed = 1111))  
  
#Present results  
lda\_topics <- lda\_out %>%  
 tidy(matrix = "beta")   
  
word\_probs <- lda\_topics %>%  
 group\_by(topic) %>%  
 slice\_max(order\_by = beta, n = 10) %>%  
 ungroup() %>%  
 mutate(term = fct\_reorder(term, beta))  
  
# bar chart  
word\_probs %>%   
 ggplot(aes(beta, term, fill = topic)) +  
 geom\_col(show.legend = FALSE) +  
 facet\_wrap(~ as.factor(topic), scales = "free") +  
 labs(x = "Probability",  
 y = NULL)+  
 scale\_y\_reordered()



# word cloud  
 library(wordcloud)   
pal <- brewer.pal(8,"Dark2")  
  
for (i in c(1:n\_topics))  
{  
 topic <- lda\_topics %>%  
 group\_by(topic) %>%   
 filter(topic==i)  
  
topic %>%  
 with(wordcloud(term, beta, random.order = FALSE,   
 max.words = 50, colors=pal))  
}



# topic distribution for each documents  
lda\_documents = lda\_out %>%  
 tidy(matrix = "gamma")   
  
lda\_documents %>%   
 ggplot() +  
 geom\_col(aes(x = document, y = gamma, fill = factor(topic)))+  
 labs(fill = 'Topics')



1. Changing the number of topics from 2 to 3 and comment on the performance of LDA.

library(tidyverse)  
library(tidytext)  
library(tm)  
library(wordcloud)  
  
#1. https://www.ncaa.com/news/softball/article/2024-02-05/6-storylines-note-ahead-2024-college-softball-season  
#2. https://www.flosoftball.com/articles/5043839-what-its-like-being-a-college-softball-player  
#3. #https://www.wbur.org/news/2024/04/15/boston-marathon-128th-race-runners-spectators-live-updates  
#4. https://theathletic.com/5417718/2024/04/15/boston-marathon-hellen-obiri-sisay-lemma/  
#5. https://d1softball.com/inside-the-numbers-on-secs-toughest-out-ace-up-dukes-sleeve-and-pitching-roads-to-okc/  
#6. https://www.nytimes.com/2022/05/12/well/dog-behavior.html  
#7. https://www.medicalnewstoday.com/articles/322868  
#8. https://www.nationalgeographic.com/animals/article/150720-dogs-animals-science-pets-evolution-intelligence  
  
  
df <- read\_csv("~/Applied Analystics SAS Prog/mymath475/assign16topics2.csv")

## Rows: 8 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (2): texts, links  
## dbl (1): document  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

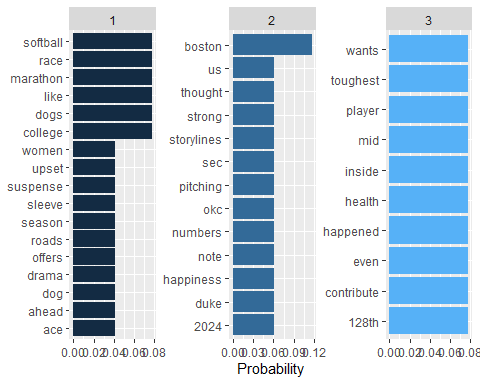
#Create doc- term matrix (DTM)  
# create the DTM  
df\_tm <- df %>%   
 unnest\_tokens(output = word, input = texts) %>%   
 anti\_join(get\_stopwords()) %>%   
 anti\_join(tibble(word = c(letters, LETTERS, "oh", 'just', 've', as.character(c(1:100)))))

## Joining with `by = join\_by(word)`  
## Joining with `by = join\_by(word)`

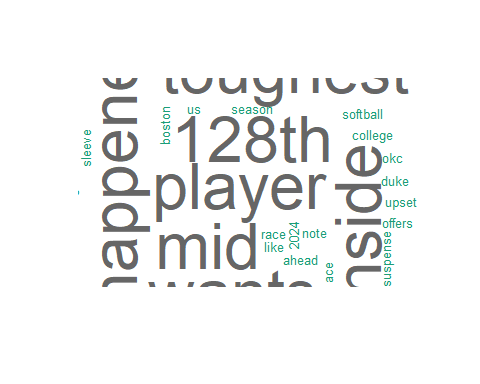
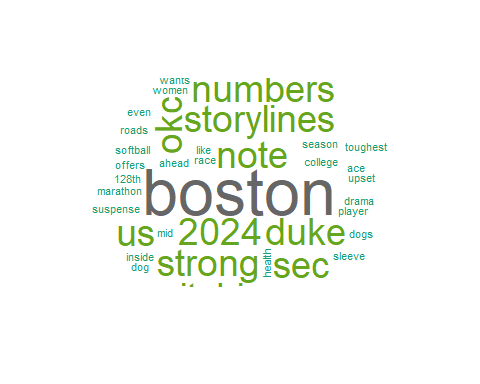
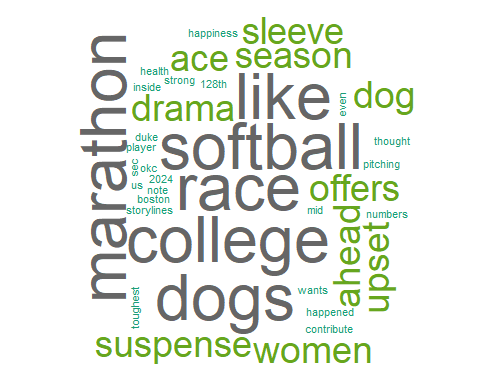
word\_freq <- df\_tm %>%   
 group\_by(document) %>% count(word, sort = TRUE)  
  
df\_dtm <- word\_freq %>%   
 cast\_dtm(document = document, term = word, n)  
  
tm::inspect(df\_dtm)

## <<DocumentTermMatrix (documents: 8, terms: 40)>>  
## Non-/sparse entries: 46/274  
## Sparsity : 86%  
## Maximal term length: 10  
## Weighting : term frequency (tf)  
## Sample :  
## Terms  
## Docs 2024 ahead boston college dogs like marathon note race softball  
## 1 1 1 0 1 0 0 0 1 0 1  
## 2 0 0 0 1 0 1 0 0 0 1  
## 3 0 0 1 0 0 0 1 0 0 0  
## 4 0 0 1 0 0 0 1 0 2 0  
## 5 0 0 0 0 0 0 0 0 0 0  
## 6 0 0 0 0 0 0 0 0 0 0  
## 7 0 0 0 0 1 0 0 0 0 0  
## 8 0 0 0 0 1 1 0 0 0 0

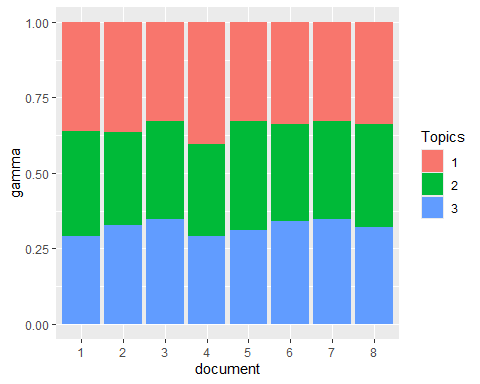
#Topic Modeling  
library(topicmodels)  
  
# Perform Topic Modeling  
  
n\_topics = 3 # set the number of topics  
  
lda\_out <-  
 LDA(df\_dtm, k = n\_topics, method = 'Gibbs',   
 control = list(seed = 1111))  
  
#Present results  
lda\_topics <- lda\_out %>%  
 tidy(matrix = "beta")   
  
word\_probs <- lda\_topics %>%  
 group\_by(topic) %>%  
 slice\_max(order\_by = beta, n = 10) %>%  
 ungroup() %>%  
 mutate(term = fct\_reorder(term, beta))  
  
# bar chart  
word\_probs %>%   
 ggplot(aes(beta, term, fill = topic)) +  
 geom\_col(show.legend = FALSE) +  
 facet\_wrap(~ as.factor(topic), scales = "free") +  
 labs(x = "Probability",  
 y = NULL)+  
 scale\_y\_reordered()



# word cloud  
 library(wordcloud)   
pal <- brewer.pal(8,"Dark2")  
  
for (i in c(1:n\_topics))  
{  
 topic <- lda\_topics %>%  
 group\_by(topic) %>%   
 filter(topic==i)  
  
topic %>%  
 with(wordcloud(term, beta, random.order = FALSE,   
 max.words = 50, colors=pal))  
}



# topic distribution for each documents  
lda\_documents = lda\_out %>%  
 tidy(matrix = "gamma")   
  
lda\_documents %>%   
 ggplot() +  
 geom\_col(aes(x = document, y = gamma, fill = factor(topic)))+  
 labs(fill = 'Topics')



1. Find a text dataset and perform text modeling on the dataset.

library(tidyverse)  
library(tidytext)  
library(tm)  
library(wordcloud)  
  
#CNN news highlights text  
df <- read\_csv("~/Applied Analystics SAS Prog/mymath475/CNNtext.csv")

## Rows: 11490 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (3): id, article, highlights  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

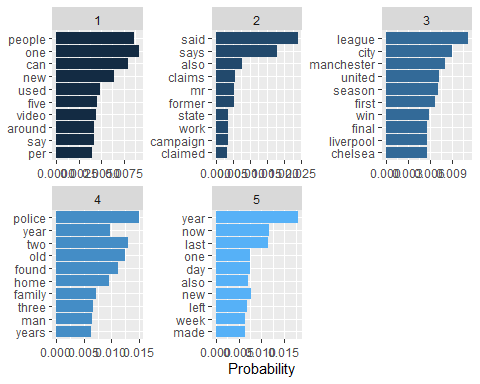
#Create doc- term matrix (DTM)  
# create the DTM  
df\_tm <- df %>%   
 unnest\_tokens(output = word, input = highlights) %>%   
 anti\_join(get\_stopwords()) %>%   
 anti\_join(tibble(word = c(letters, LETTERS, "oh", 'just', 've', as.character(c(1:100)))))

## Joining with `by = join\_by(word)`  
## Joining with `by = join\_by(word)`

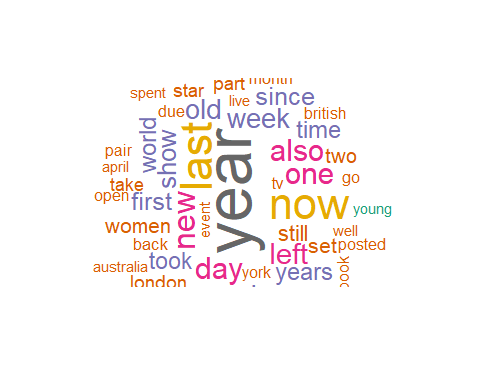
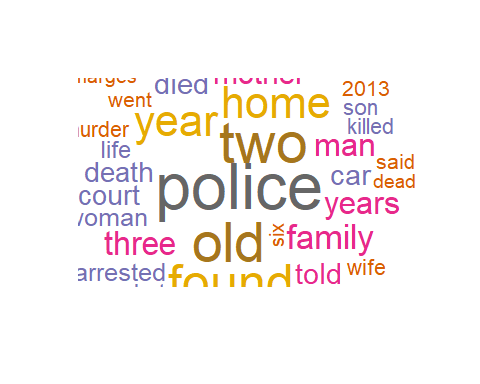
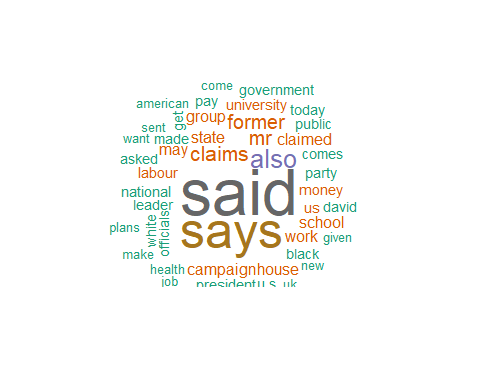
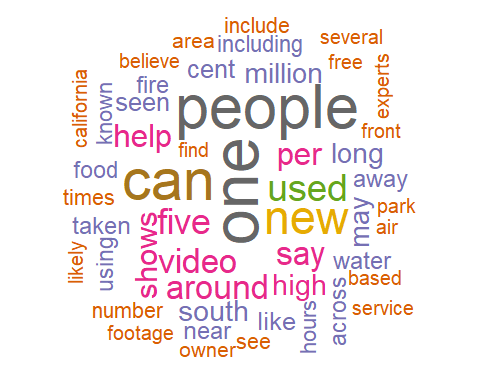
word\_freq <- df\_tm %>%   
 group\_by(id) %>% count(word, sort = TRUE)  
  
df\_dtm <- word\_freq %>%   
 cast\_dtm(document = id, term = word, n)  
  
tm::inspect(df\_dtm)

## <<DocumentTermMatrix (documents: 11490, terms: 37297)>>  
## Non-/sparse entries: 341637/428200893  
## Sparsity : 100%  
## Maximal term length: 33  
## Weighting : term frequency (tf)  
## Sample :  
## Terms  
## Docs also last new old one police said  
## 3815f10f7a7df0067d9ef27fb9e68d1803a615d0 0 0 0 0 0 0 0  
## 5a214c5117ec8b32d076653147711a9d2426fac2 0 5 0 0 0 0 0  
## 613e86fb4098c7bb4058aa17f5db4b5e4a1a3506 2 0 0 0 3 0 0  
## 6ddf76d9a7fcdbb060b394970e9558faa819eccc 0 0 2 0 0 0 0  
## 7cee65d7accdda886e2d9d2a2b7d02e3af72e41d 1 0 0 2 0 0 0  
## 8c9120d228fecd6791d7e5b5641549964500f18e 0 0 1 0 0 0 1  
## 95fc90b781eda59646706905b675ce5219725be1 0 0 1 0 0 1 0  
## cbed2a778a5e17d729c6e5ee5691b88710620dd7 1 0 2 0 0 0 0  
## d8fbffc5ba638a17bab565436ae8b3f197d7eb53 2 0 0 0 1 0 0  
## fa5594de0a40cf58a2aab30056f1860ed1177c9b 0 0 4 0 0 0 0  
## Terms  
## Docs says two year  
## 3815f10f7a7df0067d9ef27fb9e68d1803a615d0 0 0 0  
## 5a214c5117ec8b32d076653147711a9d2426fac2 0 3 0  
## 613e86fb4098c7bb4058aa17f5db4b5e4a1a3506 0 0 0  
## 6ddf76d9a7fcdbb060b394970e9558faa819eccc 0 0 2  
## 7cee65d7accdda886e2d9d2a2b7d02e3af72e41d 0 0 1  
## 8c9120d228fecd6791d7e5b5641549964500f18e 1 1 1  
## 95fc90b781eda59646706905b675ce5219725be1 0 1 1  
## cbed2a778a5e17d729c6e5ee5691b88710620dd7 0 1 1  
## d8fbffc5ba638a17bab565436ae8b3f197d7eb53 4 0 1  
## fa5594de0a40cf58a2aab30056f1860ed1177c9b 0 0 1

#Topic Modeling  
library(topicmodels)  
  
# Perform Topic Modeling  
  
n\_topics = 5 # set the number of topics  
  
lda\_out <-  
 LDA(df\_dtm, k = n\_topics, method = 'Gibbs',   
 control = list(seed = 1111))  
  
#Present results  
lda\_topics <- lda\_out %>%  
 tidy(matrix = "beta")   
  
word\_probs <- lda\_topics %>%  
 group\_by(topic) %>%  
 slice\_max(order\_by = beta, n = 10) %>%  
 ungroup() %>%  
 mutate(term = fct\_reorder(term, beta))  
  
# bar chart  
word\_probs %>%   
 ggplot(aes(beta, term, fill = topic)) +  
 geom\_col(show.legend = FALSE) +  
 facet\_wrap(~ as.factor(topic), scales = "free") +  
 labs(x = "Probability",  
 y = NULL)+  
 scale\_y\_reordered()



# word cloud  
 library(wordcloud)   
pal <- brewer.pal(8,"Dark2")  
  
for (i in c(1:n\_topics))  
{  
 topic <- lda\_topics %>%  
 group\_by(topic) %>%   
 filter(topic==i)  
  
topic %>%  
 with(wordcloud(term, beta, random.order = FALSE,   
 max.words = 50, colors=pal))  
}



# topic distribution for each documents  
lda\_documents = lda\_out %>%  
 tidy(matrix = "gamma")   
  
lda\_documents %>%   
 ggplot() +  
 geom\_col(aes(x = document, y = gamma, fill = factor(topic)))+  
 labs(fill = 'Topics')

