Tammy Glazer Homework 5

11/27/2019

Preprocessing and (light) EDA

1. Load the platforms.csv file containing the 2016 Democratic and Republican party platforms. Note the 2x2 format, where each row is a document, with the party recorded as a separate feature. Also, load the individual party .txt files as a corpus.

```
library(tidyverse)
library(tm)
library(grid)
library(wordcloud)
library(wordcloud2)
library(tidytext)
library(knitr)
library(grid)
library(gridExtra)
library(lda)
library(topicmodels)
library(tidytext)
library(udpipe)
library(tokenizers)
library(caret)
```

```
# platforms <- read_csv('platforms.csv', col_names=TRUE)
dir <- "/Users/Tammy/Documents/_MSCAPP/Fall 2019/Unsupervised ML/PS5/Problem-Set-5/Party Platforms Data/
corpus <- VCorpus(DirSource(dir), readerControl = list(language="en"))
corpus
```

```
## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 0
## Content:  documents: 2
```

2. Create a document-term matrix and preprocess the platforms by the following criteria (at a minimum): 1. Convert to lowercase, 2. Remove the stopwords, 3. Remove the numbers, 4. Remove all punctuation, 5. Remove the whitespace.

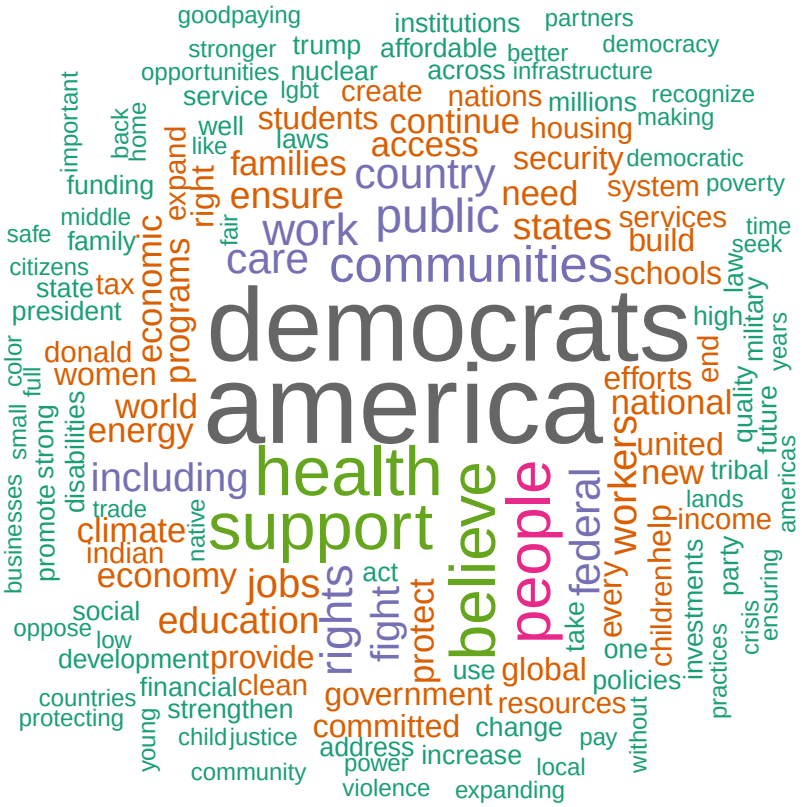
```
# Convert to lowercase
corpus <- tm_map(corpus, tolower)
corpus <- tm_map(corpus, PlainTextDocument)
# Remove stopwords
corpus <- tm_map(corpus, removeWords, stopwords("en"))
corpus <- tm_map(corpus, PlainTextDocument)
# Remove numbers
```

```
corpus <- tm_map(corpus, removeNumbers)
corpus <- tm_map(corpus, PlainTextDocument)
# Remove punctuation
corpus <- tm_map(corpus, removePunctuation)
corpus <- tm_map(corpus, PlainTextDocument)
# Minor custom adjustments based on skimming the corpus
for (j in seq(corpus)) {
  corpus[[j]] <- gsub("besttrained", "best trained", corpus[[j]])
  corpus[[j]] <- gsub("bestequipped", "best equipped", corpus[[j]])
  corpus[[j]] <- gsub("inequality-", "best equipped", corpus[[j]])
  corpus[[j]] <- gsub("-", "", corpus[[j]])
  corpus[[j]] <- gsub("lowincome", "low income", corpus[[j]])
  corpus[[j]] <- gsub("  ", " ", corpus[[j]])
  corpus[[j]] <- gsub("highquality", "high quality", corpus[[j]])
  corpus[[j]] <- gsub("sixday", "six day", corpus[[j]])
  corpus[[j]] <- gsub("economyincluding", "economy including", corpus[[j]])
  corpus[[j]] <- gsub("americans", "america", corpus[[j]])
  corpus[[j]] <- gsub("american", "america", corpus[[j]])
  corpus[[j]] <- gsub("/", " ", corpus[[j]])
  corpus[[j]] <- gsub("’", " ", corpus[[j]])
  corpus[[j]] <- gsub("-", " ", corpus[[j]])
  corpus[[j]] <- gsub("\\|", " ", corpus[[j]])
  corpus[[j]] <- gsub("@", " ", corpus[[j]])
  corpus[[j]] <- gsub("\u2028", " ", corpus[[j]])
}
corpus <- tm_map(corpus, PlainTextDocument)

# Remove terms that appear frequently with little meaning
corpus <- tm_map(corpus, removeWords, c('will', 'also', 'must', 'can', 'make', 'many'))
corpus <- tm_map(corpus, PlainTextDocument)
# Remove whitespace again
corpus <- tm_map(corpus, stripWhitespace)
corpus <- tm_map(corpus, PlainTextDocument)

# Create a document-term matrix
dtm_full <- DocumentTermMatrix(corpus, control=list(minDocFreq=2, minWordLength=2))
dtm_d <- DocumentTermMatrix(corpus[1], control=list(minDocFreq=2, minWordLength=2))
dtm_r <- DocumentTermMatrix(corpus[2], control=list(minDocFreq=2, minWordLength=2))

# Explore term frequency
frequency_full <- sort(colSums(as.matrix(dtm_full)), decreasing=TRUE)
frequency_d <- sort(colSums(as.matrix(dtm_d)), decreasing=TRUE)
frequency_r <- sort(colSums(as.matrix(dtm_r)), decreasing=TRUE)
```

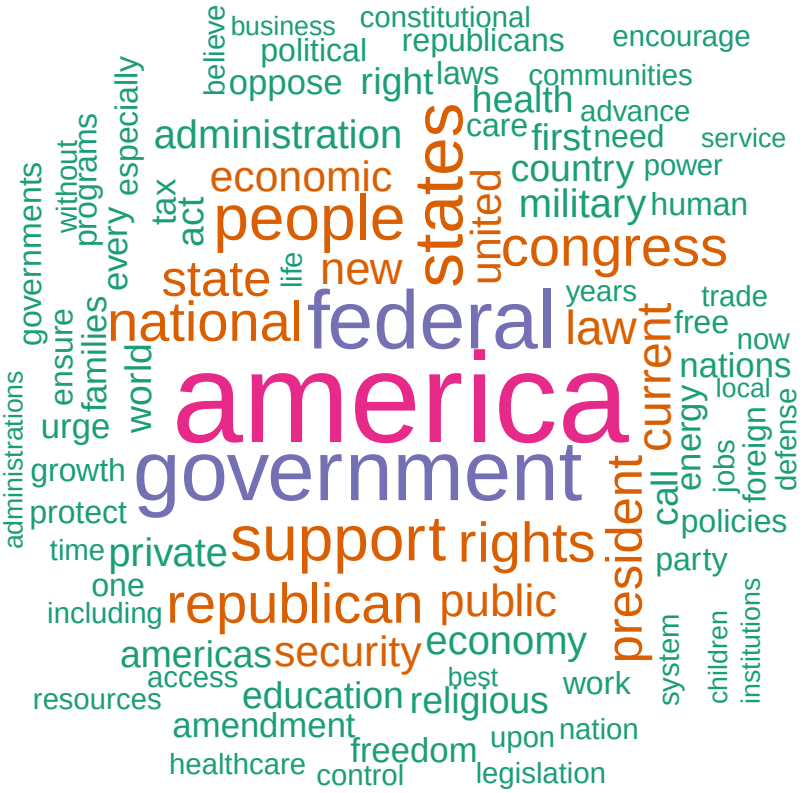
3. Visually inspect your cleaned documents by creating a wordcloud for each major party's platform. Based on this naive visualization, offer a few-sentence description of general patterns you see (eg. What are commonly used words? What are less commonly used words? Can you get a sense of differences between the parties at this early stage?)

Note: The titles are not rendering, but the first wordcloud is for the Democratic platform and the second is for the Republican platform.

```
set.seed(111)

# Figure Title: Word Cloud Democrat
d <- wordcloud(names(frequency_d), frequency_d,
               scale=c(3.5, 0.5),
               min.freq = 18,
               random.order = FALSE,
               rot.per = 0.3,
               title = "Word Cloud Democrat",
               colors = brewer.pal(10, "Dark2"))
```




```
# Figure Title: Word Cloud Republican
r <- wordcloud(names(frequency_r), frequency_r,
               scale=c(4, 0.4),
               min.freq = 29,
               random.order = FALSE,
               rot.per = 0.25,
               main = "Word Cloud Republican",
               colors = brewer.pal(4, "Dark2"))
```



The Democratic platform appears to strongly emphasize the topic of health and appears to prioritize the theme of community (public, communities, people). There is wide variety of secondary topics mentioned, including housing, social services, women/children/families, climate, education, and the economy, as well as a running theme of affordability. Less common terms tend to focus on areas like protection, strength, violence, trade, and business. The Republican platform, meanwhile, appears to strongly emphasize institutions (government, state, administration, congress, military) as well as the topics of the law (freedom, constitution, laws, policies) and security. Less common terms tend to focus on areas like foreign affairs and social services. At this early stage, we can already get a sense of the differences between parties. It seems that the Democratic platform places a greater emphasis on discussing social issues and services affecting specific populations, whereas the Republic platform emphasizes topics that may be less population-specific, like the law, the national economy, and the administration. It is interesting to note that the Democratic platform seems to use a few words many times, whereas the Republican platform demonstrates less variation in the number of times each common term is used.

Sentiment Analysis

4. Use the "Bing" and "AFINN" dictionaries to calculate the sentiment of each cleaned party platform. Present the results hower you'd like (eg. visually and/or numerically).

```
# Create a tibble with one word per row (Democrat)
one_term_doc_row_d <- tidy(dtm_d)
# Perform inner join with bing dictionary
sentiments_d_b <- one_term_doc_row_d %>%
                  inner_join(get_sentiments("bing"), by = c(term = "word"))
```

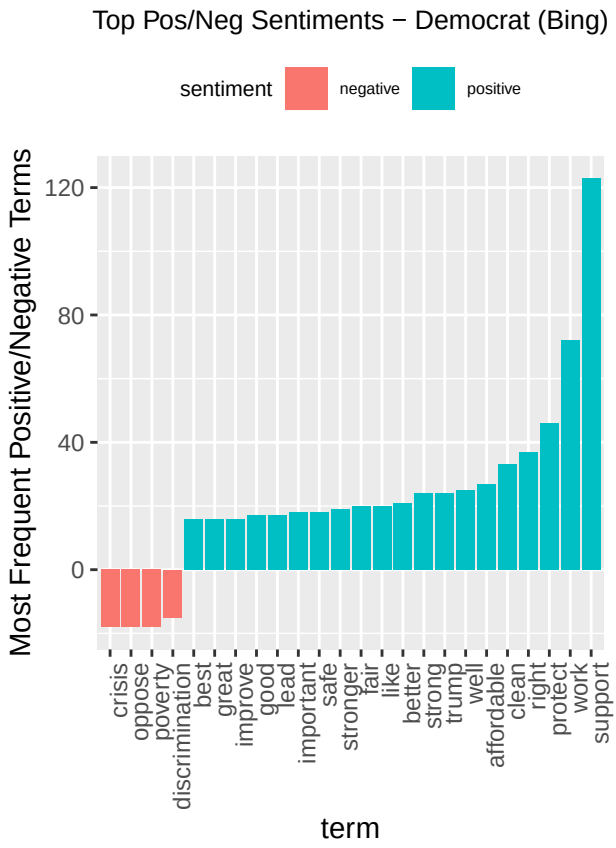
```r
# Calculate Democrat sentiment score (bing)
score_d_b <- sentiments_d_b %>%
             count(sentiment, wt = count) %>%
             spread(sentiment, n, fill = 0) %>%
             mutate(tone = (positive - negative) / (positive + negative))
# Perform inner join with AFINN dictionary
sentiments_d_a <- one_term_doc_row_d %>%
                  inner_join(get_sentiments("afinn"), by = c(term = "word"))
# Calculate Democrat sentiment score (AFINN)
score_d_a <- sentiments_d_a %>%
             mutate(weighted_value = (count * value)) %>%
             mutate(score = sum(weighted_value) / sum(count))


# Create a tibble with one word per row (Republican)
one_term_doc_row_r <- tidy(dtm_r)
# Perform inner join with bing dictionary
sentiments_r_b <- one_term_doc_row_r %>%
                inner_join(get_sentiments("bing"), by = c(term = "word"))
# Calculate sentiment score
score_r_b <- sentiments_r_b %>%
             count(sentiment, wt = count) %>%
             spread(sentiment, n, fill = 0) %>%
             mutate(tone = (positive - negative) / (positive + negative))
# Perform inner join with AFINN dictionary
sentiments_r_a <- one_term_doc_row_r %>%
                  inner_join(get_sentiments("afinn"), by = c(term = "word"))
# Calculate Republican sentiment score (AFINN)
score_r_a <- sentiments_r_a %>%
             mutate(weighted_value = (count * value)) %>%
             mutate(score = sum(weighted_value) / sum(count))

results <- data.frame("Party" = c("Democrat", "Republican", "Democrat", "Republican"),
                      "Dictionary" = c("Bing", "Bing", "AFINN", "AFINN"),
                      "Score" = c(score_d_b$tone,
                                  score_r_b$tone,
                                  score_d_a$score[1],
                                  score_r_a$score[1]))
kable(results, caption="Sentiment of Each Platform by Dictionary Used")
```

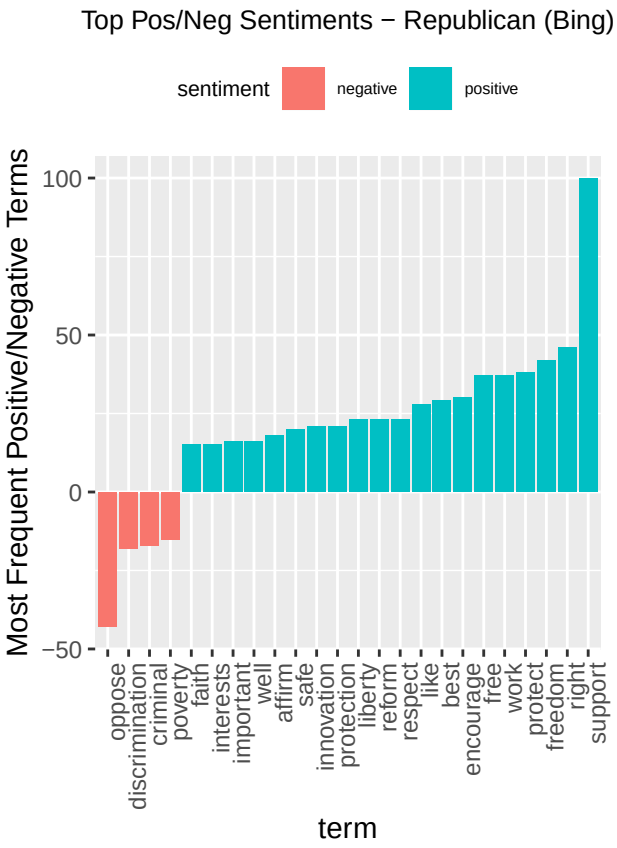
Table 1: Sentiment of Each Platform by Dictionary Used

Party	Dictionary	Score
Democrat	Bing	0.2581385
Republican	Bing	0.1179596
Democrat	AFINN	0.5665514
Republican	AFINN	0.3554217

```r
# Visualize words that contribute to positive and negative sentiment (Bing)
d_b <- sentiments_d_b %>%
       count(sentiment, term, wt = count) %>%
```

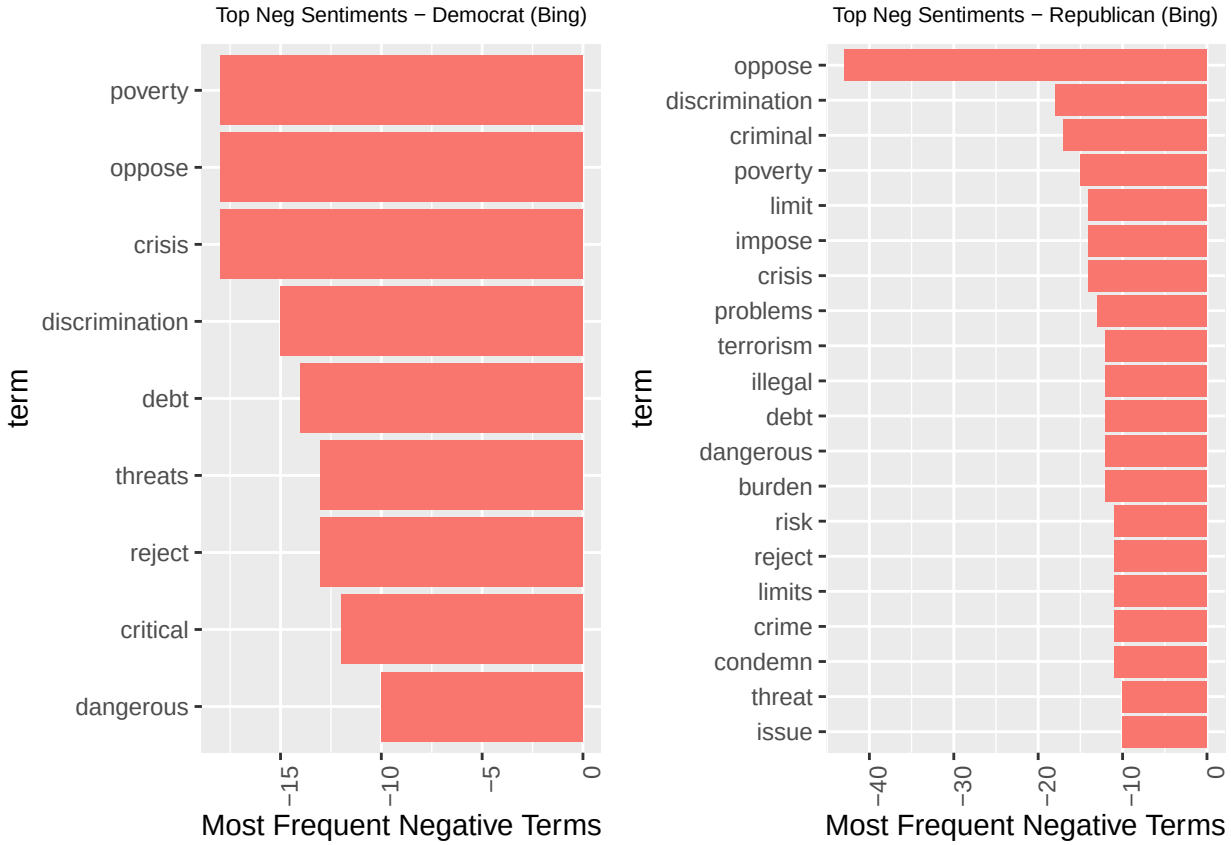
```
       filter(n >= 15) %>%
       mutate(n = ifelse(sentiment == "negative", -n, n)) %>%
       mutate(term = reorder(term, n)) %>%
       ggplot(aes(term, n, fill = sentiment)) +
         geom_bar(stat = "identity") +
         theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
         ylab("Most Frequent Positive/Negative Terms") +
         ggtitle("Top Pos/Neg Sentiments - Democrat (Bing)") +
         theme(plot.title = element_text(hjust = 0.5)) +
         theme(plot.title = element_text(size=10)) +
         theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
         theme(legend.position="top") +
         theme(legend.title = element_text(size=8)) +
         theme(legend.text = element_text(size=6))

r_b <- sentiments_r_b %>%
       count(sentiment, term, wt = count) %>%
       filter(n >= 15) %>%
       mutate(n = ifelse(sentiment == "negative", -n, n)) %>%
       mutate(term = reorder(term, n)) %>%
       ggplot(aes(term, n, fill = sentiment)) +
         geom_bar(stat = "identity") +
         theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
         ylab("Most Frequent Positive/Negative Terms") +
         ggtitle("Top Pos/Neg Sentiments - Republican (Bing)") +
         theme(plot.title = element_text(hjust = 0.5)) +
         theme(plot.title = element_text(size=10)) +
         theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
         theme(legend.position = "top") +
         theme(legend.title = element_text(size=8)) +
         theme(legend.text = element_text(size=6))

grid.arrange(d_b, r_b, ncol=2, nrow=1)
```





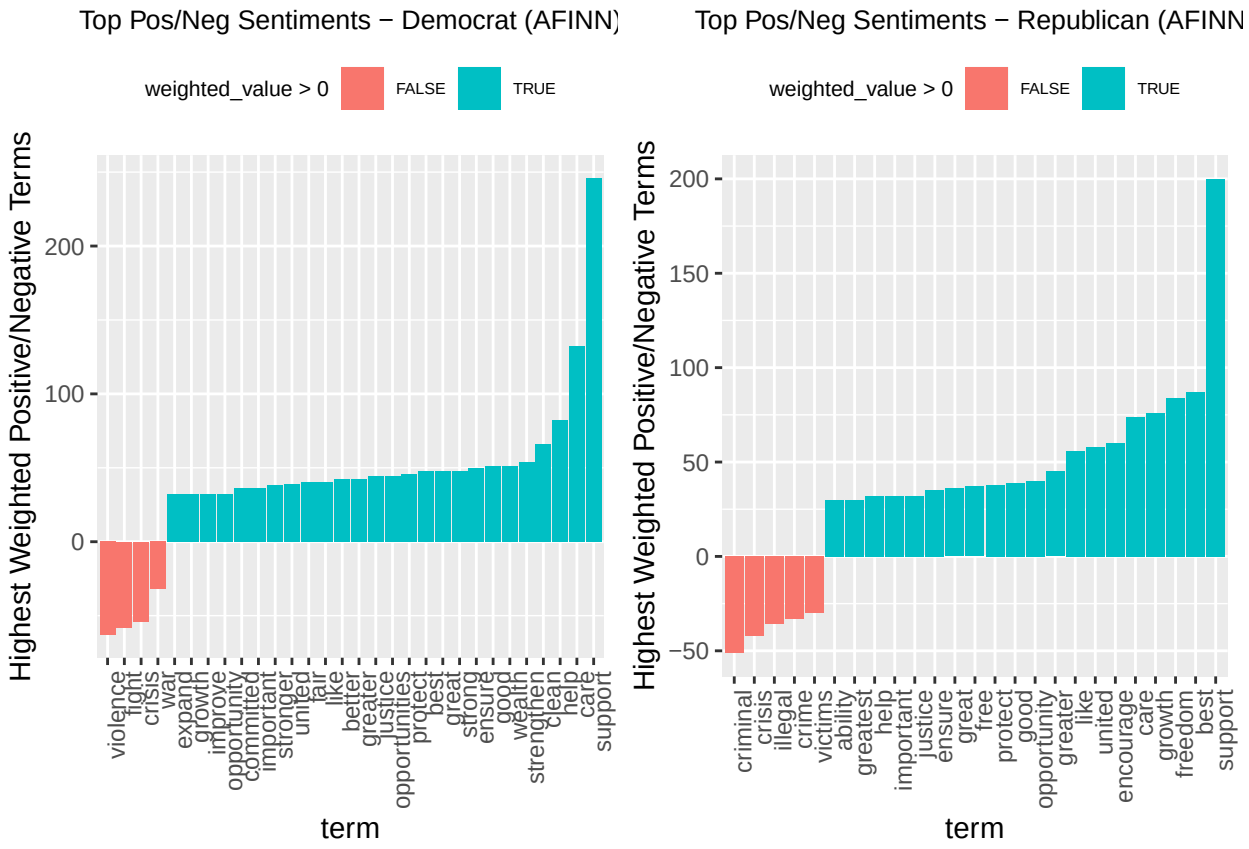

```r
# Deeper dive into negative terms (Bing)
bing_neg_d <- sentiments_d_b %>%
       count(sentiment, term, wt = count) %>%
       filter(n >= 10 & sentiment == 'negative') %>%
       mutate(n = -n) %>%
       mutate(term = reorder(term, -n)) %>%
       ggplot(aes(term, n, fill = 'salmon')) +
       geom_bar(stat = "identity") +
       theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
       ylab("Most Frequent Negative Terms") +
       ggtitle("Top Neg Sentiments - Democrat (Bing)") +
       theme(plot.title = element_text(hjust = 0.5)) +
       theme(plot.title = element_text(size=8)) +
       theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
       theme(legend.position = "none") +
       coord_flip()

bing_neg_r <- sentiments_r_b %>%
       count(sentiment, term, wt = count) %>%
       filter(n >= 10 & sentiment == 'negative') %>%
       mutate(n = -n) %>%
       mutate(term = reorder(term, -n)) %>%
       ggplot(aes(term, n, fill = 'salmon')) +
       geom_bar(stat = "identity") +
       theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
       ylab("Most Frequent Negative Terms") +
       ggtitle("Top Neg Sentiments - Republican (Bing)") +
```

```
        theme(plot.title = element_text(hjust = 0.5)) +
        theme(plot.title = element_text(size=8)) +
        theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
        theme(legend.position = "none") +
        coord_flip()

grid.arrange(bing_neg_d, bing_neg_r, ncol=2, nrow=1)
```



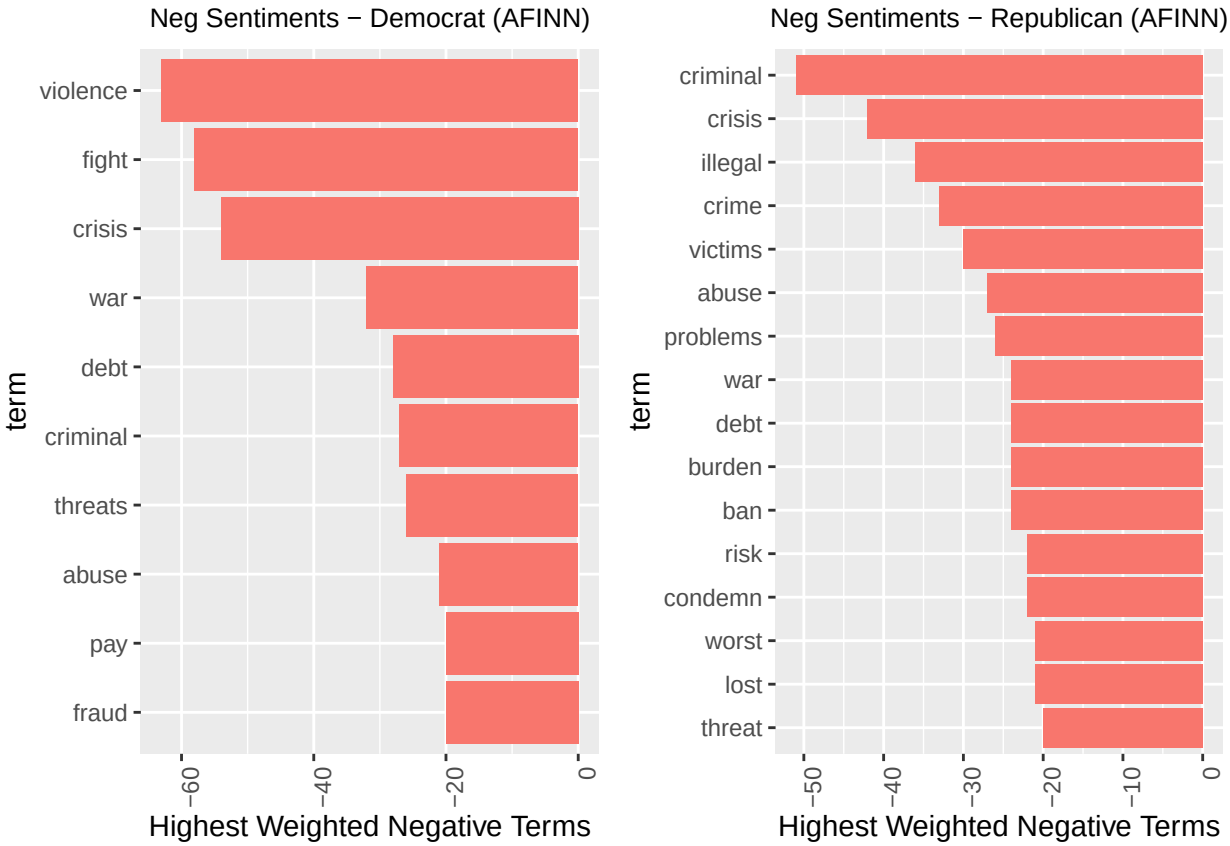

```
# Visualize words that contribute to positive and negative sentiment (AFINN)
d_a <- sentiments_d_a %>%
        mutate(weighted_value = (count * value)) %>%
        filter(weighted_value >= 30 | weighted_value <= -30) %>%
        arrange(weighted_value) %>%
        ggplot(aes(reorder(term, weighted_value), weighted_value, fill = weighted_value > 0)) +
          geom_bar(stat = "identity") +
          theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
          ylab("Highest Weighted Positive/Negative Terms") +
          xlab("term") +
          ggtitle("Top Pos/Neg Sentiments - Democrat (AFINN)") +
          theme(plot.title = element_text(hjust = 0.5)) +
          theme(plot.title = element_text(size=10)) +
          theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
          theme(legend.position="top") +
          theme(legend.title = element_text(size=8)) +
          theme(legend.text = element_text(size=6))
```

```
r_a <- sentiments_r_a %>%
       mutate(weighted_value = (count * value)) %>%
       filter(weighted_value >= 30 | weighted_value <= -30) %>%
       arrange(weighted_value) %>%
       ggplot(aes(reorder(term, weighted_value), weighted_value, fill = weighted_value > 0)) +
         geom_bar(stat = "identity") +
         theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
         ylab("Highest Weighted Positive/Negative Terms") +
         xlab("term") +
         ggtitle("Top Pos/Neg Sentiments - Republican (AFINN)") +
         theme(plot.title = element_text(hjust = 0.5)) +
         theme(plot.title = element_text(size=10)) +
         theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
         theme(legend.position="top") +
         theme(legend.title = element_text(size=8)) +
         theme(legend.text = element_text(size=6))

grid.arrange(d_a, r_a, ncol=2, nrow=1)
```




```
# Deeper dive into negative terms (AFINN)
afinn_neg_d <- sentiments_d_a %>%
       mutate(weighted_value = (count * value)) %>%
       filter(weighted_value <= -20) %>%
       arrange(weighted_value) %>%
       ggplot(aes(reorder(term, -weighted_value), weighted_value, fill = 'salmon')) +
         geom_bar(stat = "identity") +
```

```
        theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
        ylab("Highest Weighted Negative Terms") +
        xlab("term") +
        ggtitle("Neg Sentiments - Democrat (AFINN)") +
        theme(plot.title = element_text(hjust = 0.5)) +
        theme(plot.title = element_text(size=10)) +
        theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
        theme(legend.position = "none") +
        coord_flip()

afinn_neg_r <- sentiments_r_a %>%
      mutate(weighted_value = (count * value)) %>%
      filter(weighted_value <= -20) %>%
      arrange(weighted_value) %>%
      ggplot(aes(reorder(term, -weighted_value), weighted_value, fill = 'salmon')) +
        geom_bar(stat = "identity") +
        theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
        ylab("Highest Weighted Negative Terms") +
        xlab("term") +
        ggtitle("Neg Sentiments - Republican (AFINN)") +
        theme(plot.title = element_text(hjust = 0.5)) +
        theme(plot.title = element_text(size=10)) +
        theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
        theme(legend.position = "none") +
        coord_flip()

grid.arrange(afinn_neg_d, afinn_neg_r, ncol=2, nrow=1)
```



5. Compare and discuss the sentiments of these platforms: which party tends to be more optimistic about the future? Does this comport with your perceptions of the parties?

According to my analysis, the Democratic party tends to be more optimistic about the future than the Republican party. Specifically, the Democratic party had a higher, positive tone score using the Bing dictionary (.26) compared to the Republican party (0.12), as well as a higher relative score using the AFINN dictionary (.57 compared to .36). While both parties largely use a set of positive terms more frequently than a common set of negative terms in their platforms, the most commonly sited positive word in the Democratic platform (according to Bing) is support at greater than 120 occurances, while the most commonly sited positive word in the Republican platform is also support, at only ~100 occurances. Commonly used positive terms in the Democratic platform include protect, clean, and affordable, as compared to freedom, protect, and work in the Republican platform. Meanwhile, the Republican party uses the word 'oppose' over 40 times (Bing), while the Democratic party does not use any of the negative terms from the Bing dictionary over 20 times. The Republican party uses 20 negative terms from the Bing dictionary over 10 times, as compared to the Democratic party which uses only 9 negative terms over 10 times.

According to the AFINN dictionary, the Democratic party uses a greater variety of positive terms (selected from the most common set) than does the Republican party. The types of terms used by each party also differ - the most frequently cited negative terms for the Republican party include words like 'criminal' and 'illegal' which correspond with a theme of legality, while the most frequently cited negative terms for the Democratic party include words like 'violence', 'fight', and 'war' which correspond with a theme of divisiveness and fighting. Common positive themes in the Democratic platform (using AFINN) include empowerment and opportunity, while common positive themes in the Republican platform include growth and protection.

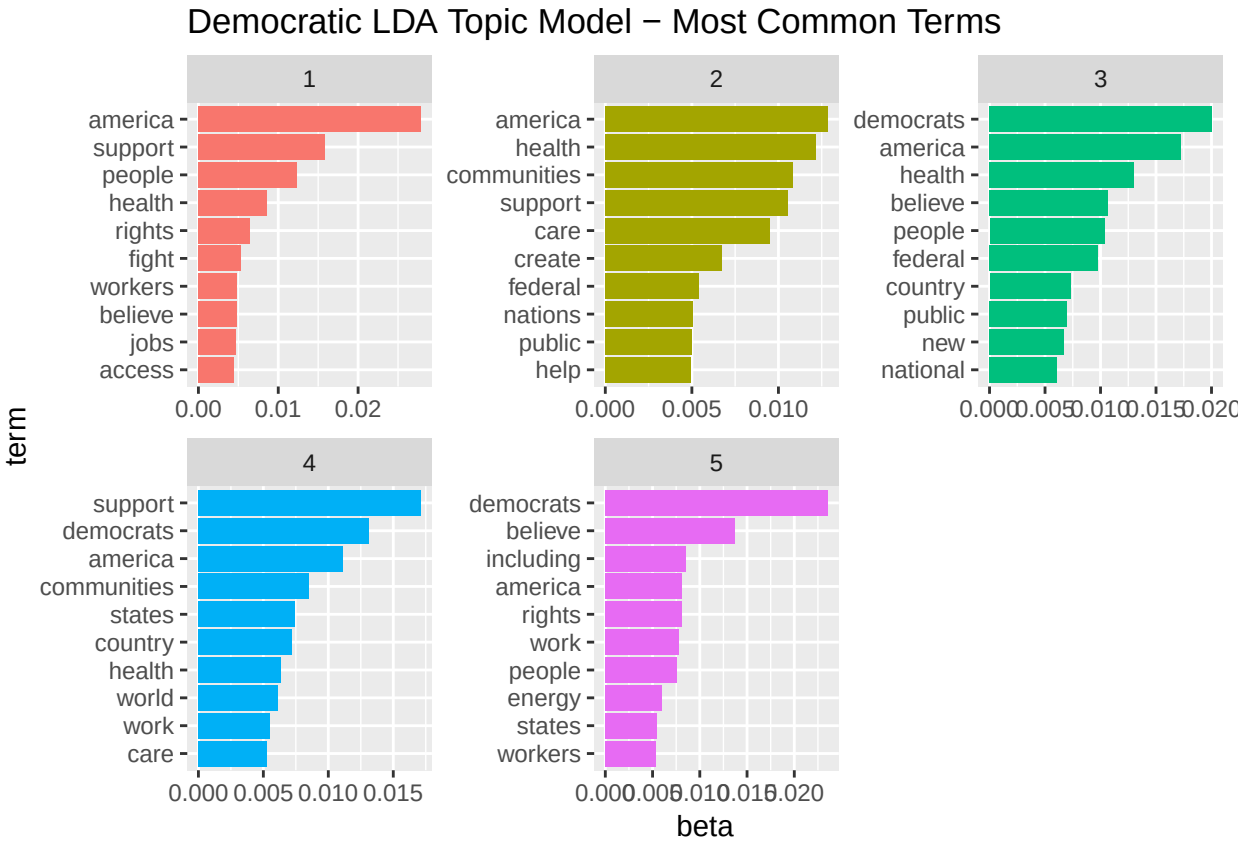
In general and as objectively as possible, I tend to think of the Democratic party as being more realistic, and therefore, at times, more pessimistic about the future of our country, while I tend to think of the

Republican party as being unrealistically optimistic, so these general findings are surprising. However, the specific themes extracted from the text do comport with my perceptions of the parties. I believe that the Democratic party focuses more on unity and resolving divisions between groups, while the Republican party focuses more on preserving institutions and criminalizing those who have broken the law. Similarly, I believe that the Democratic party values community empowerment while the Republican party more highly values economic growth. These themes can be extracted from the texts.

Topic Models

6. With a general sense of sentiments of the party platforms (ie. the tones related to how parties talk about their roles in the political future), now explore the topics they are highlighting in their platforms. This will give a sense of the key policy areas they're most interested in. Fit a topic model for each of the major parties (ie. two topic models) using the latent Dirichlet allocation algorithm, initialized at k=5 topics as a start. Present the results however you'd like (eg. visually and/or numerically).

```
# Set seed and create a 5-topic LDA model
d_lda <- LDA(dtm_d, k=5, control=list(seed=111))
d_topics <- tidy(d_lda, matrix='beta')

# Visualize the 10 terms most common within each topic
# Citation: https://www.tidytextmining.com/topicmodeling.html
d_top <- d_topics %>%
         group_by(topic) %>%
         top_n(10, beta) %>%
         ungroup() %>%
         arrange(topic, -beta) %>%
         mutate(term=reorder_within(term, beta, topic)) %>%
         ggplot(aes(term, beta, fill=factor(topic))) +
         geom_col(show.legend = FALSE) +
         facet_wrap(~ topic, scales='free') +
         coord_flip() +
         ggtitle("Democratic LDA Topic Model - Most Common Terms") +
         scale_x_reordered()
d_top
```



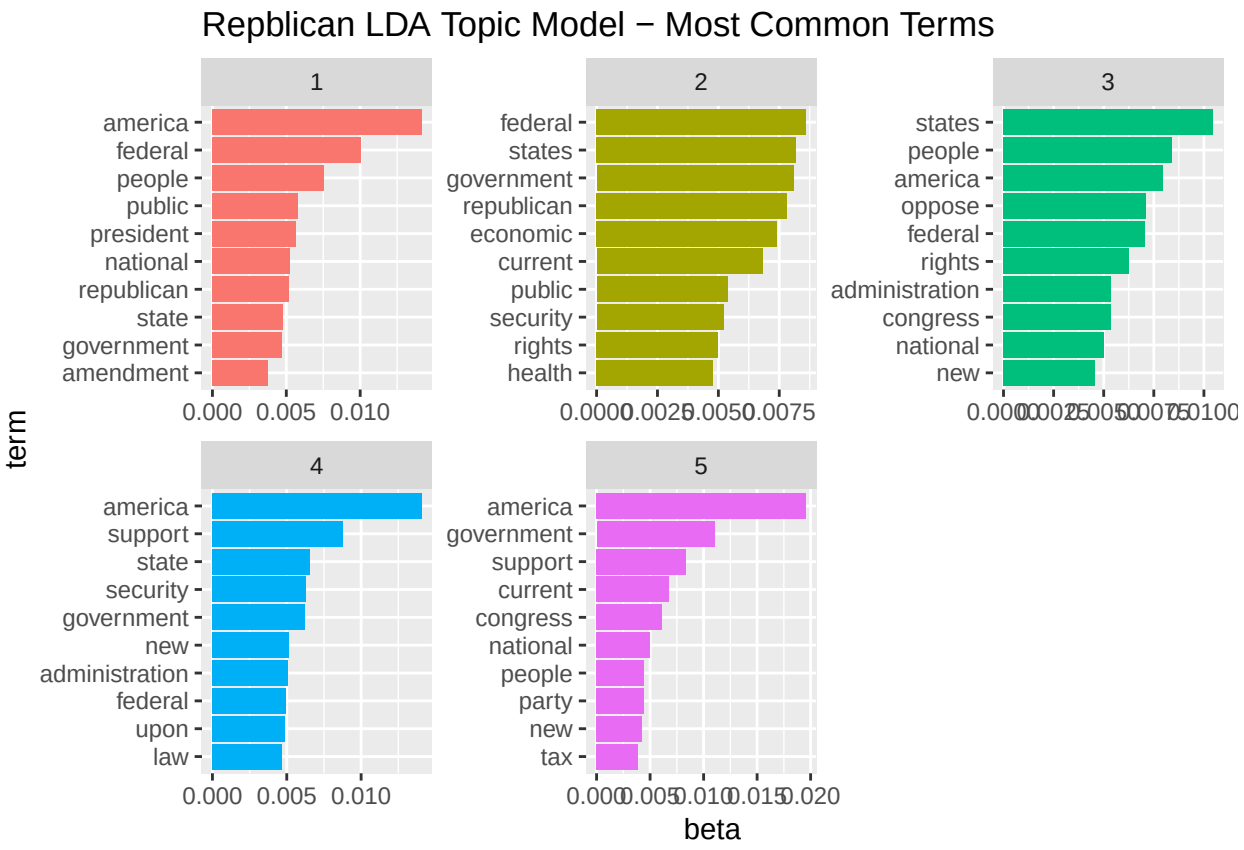

```
d_documents <- tidy(d_lda, matrix='gamma')
kable(d_documents[,2:3], caption="Democratic Topic Distribution: k=5")
```

Table 2: Democratic Topic Distribution: k=5

topic	gamma
1	0.1858133
2	0.1583558
3	0.2332862
4	0.1859996
5	0.2365452

```
# Set seed and create a 5-topic LDA model
r_lda <- LDA(dtm_r, k=5, control=list(seed=111))
r_topics <- tidy(r_lda, matrix='beta')

# Visualize the 10 terms most common within each topic
# Citation: https://www.tidytextmining.com/topicmodeling.html
r_top <- r_topics %>%
         group_by(topic) %>%
         top_n(10, beta) %>%
         ungroup() %>%
         arrange(topic, -beta) %>%
         mutate(term=reorder_within(term, beta, topic)) %>%
```

```
        ggplot(aes(term, beta, fill=factor(topic))) +
        geom_col(show.legend = FALSE) +
        facet_wrap(~ topic, scales='free') +
        coord_flip() +
        ggtitle("Repblican LDA Topic Model - Most Common Terms") +
        scale_x_reordered()
r_top
```




```
r_documents <- tidy(r_lda, matrix='gamma')
kable(r_documents[,2:3], caption="Republican Topic Distribution: k=5")
```

Table 3: Republican Topic Distribution: k=5

topic	gamma
1	0.2068831
2	0.2336585
3	0.1918160
4	0.1814805
5	0.1861620

7. Describe the general trends in topics that emerge from this stage. Are the parties focusing on similar or different topics, generally?

Based on the LDA topic model at k=5, a few key themes emerge. Based on the value of Beta, the Republican party may be focusing on the following topics: 1. National institutions (federal, president, government), 2. The law/rights (law, government, security), and 3. Identity (america, president, national, republican) (Note: due to common overlaps between topics, I included the three most unique ones of the five here). Meanwhile, the Democratic party may be focusing on the following topics: 1. Local communities (communities, states), 2. Livelihood (workers, jobs), and 3. Health (support, health, care, help) (similarly, due to common overlaps between topics, I included the three most unique ones of the five here). It appears that while both parties do have a strong focus on the American identity and promoting democracy, the Republic party tends to focus on patriotism and collective action, while the Democratic party tends to focus on public services, livelihood, and health. Therefore, I would classify the Republican rhetoric as more concerned with preserving the law, historical institutions, and the American identity, while the Democratic party may be more concerned with local reform and ensuring people's wellbeing (topic-based).

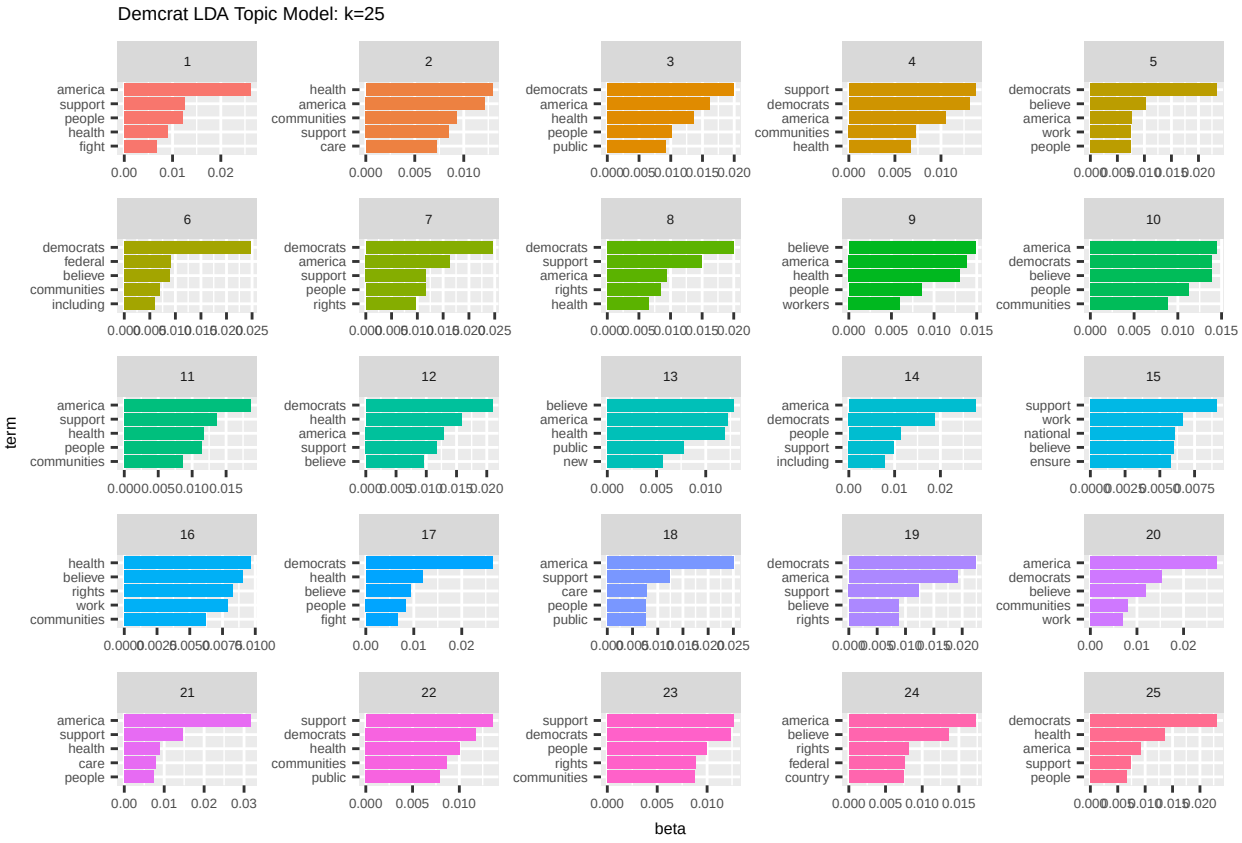
I would say that the parties are generally focusing on different themes, but that common topics between the two include jobs, health, and the American people. Finally, it is interesting to note that the model estimates that approximately 23% of the words in the Demcratic platform were generated from 3rd topic (focus: health, people, public) and 24% were generated rom the 5th topic (focus: workers, work, people), with a fairly even distribution among the other topics. The model estimates that approximately 23% of the words in the Republican platform were generated from the 2nd topic (focus: economic, security), with a fairly even distribution among the other topics.

8. Fit 6 more topic models at the following levels of k for each party: 5, 10, 25. Present the results however yo'd like (eg. visually and/or numerically).

```
# Democratic topic models
# k=5
d_lda_5 <- LDA(dtm_d, k=5, control=list(seed=111))
d_topics_5 <- tidy(d_lda_5, matrix='beta')
# k=10
d_lda_10 <- LDA(dtm_d, k=10, control=list(seed=111))
d_topics_10 <- tidy(d_lda_10, matrix='beta')
# k=25
d_lda_25 <- LDA(dtm_d, k=25, control=list(seed=111))
d_topics_25 <- tidy(d_lda_25, matrix='beta')

# Republican topic models
# k=5
r_lda_5 <- LDA(dtm_r, k=5, control=list(seed=111))
r_topics_5 <- tidy(r_lda_5, matrix='beta')
# k=10
r_lda_10 <- LDA(dtm_r, k=10, control=list(seed=111))
r_topics_10 <- tidy(r_lda_10, matrix='beta')
# k=25
r_lda_25 <- LDA(dtm_r, k=25, control=list(seed=111))
r_topics_25 <- tidy(r_lda_25, matrix='beta')
```

```
# Visualize Democrat, k=25
d_topics_25 %>%
        group_by(topic) %>%
```

```
        top_n(5, beta) %>%
        ungroup() %>%
        arrange(topic, -beta) %>%
        mutate(term=reorder_within(term, beta, topic)) %>%
        ggplot(aes(term, beta, fill=factor(topic))) +
        geom_col(show.legend = FALSE) +
        facet_wrap(~ topic, scales='free') +
        coord_flip() +
        ggtitle("Demcrat LDA Topic Model: k=25") +
        scale_x_reordered() +
        theme(text=element_text(size=6))
```



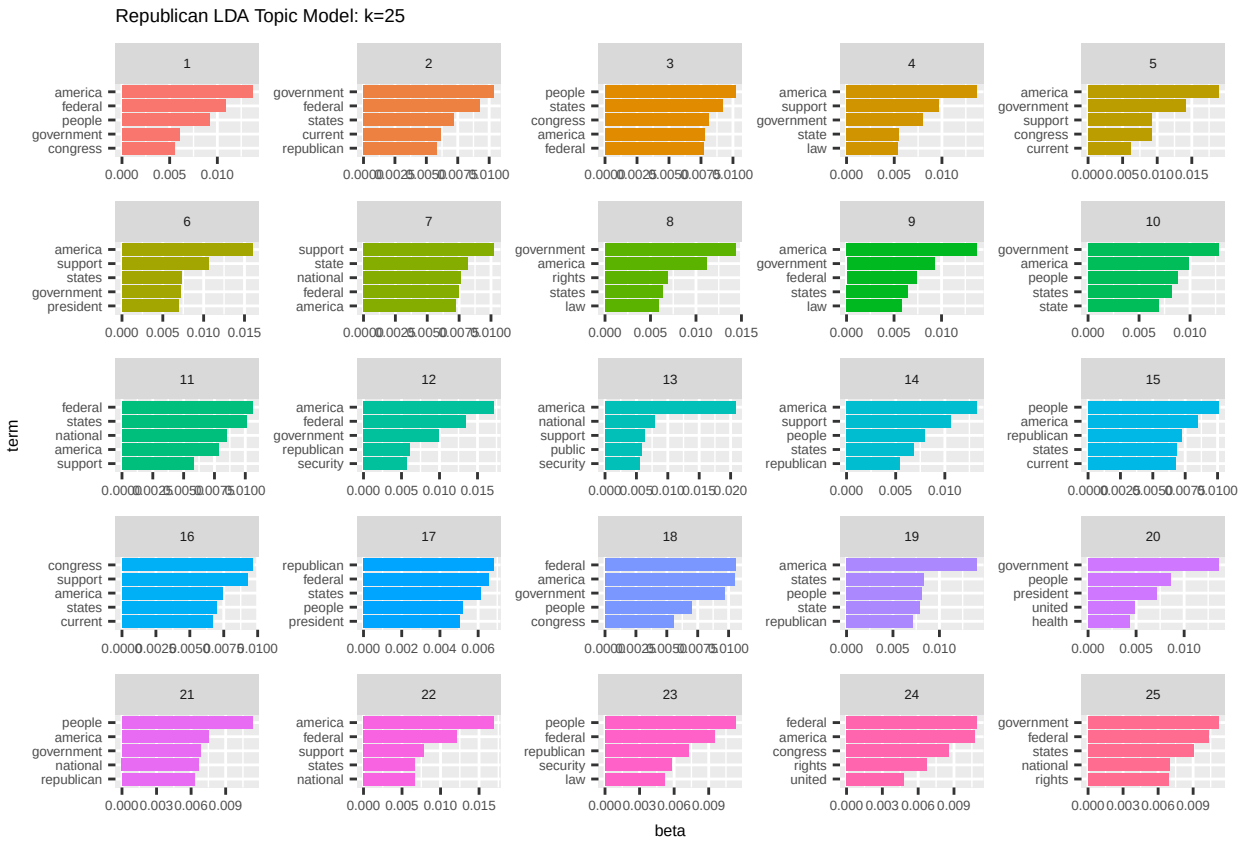

```
d_documents_25 <- tidy(d_lda_25, matrix='gamma')
kable(d_documents_25[,2:3], caption="Democratic Topic Distribution: k=25")
```

Table 4: Democratic Topic Distribution: k=25

topic	gamma
1	0.0398433
2	0.0327209
3	0.0660944
4	0.0359416
5	0.0533833
6	0.0334375
7	0.0468964

topic	gamma
8	0.0310043
9	0.0313974
10	0.0361580
11	0.0398647
12	0.0743539
13	0.0296307
14	0.0351501
15	0.0308905
16	0.0302484
17	0.0354335
18	0.0608167
19	0.0351057
20	0.0372867
21	0.0438028
22	0.0316704
23	0.0350266
24	0.0403028
25	0.0335393

```r
# Visualize Republican, k=25
r_topics_25 %>%
        group_by(topic) %>%
        top_n(5, beta) %>%
        ungroup() %>%
        arrange(topic, -beta) %>%
        mutate(term=reorder_within(term, beta, topic)) %>%
        ggplot(aes(term, beta, fill=factor(topic))) +
        geom_col(show.legend = FALSE) +
        facet_wrap(~ topic, scales='free') +
        coord_flip() +
        ggtitle("Republican LDA Topic Model: k=25") +
        scale_x_reordered() +
        theme(text=element_text(size=6))
```



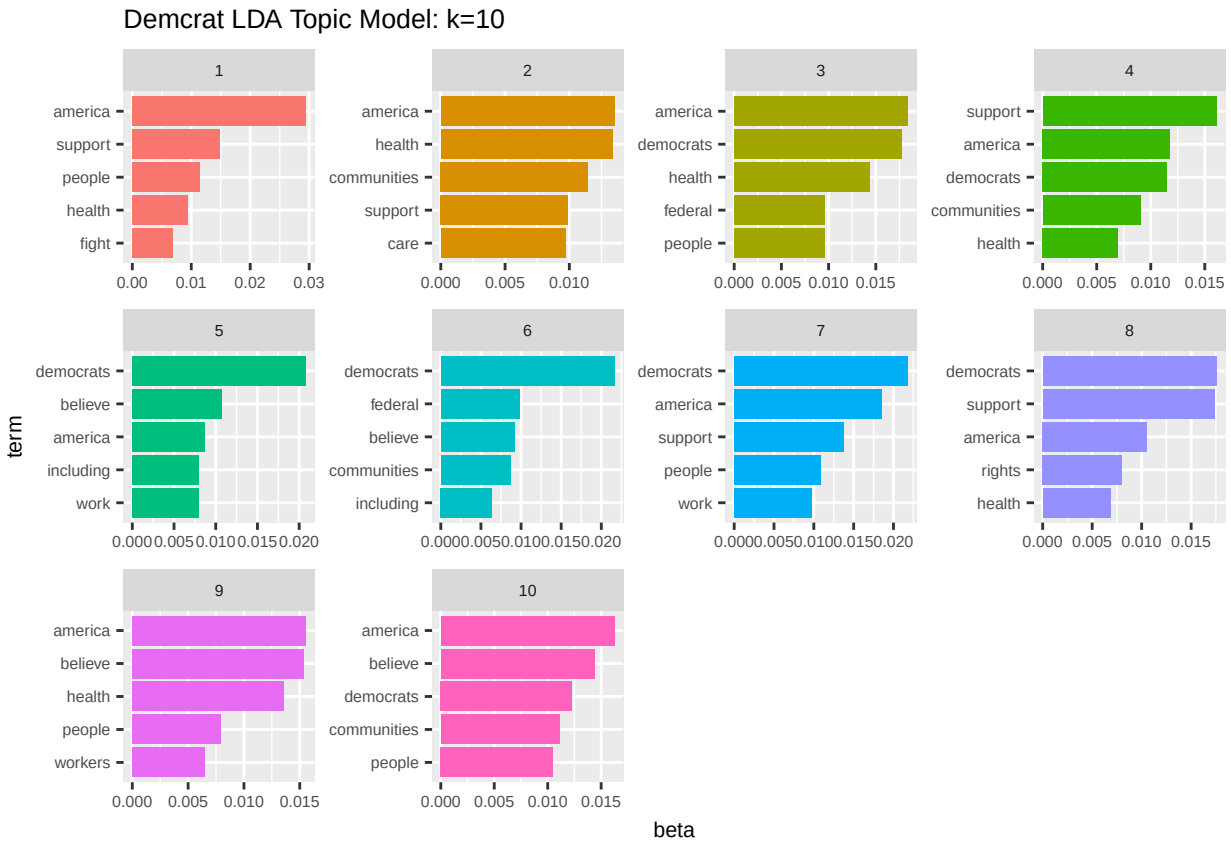

```
r_documents_25 <- tidy(r_lda_25, matrix='gamma')
kable(r_documents_25[,2:3], caption="Republican Topic Distribution: k=25")
```

Table 5: Republican Topic Distribution: k=25

topic	gamma
1	0.0429819
2	0.0480617
3	0.0373719
4	0.0392318
5	0.0403076
6	0.0458069
7	0.0376379
8	0.0390024
9	0.0389077
10	0.0388777
11	0.0400910
12	0.0447995
13	0.0462580
14	0.0383499
15	0.0379091
16	0.0380075
17	0.0370373
18	0.0389031
19	0.0383178
20	0.0365076

topic	gamma
21	0.0389624
22	0.0416744
23	0.0370013
24	0.0381315
25	0.0398622

```
# Visualize Democrat, k=10
d_topics_10 %>%
     group_by(topic) %>%
     top_n(5, beta) %>%
     ungroup() %>%
     arrange(topic, -beta) %>%
     mutate(term=reorder_within(term, beta, topic)) %>%
     ggplot(aes(term, beta, fill=factor(topic))) +
     geom_col(show.legend = FALSE) +
     facet_wrap(~ topic, scales='free') +
     coord_flip() +
     ggtitle("Demcrat LDA Topic Model: k=10") +
     scale_x_reordered() +
     theme(text=element_text(size=8))
```

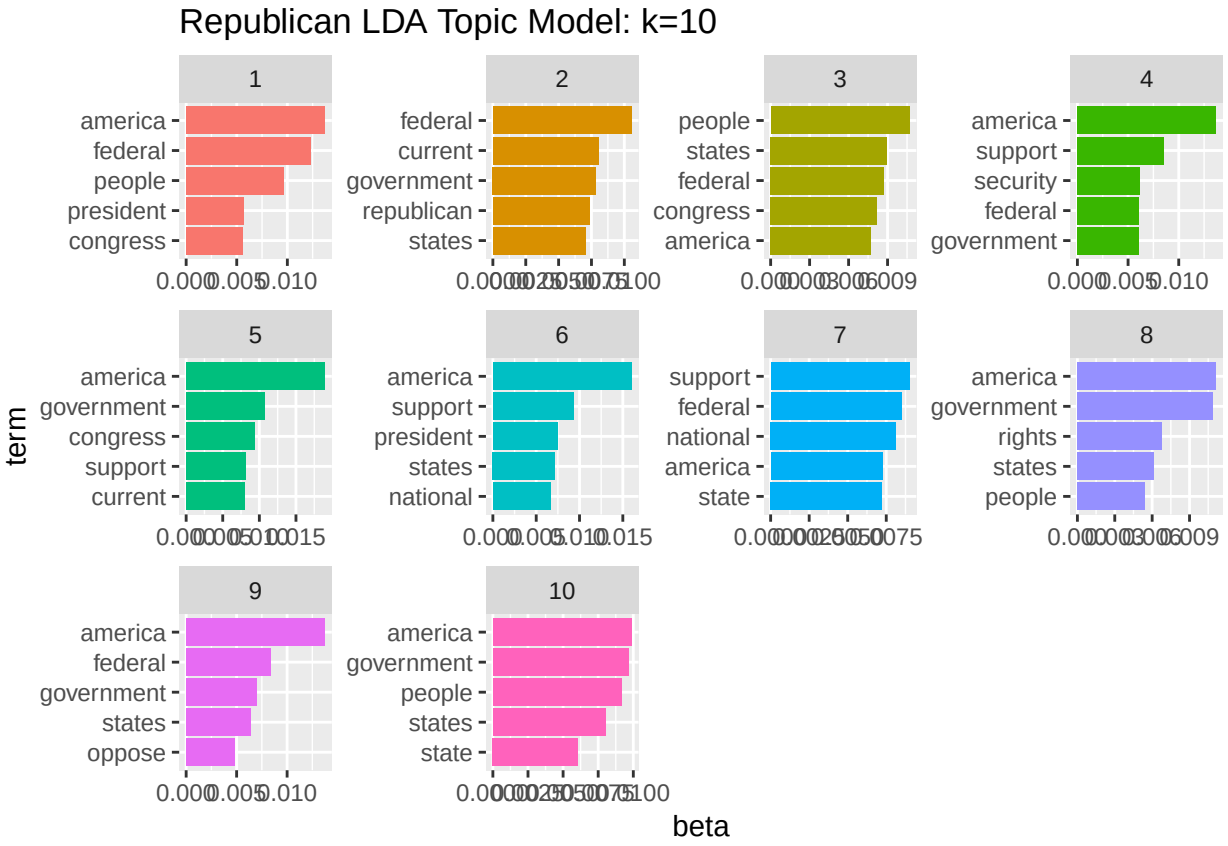



```
d_documents_10 <- tidy(d_lda_10, matrix='gamma')
kable(d_documents_10[,2:3], caption="Democratic Topic Distribution: k=10")
```

Table 6: Democratic Topic Distribution: k=10

topic	gamma
1	0.1058375
2	0.0770175
3	0.1474653
4	0.1008934
5	0.1350758
6	0.0709182
7	0.1173511
8	0.0753458
9	0.0757122
10	0.0943832

```
# Visualize Republican, k=10
r_topics_10 %>%
         group_by(topic) %>%
         top_n(5, beta) %>%
         ungroup() %>%
         arrange(topic, -beta) %>%
         mutate(term=reorder_within(term, beta, topic)) %>%
         ggplot(aes(term, beta, fill=factor(topic))) +
         geom_col(show.legend = FALSE) +
         facet_wrap(~ topic, scales='free') +
         coord_flip() +
         ggtitle("Republican LDA Topic Model: k=10") +
         scale_x_reordered() +
         theme_update(text = element_text(size=8))
```




```
r_documents_10 <- tidy(r_lda_10, matrix='gamma')
kable(r_documents_10[,2:3], caption="Republican Topic Distribution: k=10")
```

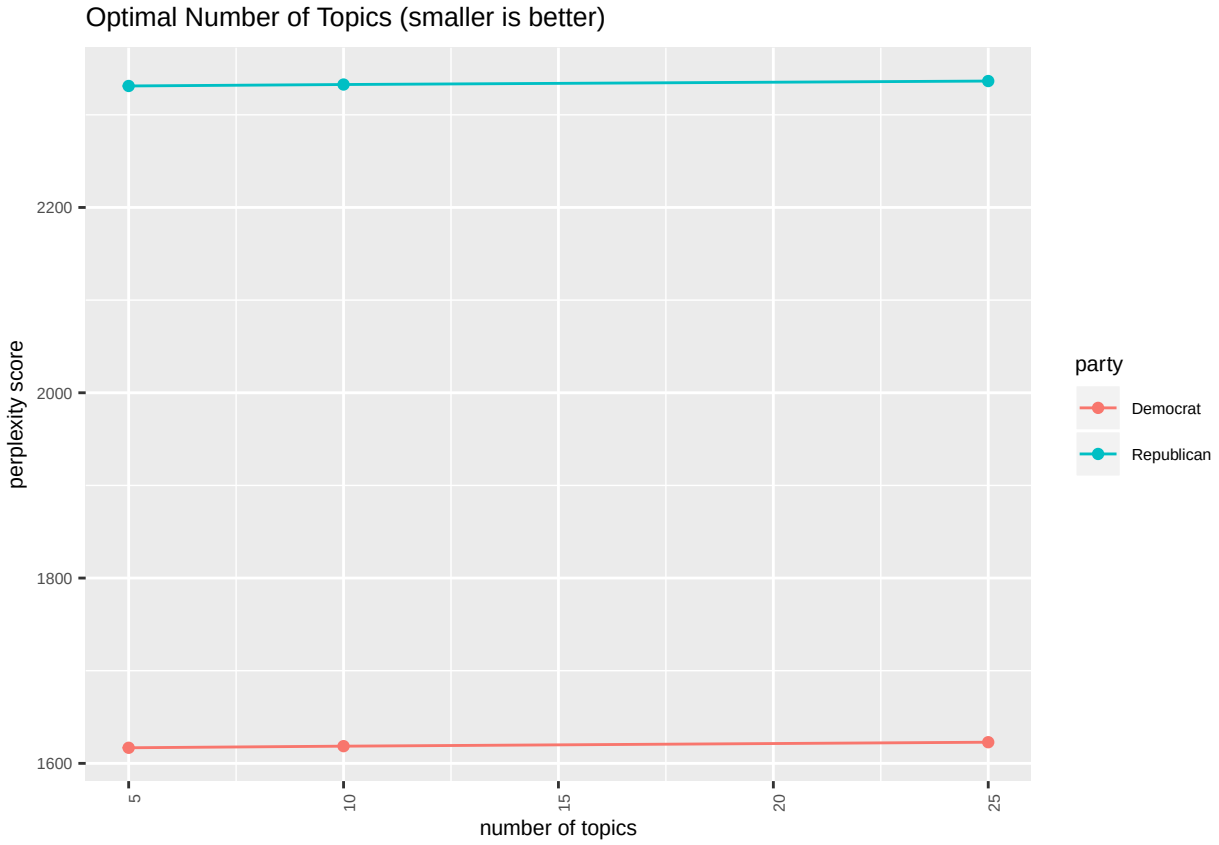
Table 7: Republican Topic Distribution: k=10

topic	gamma
1	0.1084265
2	0.1317343
3	0.0944904
4	0.0896342
5	0.0929721
6	0.1102843
7	0.0894051
8	0.0946160
9	0.0906151
10	0.0978219

Note: k=5 is presented in question 6.

9. Calculate the perplexity of each model iteration and describe which technically fits the best.

```
# Calculate perplexity of each model
p_d_5 <- perplexity(d_lda_5)
p_d_10 <- perplexity(d_lda_10)
p_d_25 <- perplexity(d_lda_25)
p_r_5 <- perplexity(r_lda_5)
p_r_10 <- perplexity(r_lda_10)
p_r_25 <- perplexity(r_lda_25)

# Calculate perplexity of each model and store in data frame
results <- data.frame("party" = c("Democrat", "Democrat", "Democrat",
                                  "Republican", "Republican", "Republican"),
                      "k" = c(5, 10, 25, 5, 10, 25),
                      "perplexity" = c(p_d_5, p_d_10, p_d_25, p_r_5, p_r_10, p_r_25))

results %>% group_by(party, k, perplexity) %>%
            summarize(perplexity_score = perplexity) %>%
            ggplot(aes(x=k, y=perplexity_score, group=party, color=party)) +
                  geom_line() +
                  geom_point() +
                  ggtitle("Optimal Number of Topics (smaller is better)") +
                  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
                  xlab("number of topics") +
                  ylab("perplexity score")
```



```
kable(results, caption="Perplexity score by model specification")
```

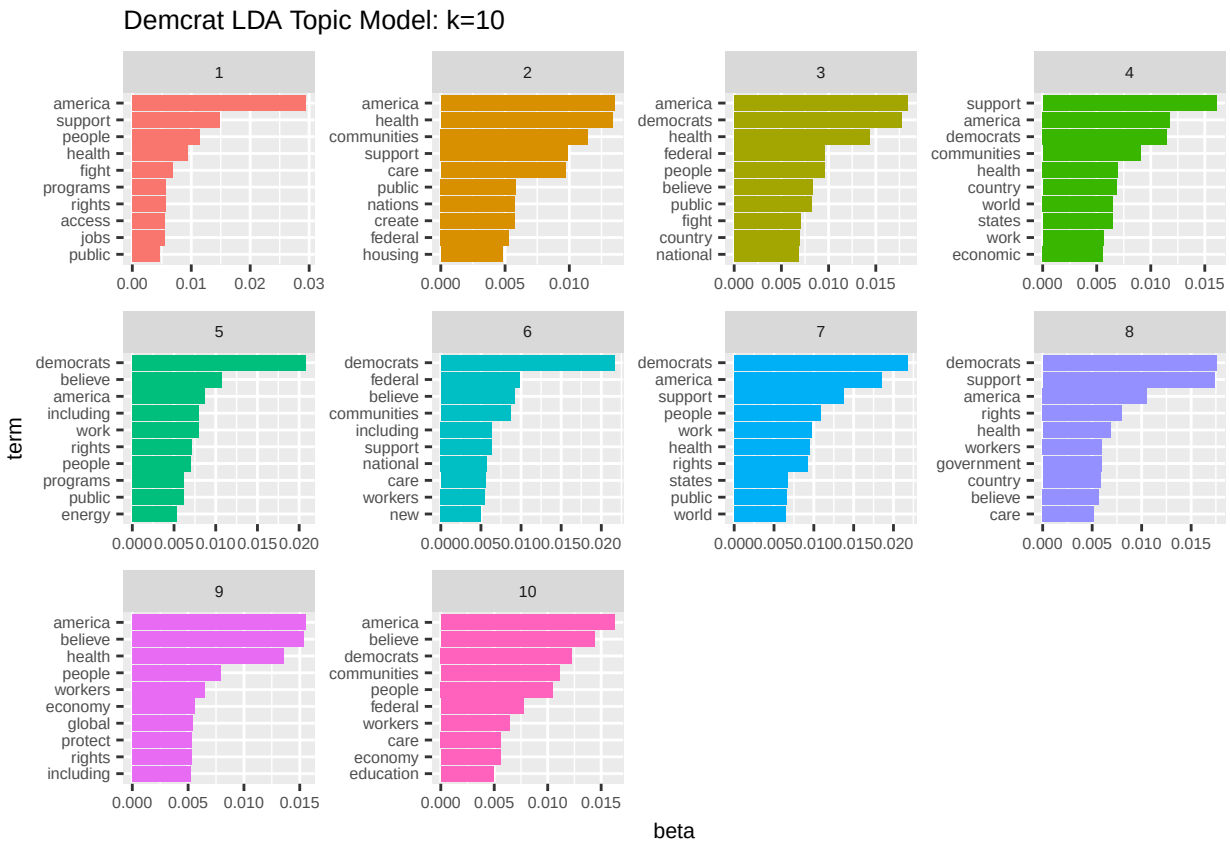
Table 8: Perplexity score by model specification

party	k	perplexity
Democrat	5	1616.817
Democrat	10	1618.557
Democrat	25	1622.817
Republican	5	2331.134
Republican	10	2332.691
Republican	25	2336.435

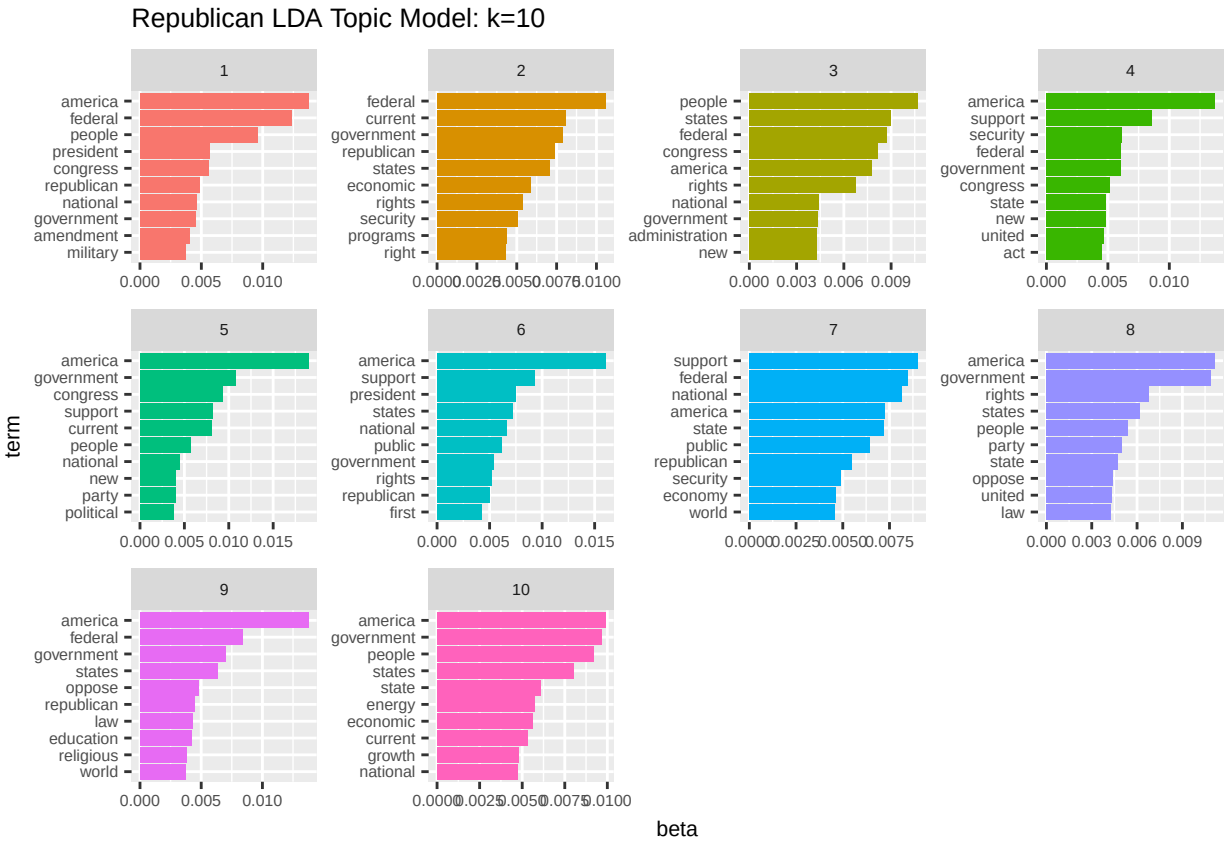
Technically, the model for each party platform that fits the best (in this case!) is the one with the lowest perplexity. While the perplexity is close for all 3 specifications (k=5, k=10, k=25), the model with the lowest score for both platforms is k=5 indicating that it is the best fit, but only marginally. Perplexity is a measure of how well a model predicts a sample, and is equal to exp(-log likelihood(w) / Number of tokens). This finding, therefore, is counterintuitive, because perplexity should decrease as the number of topics increases, similarly to the way R2 increases in a statistical model with each added explanatory variable due to increased proportion of variance explained.

10. Building on the previous question, display a barplot of the k=10 model for each party, and offer some general inferences as to the main trends that emerge. Are there similar themes between the parties? Do you think k=10 likely picks up differences more efficiently? Why or why not?

```
# Visualize Democrat, k=10
d_topics_10 %>%
     group_by(topic) %>%
     top_n(10, beta) %>%
     ungroup() %>%
     arrange(topic, -beta) %>%
     mutate(term=reorder_within(term, beta, topic)) %>%
     ggplot(aes(term, beta, fill=factor(topic))) +
     geom_col(show.legend = FALSE) +
     facet_wrap(~ topic, scales='free') +
     coord_flip() +
     ggtitle("Demcrat LDA Topic Model: k=10") +
     scale_x_reordered() +
     theme_update(text = element_text(size=8))
```




```
# Visualize Republican, k=10
r_topics_10 %>%
          group_by(topic) %>%
          top_n(10, beta) %>%
          ungroup() %>%
          arrange(topic, -beta) %>%
          mutate(term=reorder_within(term, beta, topic)) %>%
          ggplot(aes(term, beta, fill=factor(topic))) +
          geom_col(show.legend = FALSE) +
          facet_wrap(~ topic, scales='free') +
          coord_flip() +
          ggtitle("Republican LDA Topic Model: k=10") +
          scale_x_reordered() +
          theme_update(text = element_text(size=8))
```



Examining the democratic platform at k=10, it appears that several of the topics share similar themes centering around America, the public, access to programming, and healthcare. Looking at the most common terms in each topic, is difficult to discern clear distinctions between the topics based only on these terms, but it is likely that less common terms may be driving some of the dissimilarity. Other themes in the Democratic platform at k=10 include rights and the law, as well as the global economy. The Republican party displays similar themes at k=10. Similarly, there is a general emphasis on institutions like the administration and on the law/rights, with a heavier emphasis on the miliary and security that is less present in the Democratic platform.

Given the similarity between the topics in each platform as well as across platforms, I believe that k=10 may represent too many topics to be generalizable. In other words, there is significant overlap between topics, and it is possible that a lower value of k may result in more clearly distinct topics, such as public service provision, security/defense, and the economy/labor market. In this case (counterintuitively), because lower values of k have lower perplexity scores and higher log-likelihood ratios, I believe that lower values of k may be more efficient at picking up major differences between topics in a way that is easy to interpret. While high values of k may overfit a particular dataset and perform well using a training dataset, models with high values of k may be less generalizable to new corpuses and therefore perform poorly with testing datasets. In this case, k=10 results in similar themes between parties and topics, and may only pick up nuances among terms that appear infrequently in the platforms (like Zika).

Conclusion

11. Per the opening question, based on your analyses (including exploring party brands, general tones/sentiments, political outlook, and policy priorities), which party would you support in the 2020 election (again, this is hypothetical)?

Based entirely on this analysis, I would support the Democratic party in the 2020 selection. First, based on EDA and exploring the word clouds/brands for each of the platforms, I noticed that the Democratic party strongly emphasizes the topic of health and the theme of community, with a secondary emphasis on social services, climate, education, and the economy. I am personally and academically motivated by social reform, specifically in the areas of education, public health, and child protection, so these themes are important to me. I am less politically concerned with the topics of trade and business, which were mentioned far more infrequently in this platform. By comparison, the Republican party platform emphasizes institutions and security. I believe that institutions should continuously be reformed over time, making me less concerned with the preservation of these institutions for the sake of the American identity. The Republican platform infequently mentions foreign affairs, which is a chief concern of mine. I have lived in over 6 countries and visited over 35, so I support a party that maintains a global focus.

From a general tone/sentiment standpoint, I would also support the Democratic party in the 2020 election. The Democratic party tends to be more optimistic about the future than the Republican party (based on sentiment scores using the Bing and AFINN dictionaries), and uses a greater variety of positive terms. In general, I am optimistic about the future and objectively would align myself with the more optimistic party. More specifically, the Republican party platform's frequent negative terms includes words like 'criminal' and 'illegal' which correspond with a theme of legality, while the Democratic party platform's frequent negative terms focus on 'violence' and 'fight', which correspond with a theme of division. I am more concerned with preserving peace than criminalizing, and therefore feel that I more closely align with the Democratic party based on this aspect of the party's political outlook.

Finally, from a policy priority perspective, the Republican party seems to focus more on national institutions, the law/rights, and the American identity, while the Democratic party tends to focus on local communities, livelihood, and healthcare. Thematically, I am more politically concerned with job creation, healthcare, education, and foreign affairs, which are priority areas that tend to more closely align with those expressed in the Democratic platform. However, I would definitely want to do a deeper analysis, working with different methods, models, and values of k, to better understand the topics that each platform is most concerned with to make a final decision. Based on this first analysis focused on party brands, tones/sentiments, political outlook, and policy priorities, I feel comfortable moving forward in support of the Democratic party in the 2020 election.