NLP, Log-Odds Notes

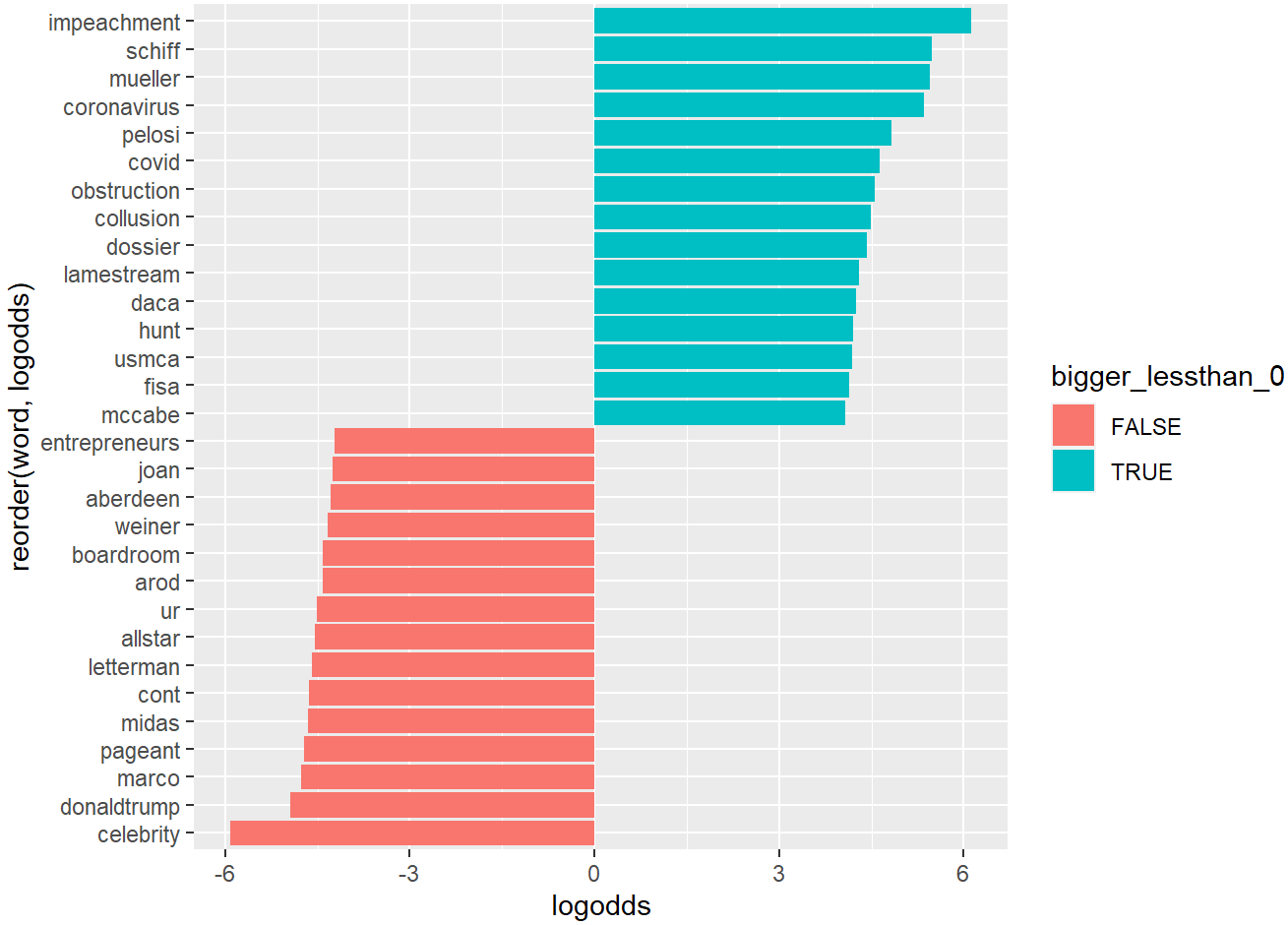
2023-11-27

```
require(tidyverse)
```

```
## Loading required package: tidyverse
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.2     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.4     ✓ tibble    3.2.1
## ✓ lubridate 1.9.2     ✓ tidyr     1.3.0
## ✓ purrr     1.0.1
## ── 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 becom
e errors
```

```
tweet_words <- read_rds('https://github.com/jbisbee1/DS1000_F2023/raw/main/Lectures/8_Clustering
_NLP/data/Trump_tweet_words.Rds')


# Step 1: Calculate counts
odds1 <- tweet_words %>%
  mutate(PostPresident = Tweeting.date >= as.Date('2016-11-04')) %>%
  group_by(word) %>%
  count(word,PostPresident) %>%
  filter(sum(n) >= 5) %>%
  spread(PostPresident,n,fill = 0) %>%
  ungroup() %>%
  mutate(totPre = sum(`FALSE`),
         totPost = sum(`TRUE`))


# Step 2: Calculating Probabilities
odds2 <- odds1 %>%
  mutate(probPre = (`FALSE` + 1)/ (totPre + 1),
         probPost = (`TRUE` + 1) / (totPost + 1))

# Step 3: Calculate Odds
odds3 <- odds2 %>%
  mutate(odds = probPost / probPre)

# Step 4: Log it!
odds4 <- odds3 %>%
  mutate(logodds = log(odds))

# Create pretty plot
odds4 %>%
  mutate(bigger_lessthan_0 = logodds > 0) %>%
  group_by(bigger_lessthan_0) %>%
  top_n(15,wt = abs(logodds)) %>%
  ggplot(aes(x = logodds,y = reorder(word,logodds),fill = bigger_lessthan_0)) +
  geom_bar(stat = 'identity')
```

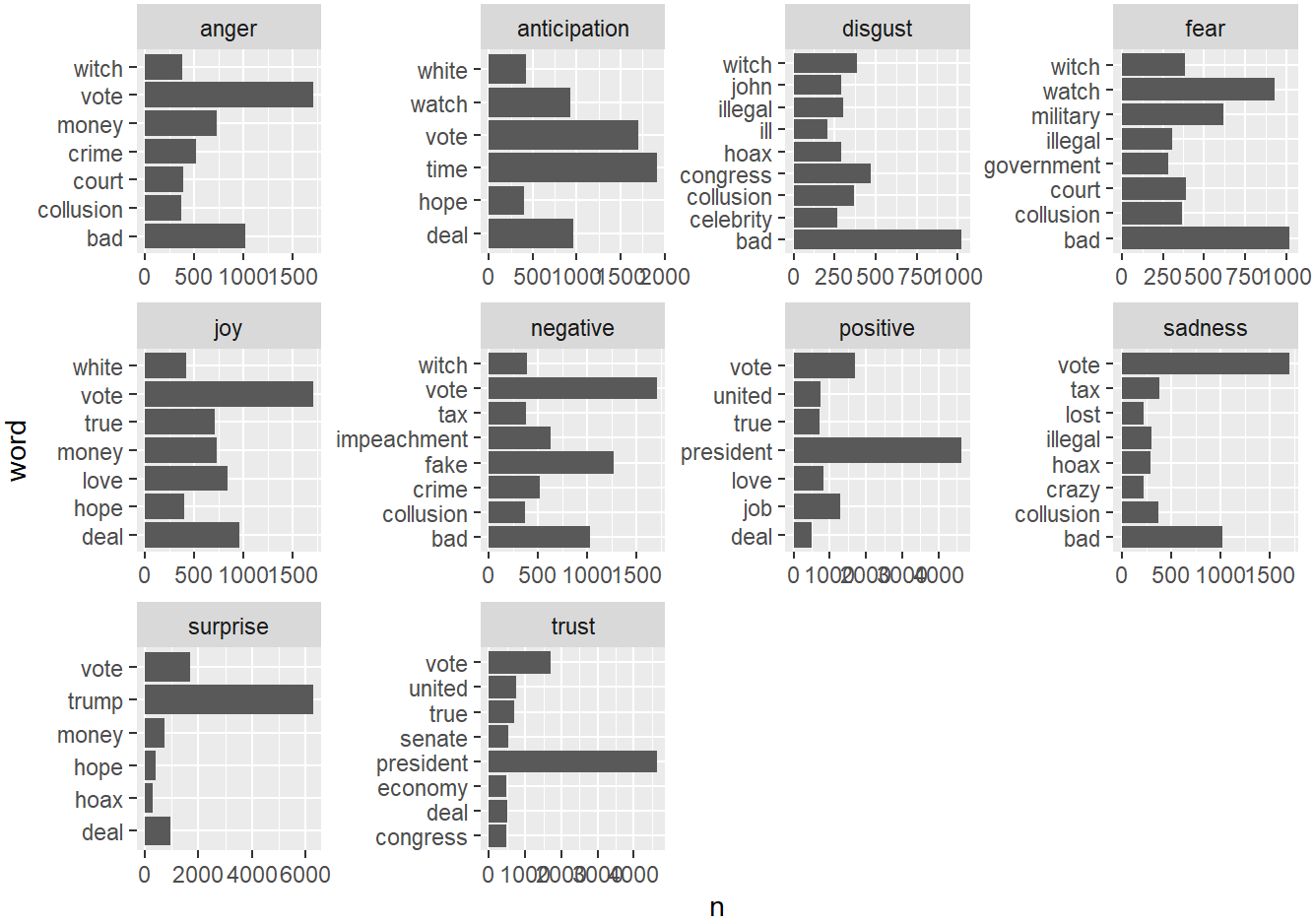


Sentiment

```
nrc <- read_rds('https://github.com/jbisbee1/DS1000_F2023/raw/main/Lectures/8_Clustering_NLP/data/nrc.Rds')

# Calculate proportion of words (again)
word_freq <- tweet_words %>%
  mutate(PostPresident = ifelse(Tweeting.date >= as.Date('2016-11-04'),
                                'Post','Pre')) %>%
  group_by(PostPresident) %>%
  count(word) %>%
  filter(sum(n) > 5) %>%
  mutate(prop = prop.table(n))

# Merge
word_freq_sentiment <- word_freq %>%
  inner_join(nrc,by = 'word')
```

```
## Warning in inner_join(., nrc, by = "word"): Detected an unexpected many-to-many relationship
between `x` and `y`.
## ℹ Row 8 of `x` matches multiple rows in `y`.
## ℹ Row 2 of `y` matches multiple rows in `x`.
## ℹ If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```
# Visualization of top words
word_freq_sentiment %>%
  group_by(sentiment) %>%
  top_n(10,wt = n) %>%
  ggplot(aes(x = n,y = word)) +
  geom_bar(stat = 'identity') +
  facet_wrap(~ sentiment,scales = 'free',nrow = 3)
```



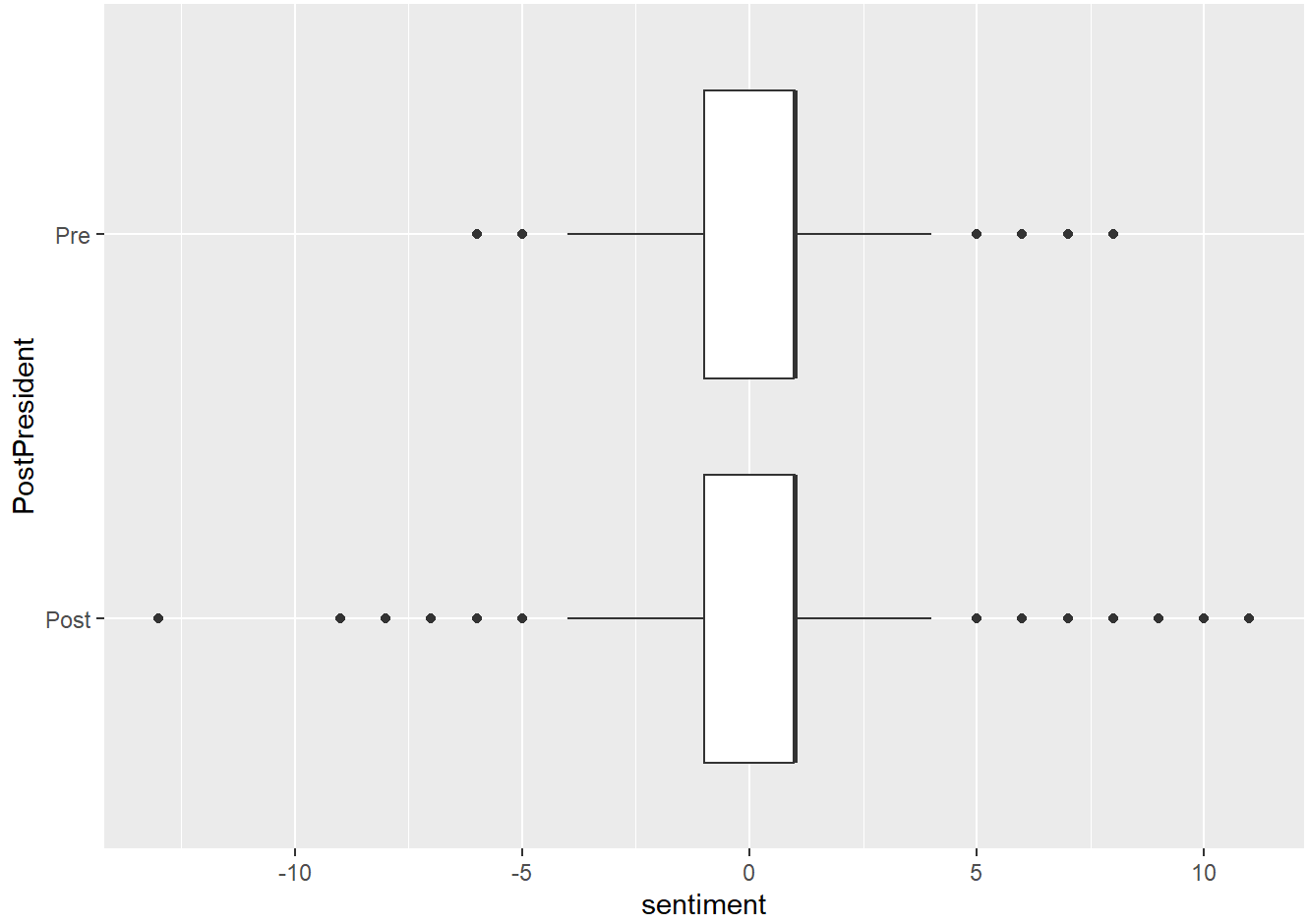

```
# Measuring sentiment as positive minus negative words
tweet_sentiment <- tweet_words %>%
  inner_join(nrc,by = 'word')
```

```
## Warning in inner_join(., nrc, by = "word"): Detected an unexpected many-to-many relationship
between `x` and `y`.
## ℹ Row 2 of `x` matches multiple rows in `y`.
## ℹ Row 12751 of `y` matches multiple rows in `x`.
## ℹ If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```
tweet_sentiment_summary <- tweet_sentiment %>%
  mutate(PostPresident = ifelse(Tweeting.date >= as.Date('2016-11-04'),'Post','Pre')) %>%
  group_by(PostPresident,sentiment) %>%
  count(document,sentiment) %>%
  arrange(document,sentiment) %>%
  filter(sentiment %in% c('positive','negative')) %>%
  pivot_wider(names_from = sentiment,
              values_from = n,
              values_fill = 0) %>%
  mutate(sentiment = positive - negative)

# Sentiment by Presidency
tweet_sentiment_summary %>%
  group_by(PostPresident) %>%
  mutate(ntweet = 1) %>%
  summarise(across(-document,sum))
```

```
## # A tibble: 2 × 5
##   PostPresident negative positive sentiment ntweet
##   <chr>            <int>    <int>     <int>  <dbl>
## 1 Post             25719    31550      5831  20419
## 2 Pre              14952    27316     12364  21347
```

```
# Visualization
tweet_sentiment_summary %>%
  ggplot(aes(x = sentiment,y = PostPresident)) +
  geom_boxplot()
```


Pre
Post
PostPresident
-10
-5
0
5
10
sentiment