# Arroyo - Problem Set 5

*Pedro Alberto Arroyo*

*11/25/2019*

## PREPROCESSING & (light) EDA

### Load .csv files

```
platforms <- read_csv("platforms.csv")
```

```
## Parsed with column specification:
## cols(
##   party = col_character(),
##   platform = col_character()
## )
```

### Load the .txt files

```
texts <- file.path("~", "Documents/rstudiodef/Problem_Sets/Problem_Set_5", "texts")
dir(texts)
```

```
## [1] "d16.txt" "r16.txt"
```

### Create DTM and Pre-Process Platforms

```
docs <- VCorpus(DirSource(texts)) %>%
  tm_map(removePunctuation) %>%
  tm_map(removeNumbers) %>%
  tm_map(tolower)  %>%
  tm_map(removeWords, stopwords("english")) %>%
  tm_map(stripWhitespace) %>%
  tm_map(stemDocument) %>%
  tm_map(PlainTextDocument)

for (j in seq(docs)) {
  docs[[j]] <- gsub("/", " ", docs[[j]])
  docs[[j]] <- gsub("-", " ", docs[[j]])
}

docs <- tm_map(docs, PlainTextDocument)

docs <- tm_map(docs, PlainTextDocument)

tdm <- TermDocumentMatrix(docs) %>%
  as.matrix()

#writeLines(as.character(docs[1]))

colnames(tdm) <- c("DEM","GOP")
```

**Visual Inspection With Wordclouds**

```
DEM <- as.matrix(tdm[,1])
DEM <- as.matrix(DEM[order(DEM, decreasing=TRUE),])

GOP <- as.matrix(tdm[,2])
GOP <- as.matrix(GOP[order(GOP, decreasing=TRUE),])

DEM <- as.matrix(tdm[,1])
DEM <- as.matrix(DEM[order(DEM, decreasing=TRUE),])

GOP <- as.matrix(tdm[,2])
GOP <- as.matrix(GOP[order(GOP, 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 (e.g., 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?*

I'm really intrigued by the fact that 'will' is the prominent word in the democratic wordcloud; it really drives home the idea that, at least in these instances, the platform was about making promises to voters.

The democratic wordcloud also has 'democrats' as a prominent term, which I take as a reflection that the democratic party put on branding itself. I do not know if this is a general trend in platforms because these are the only two platforms I've examined this way, but I do think it makes sense that the GOP wordcloud would not exhibit that same trait given that their nominee was, at the time, a party outsider who had run, in great part, against party orthodoxy.

Maybe that also helps explain another trend: the relatively higher prominence of issue-terms in the democratic wordcloud when compared to the republican wordcloud. So far, this pattern conforms to my expectation based on party branding.

Something else sort of jumps out at me. Many moons ago, I did a textual analysis of newspaper coverage over a series of years in which I compared British coverage of the EU, the WTO, the UN, and NATO using topic modeling. The major finding from that was that the language around the EU was focused on insitutions and process, while the language around the other institutions was focused on policies and outcomes. The conclusion that I drew at the time was that the EU was still a contested space, while the other institutions where more established as actors. (It would be interesting to do a comparison now and see if that patter still holds. I suspect it would not hold for the WTO or, more recently, NATO.)

These wordclouds remind me of that: the democrats were focused on outcomes, the republicans were focused on the institutions of governance: federal, government, state, congress, president, military.

It's a thin reed at this point, but it's intriguing.

```
wordcloud(rownames(DEM), DEM, min.freq =50,
          random.order = FALSE, random.color = FALSE,
          colors= c("navy", "royalblue2", "midnightblue"))
```


```
wordcloud(rownames(GOP), GOP, min.freq =50, scale=c(2, .2),
          random.order = FALSE, random.color = FALSE,
          colors= c("red4", "darkred", "firebrick4"))
```

## SENTIMENT ANALYSIS

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

```
platforms <- read_csv("platforms.csv")
```

```
## Parsed with column specification:
## cols(
##   party = col_character(),
##   platform = col_character()
## )
```

```
xx <- platforms %>%
  dplyr::select(platform)

Party <- c("Dem", "GOP")

xx <- mutate(xx, Party)
colnames(xx) <- c("Text","Party")

(un_p <- xx %>%
    unnest_tokens(output = word,
                  input = Text))
```

```
## # A tibble: 10,274 x 2
##    Party word
##    <chr> <chr>
##  1 Dem   in
##  2 Dem   2016
##  3 Dem   democrats
##  4 Dem   meet
##  5 Dem   in
##  6 Dem   philadelphia
##  7 Dem   with
##  8 Dem   the
##  9 Dem   same
## 10 Dem   basic
## # ... with 10,264 more rows
```

```
as_tibble(stop_words)
```

```
## # A tibble: 1,149 x 2
##    word        lexicon
##    <chr>       <chr>
##  1 a           SMART
##  2 a's         SMART
##  3 able        SMART
##  4 about       SMART
##  5 above       SMART
##  6 according   SMART
##  7 accordingly SMART
##  8 across      SMART
##  9 actually    SMART
## 10 after       SMART
## # ... with 1,139 more rows
```

```
stops_p <- un_p %>%
  anti_join(stop_words,
            by = "word")

stops_p %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE)
```

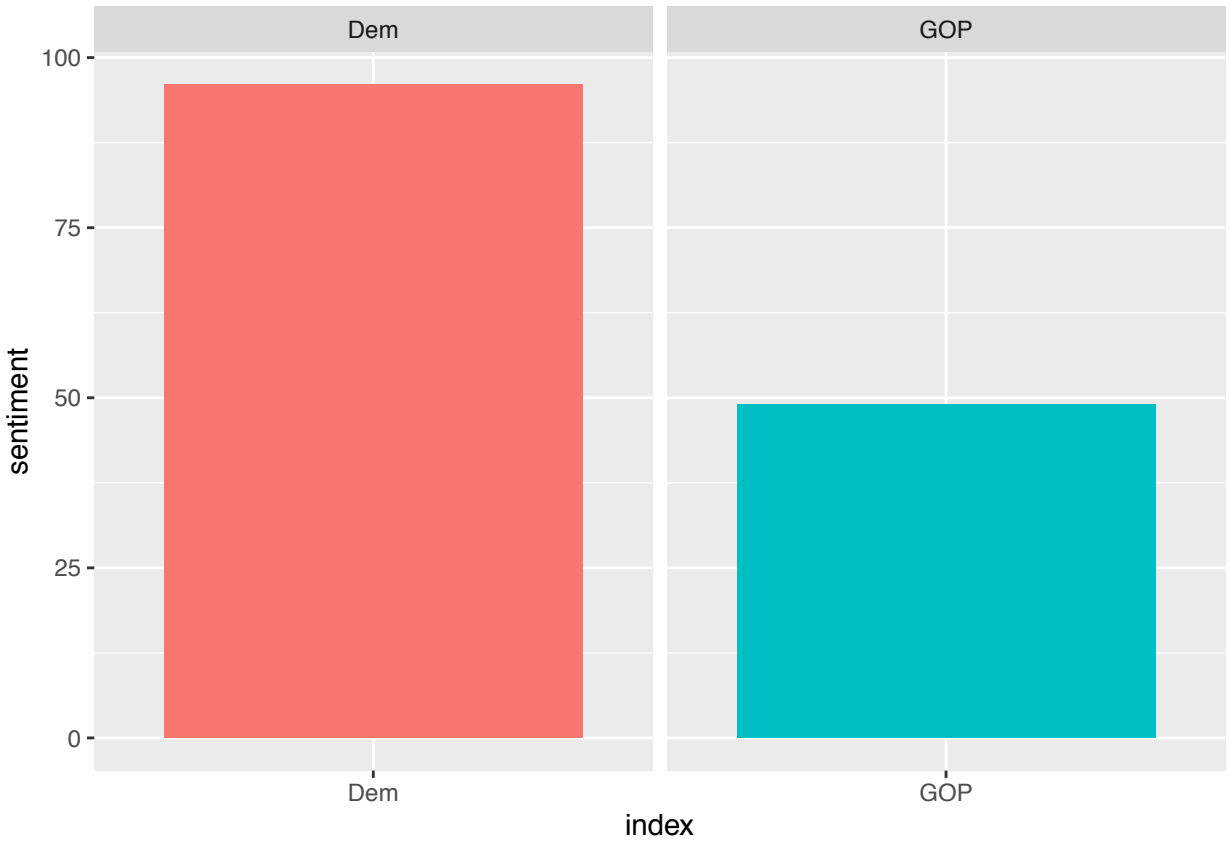
```
## Joining, by = "word"
```

```
## # A tibble: 361 x 3
##    word       sentiment     n
##    <chr>      <chr>     <int>
##  1 support    positive     31
##  2 innovation positive     19
##  3 affordable positive     11
##  4 fair       positive     10
##  5 stronger   positive      9
##  6 top        positive      9
##  7 benefits   positive      8
##  8 crisis     negative      7
##  9 free       positive      7
## 10 freedom    positive      7
## # ... with 351 more rows
```

```
p_sentiment <- stops_p %>%
  inner_join(get_sentiments("bing")) %>%
  count(Party, index = Party, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
## Joining, by = "word"
```

```
ggplot(p_sentiment, aes(index, sentiment, fill = Party)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~Party, ncol = 2, scales = "free_x")
```




*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?*

The sentiment analysis seems to indicate that the democratic platform was more possitive in sentiment than the GOP platform. That's a result that simultaniously surprises me and doesn't surprise me. I'm not surprised that the Democrats were positive given than the incumbent was a democrat and they framed their electoral strategy as (largely) a continuation of his policies. I'm also not surprised that the party that generally advocates for more affirmative federal policy would, in what amounts to policy wishlist, emphasize the positive.

At the same time, I'm not surprised that the Trump-led GOP found less reason for optimism; an outcome that feels over-determined. On the one hand, as the party out of power, there's value to be gained in playing up the country's woes. Similarly, Trump's fire and brimstone approach doesn't lend itself to a whole lot of possitivity.

But there's a counter-narritive here, too. Aren't Republicans supposed to be the "can-do" party? This is hardly Reagan's morning-in-America message. I'd love to compare (and will compare when I have caught up on sleep) the sentiment scores of Reagan's famous speech with Mario Cuomo's two-cities speech.

But that comparison also prompts me to notice that I have never read a party platform. My gut-sense for what I expect these to look like is convention speeches. I have no idea if that's a helpful comparison.

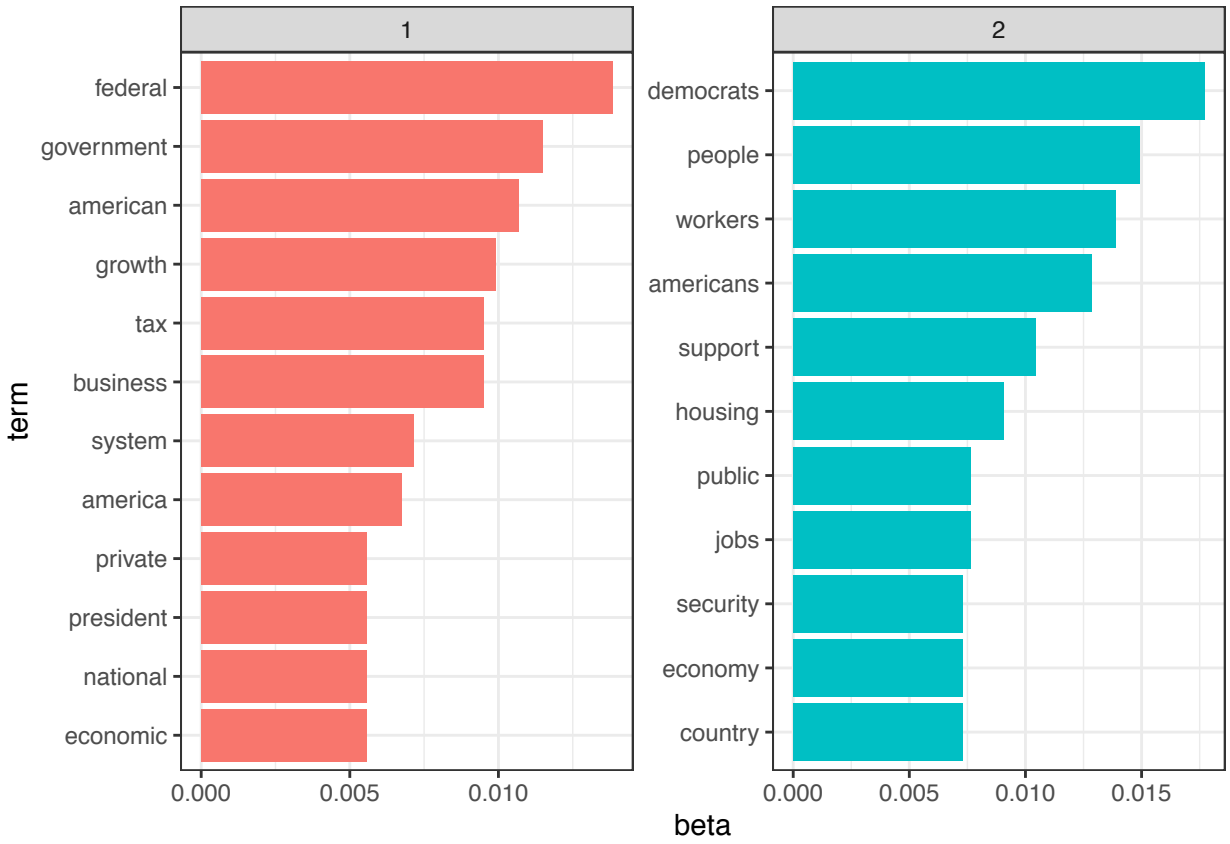
It's also really hard to get leverage with a comparison like this between just two texts. But it would be interesting to track sentiment from convention to convention, perhaps focusing on prime-time speeches, and see if there's a pattern in terms of how economic conditions, polling, recent electoral success, and so on shape the emotional valance of speeches.

Truthfully, I've always been sort of skeptical of sentiment analysis - probably because I've only ever seen bad examples of it. But, after playing with it a bit, I can imagine how I would want to use this in a project.

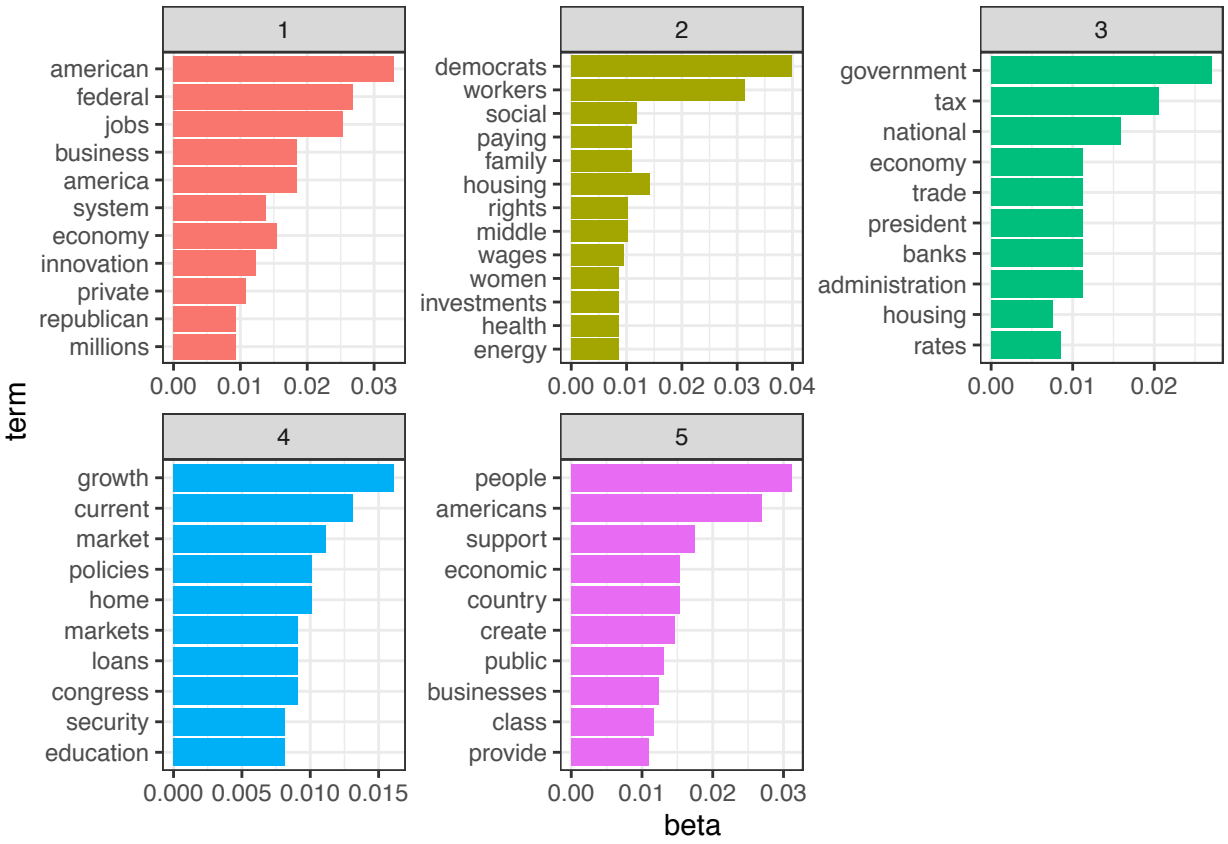
## TOPIC MODELS

I have to admit defeat here: I wasn't able to get the topic models to work. The stumbling block was figuirng out how to fit a model on just one text. I'll include below what I did do, which was fit a few models on the corpus with both texts. There are some patterns there, but I won't try to make too much of them since it wasn't what the question actually asked. I will say this, though: fitting the models on this corpus was a nice reminder of the value that comes with aiming at a target-rich enviornment. The topics that emerge from these models seem really clean, which shouldn't be too sorprising since platforms are, essentially, aggregations of party preference that are created through reconciling disperate interest groups and having folks go back and forth in committee and between committees. We talked earlier in the class about the assumptions in topic modeling; namely that texts are derived from the topics. Party platforms might be pretty close to the platonic ideal of that process.

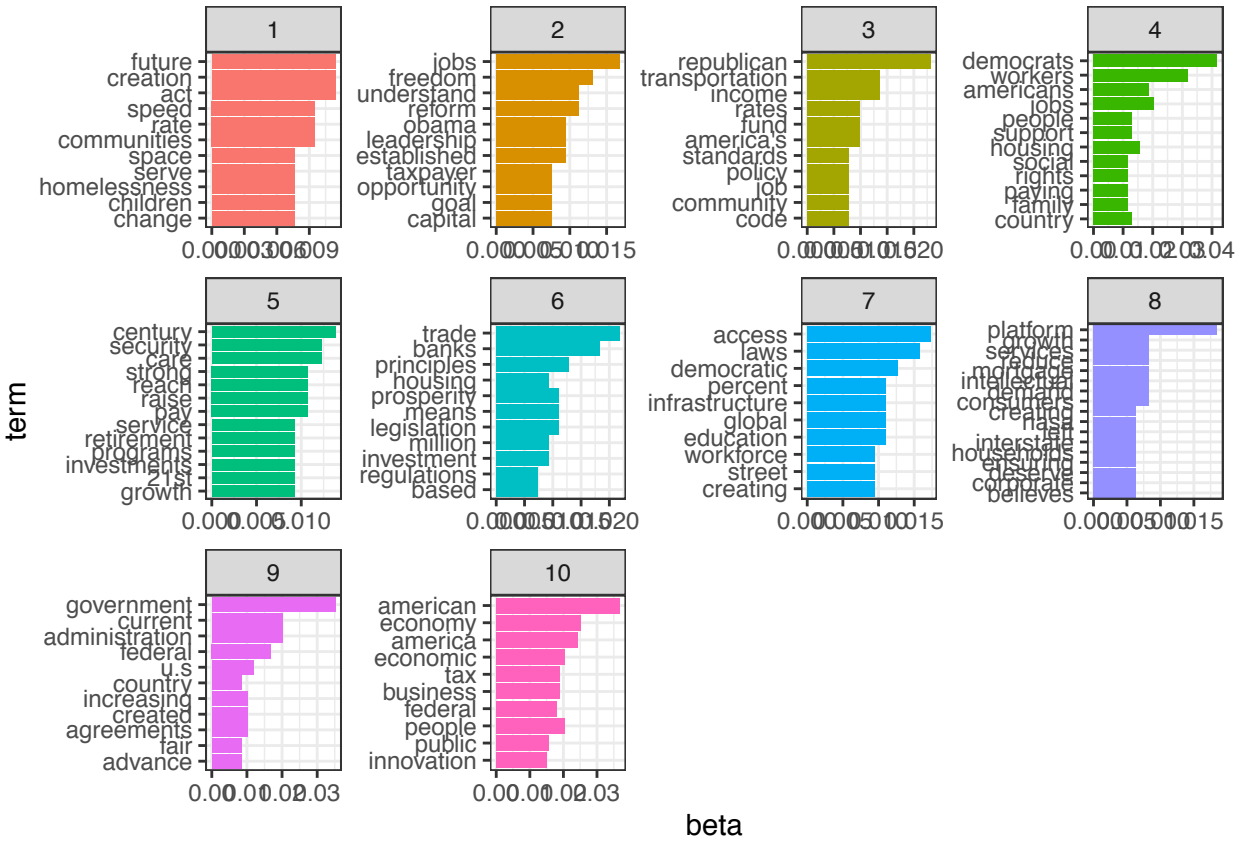
```
deet <- stops_p %>%
  filter(Party == "Dem" || "GOP") %>%
  group_by(Party) %>%
  count(word)

dtm <- cast_dtm(deet, Party, word, n)

lda_A <- LDA(dtm, k = 2,
                  method = "Gibbs", control =list(seed = 78,
                                                  verbose = 0))
topics <- tidy(lda_A,
               matrix = "beta")

terms <- topics %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)

terms %>%
  mutate(term = reorder(term, beta)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip() +
  theme_bw()
```




```
lda_B <- LDA(dtm, k = 5,
             method = "Gibbs", control =list(seed = 78,
                                             verbose = 0))
topics <- tidy(lda_B,
               matrix = "beta")

terms <- topics %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)

terms %>%
  mutate(term = reorder(term, beta)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip() +
  theme_bw()
```




```
lda_C <- LDA(dtm, k = 10,
             method = "Gibbs", control =list(seed = 78,
                                             verbose = 0))
topics <- tidy(lda_C,
               matrix = "beta")

terms <- topics %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)

terms %>%
  mutate(term = reorder(term, beta)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip() +
  theme_bw()
```


1
future
creation
act
speed
rate
communities
space
serve
homelessness
children
change
2
jobs
freedom
understand
reform
obama
leadership
established
taxpayer
opportunity
goal
capital
3
republican
transportation
income
rates
fund
america's
standards
policy
job
community
code
4
democrats
workers
americans
jobs
people
support
housing
social
rights
paying
family
country
5
century
security
care
strong
reach
raise
pay
service
retirement
programs
investments
21st
growth
6
trade
banks
principles
housing
prosperity
means
legislation
million
investment
regulations
based
7
access
laws
democratic
percent
infrastructure
global
education
workforce
street
creating
8
platform
grown
services
reduce
mortgage
intellectual
demand
consumers
creating
raise
left
interstate
households
ensuring
deserve
corporate
believes
9
government
current
administration
federal
u.s
country
increasing
created
agreements
fair
advance
10
american
economy
america
economic
tax
business
federal
people
public
innovation
term
beta