Alex Jones - EDA

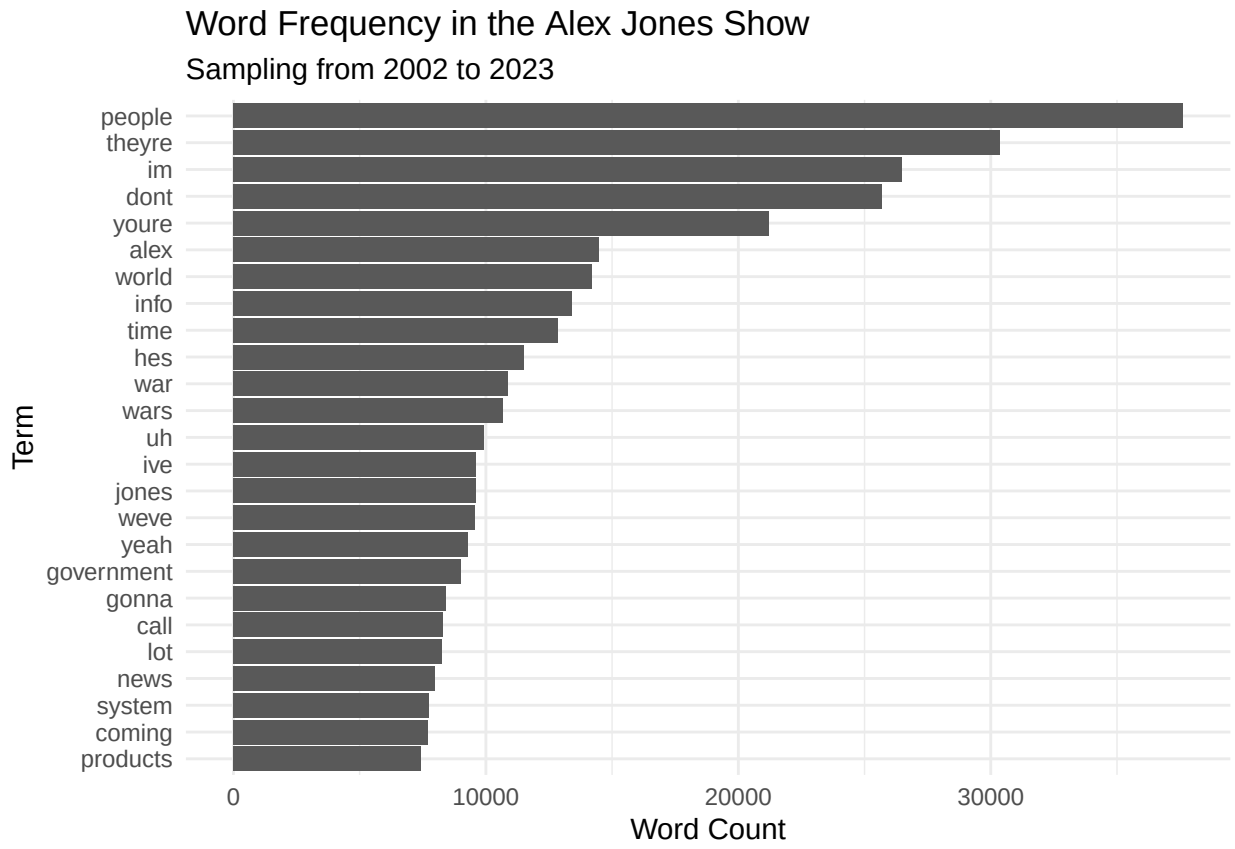
Karyn Brehmeyer

2023-06-11

Exploratory Data Analysis

```
## Joining, by = "word"
## `summarise()` has grouped output by 'word'. You can override using the
## `.groups` argument.
## Joining, by = "date"
## `summarise()` has grouped output by 'ep_year'. You can override using the
## `.groups` argument.
```

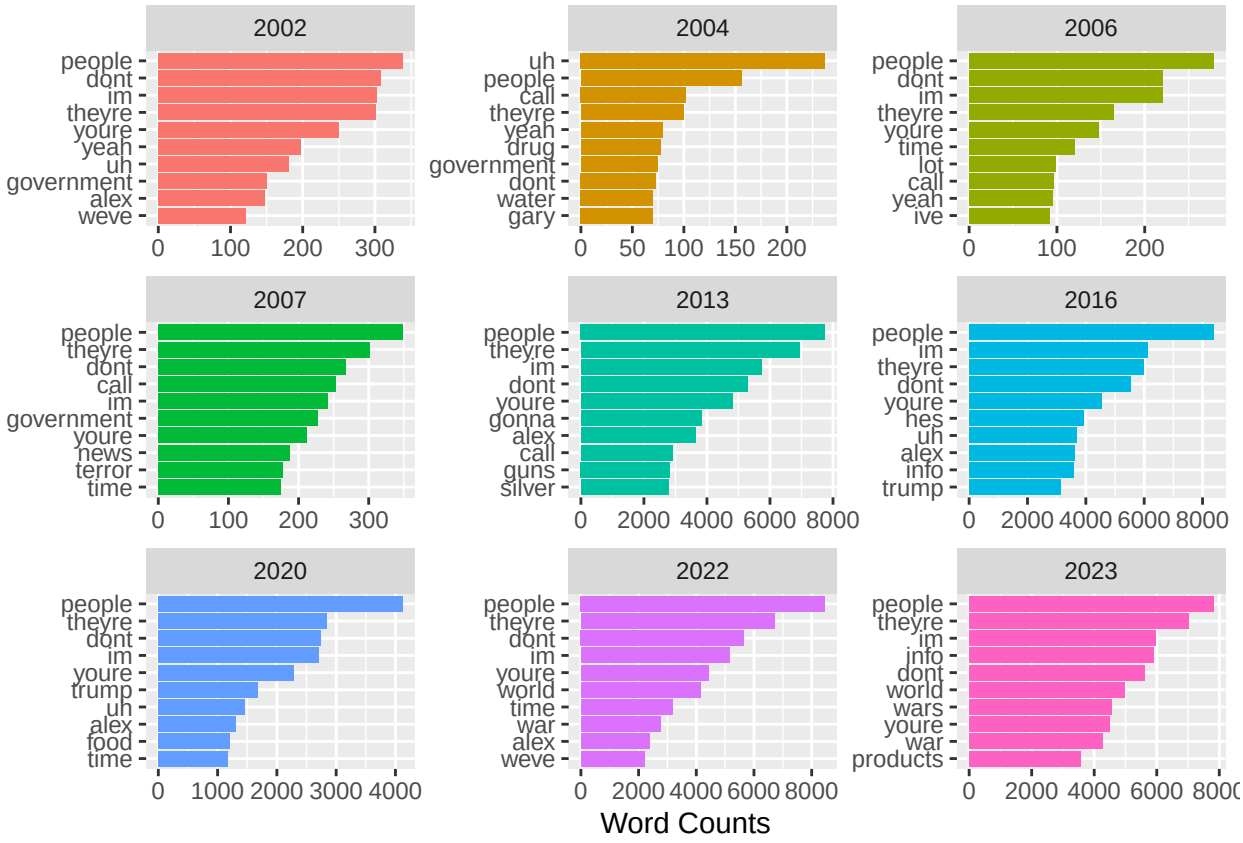
```
#top 25 all time
year_metrics %>%
  group_by(word) %>%
  summarise(word_count = sum(count)) %>%
  top_n(25, word_count) %>%
  mutate(work = reorder(word, word_count)) %>%
  ggplot(aes(work, word_count)) +
  geom_col() +
  xlab(NULL) +
  coord_flip() +
  labs(title = 'Word Frequency in the Alex Jones Show', subtitle = 'Sampling from 2002 to 2023', x = "T
  theme_minimal()
```



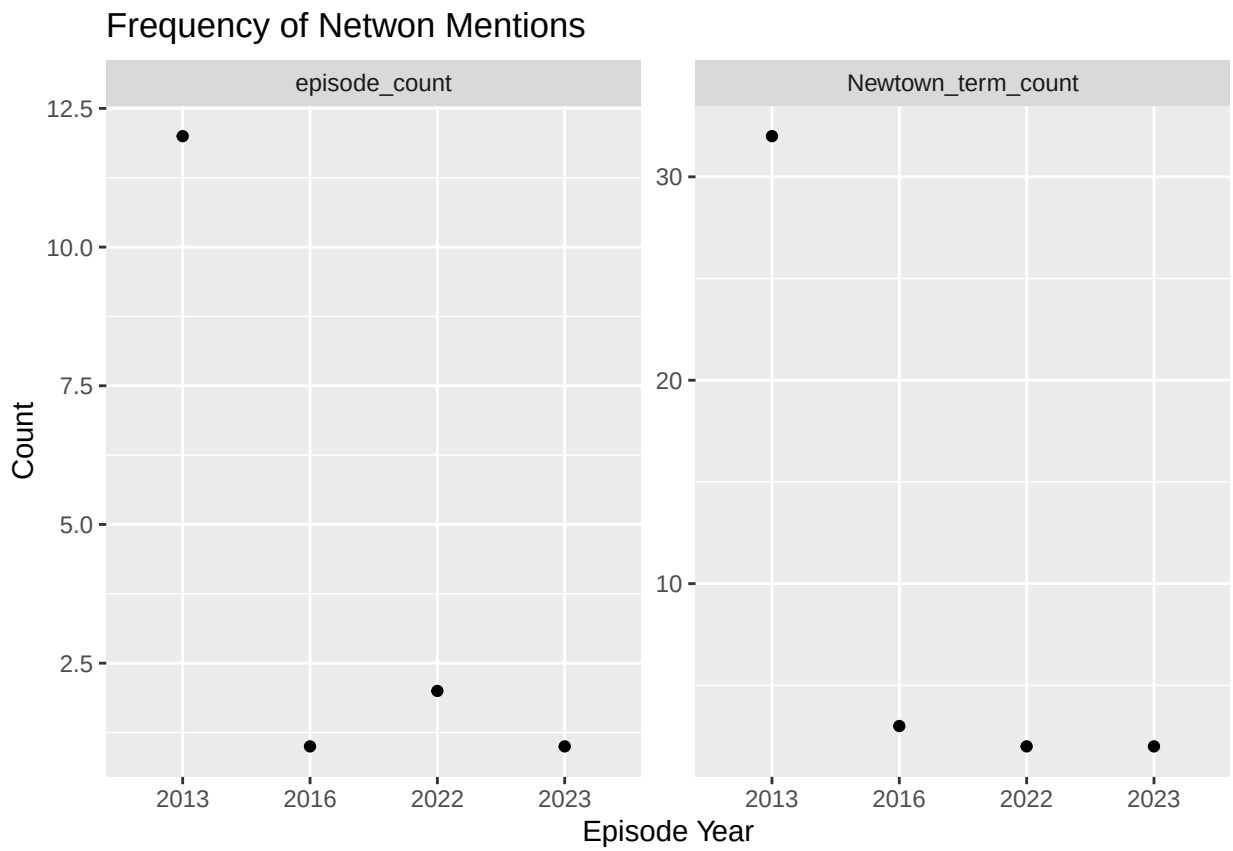

```
#word cloud
year_metrics %>%
  group_by(word) %>%
  summarise(word_count = sum(count)) %>%
  with(wordcloud(word, word_count, random.order = FALSE, max.words = 50))
```


```
#Top 20 words by count in each time period
year_metrics %>%
  group_by(ep_year, word) %>%
  summarise(word_count=sum(count)) %>%
  top_n(10, word_count) %>%
  ungroup %>%
  ggplot(aes(reorder_within(word, word_count, ep_year), word_count, fill = ep_year)) +
    geom_col(show.legend = FALSE) +
    scale_x_reordered() +
    facet_wrap(~ep_year, scales = "free") +
    coord_flip()+
    labs(x = NULL, y="Word Counts")
```

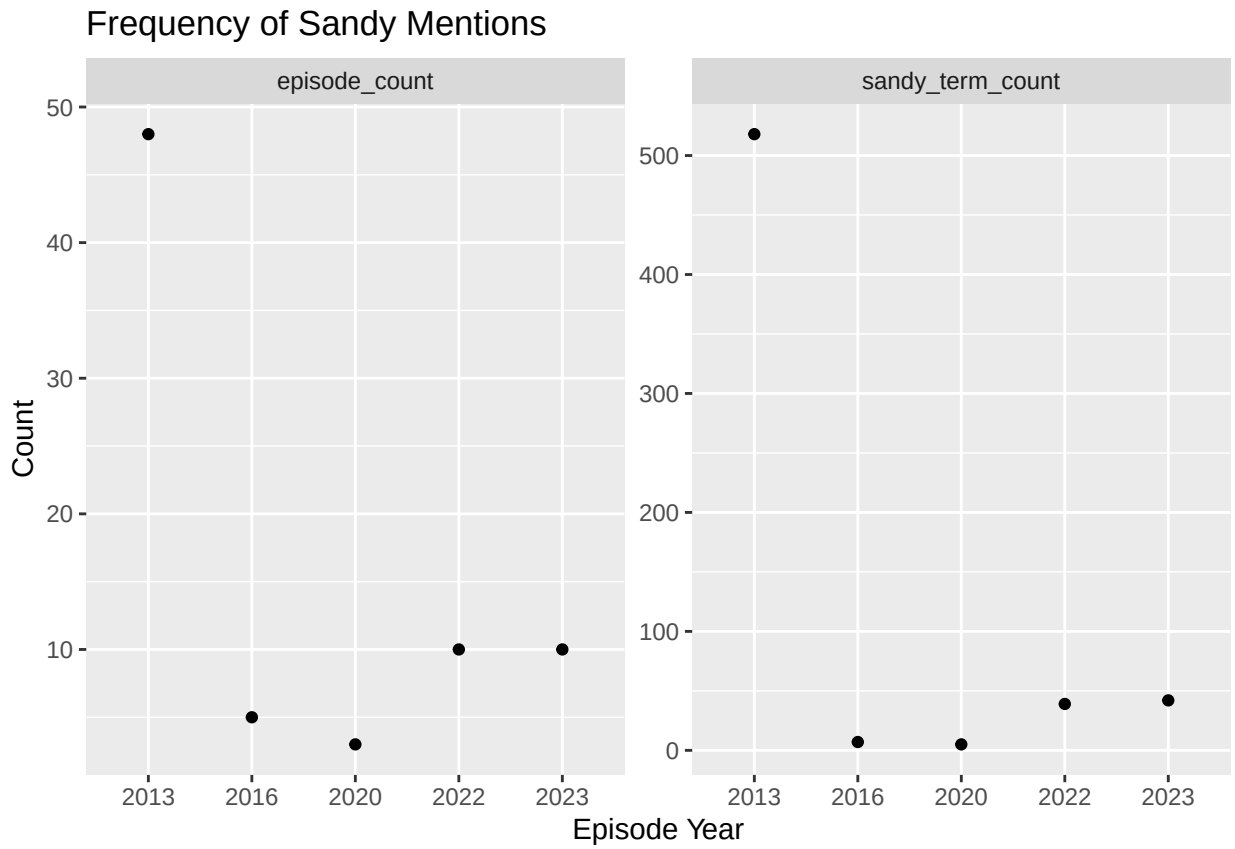
```
## `summarise()` has grouped output by 'ep_year'. You can override using the
## `.groups` argument.
```



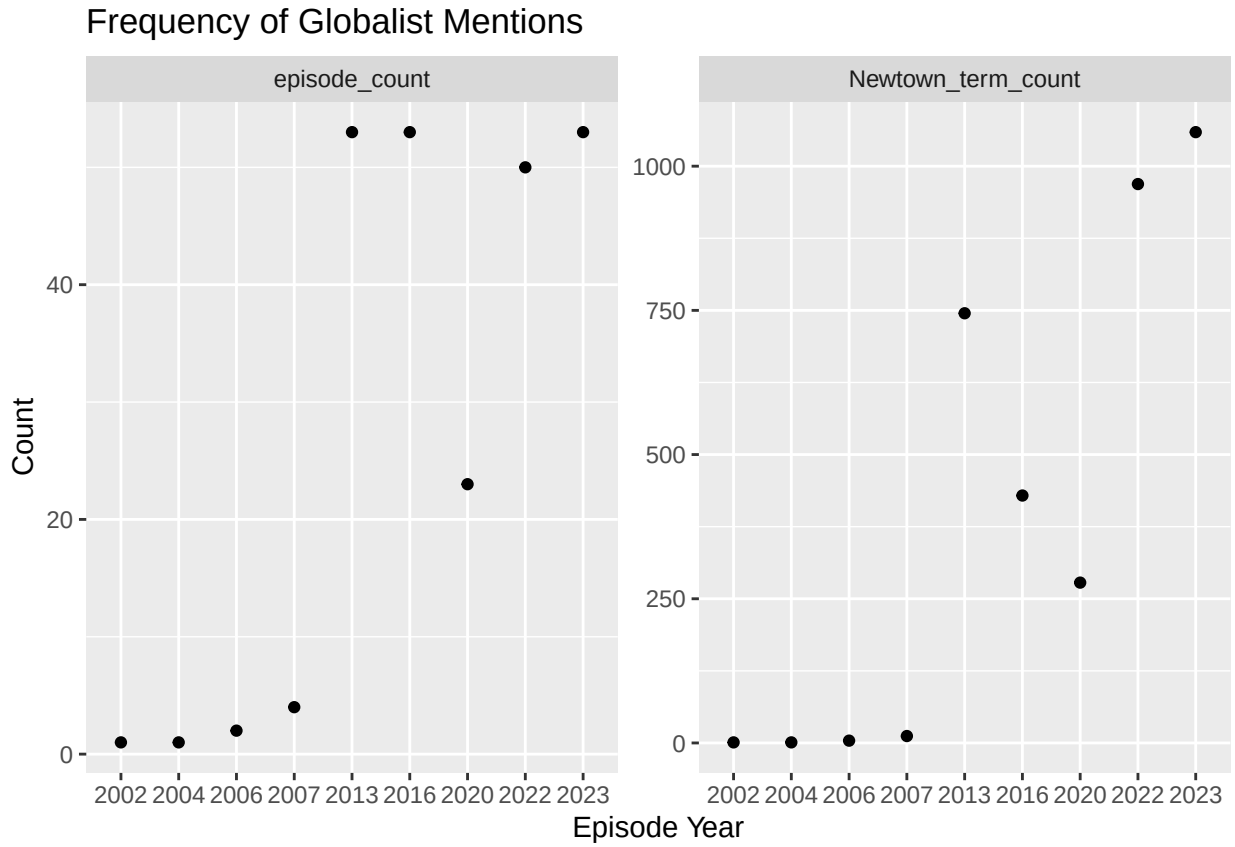

```
#plots are skewed due to uneven sampling across years
some_metrics %>%
  filter(word == "newtown") %>%
  group_by(ep_year) %>%
  summarize(
    episode_count = n_distinct(date),
    Newtown_term_count = sum(count)
  ) %>%
  gather(key = "type", value = "count", 2:3) %>%
  ggplot(aes(ep_year, count)) +
  geom_point() +
  facet_wrap(~ type, scales = "free") +
  labs(title = 'Frequency of Netwon Mentions', y = 'Count', x = 'Episode Year')
```




```
some_metrics %>%
  filter(word == 'sandy') %>%
  group_by(ep_year) %>%
  summarize(
    episode_count = n_distinct(date),
    sandy_term_count = sum(count)
  ) %>%
  gather(key = "type", value = "count", 2:3) %>%
  ggplot(aes(ep_year, count)) +
  geom_point() +
  facet_wrap(~ type, scales = "free") +
  labs(title = 'Frequency of Sandy Mentions', y = 'Count', x = 'Episode Year')
```




```
#plots are skewed due to uneven sampling across years
some_metrics %>%
  filter(word == "globalist") %>%
  group_by(ep_year) %>%
  summarize(
    episode_count = n_distinct(date),
    Newtown_term_count = sum(count)
  ) %>%
  gather(key = "type", value = "count", 2:3) %>%
  ggplot(aes(ep_year, count)) +
  geom_point() +
  facet_wrap(~ type, scales = "free") +
  labs(title = 'Frequency of Globalist Mentions', y = 'Count', x = 'Episode Year')
```




```
library(scales)
```

```
##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor
```

```
year_metrics %>%
    filter(ep_year == '2013' | ep_year == '2023') %>% #compare two years
    filter(count>500) %>%
    group_by(ep_year, word) %>%
    summarize(n = sum(count)) %>%
    mutate(proportion = n / sum(n)) %>%
    pivot_wider(id_cols = word, names_from = ep_year, values_from = proportion) %>%
    ggplot(aes(x = `2013`, y = `2023`, color = abs(`2013` - `2023`))) +
      geom_abline(color = "gray40", lty = 2) +
      geom_jitter(alpha = 0.1, size = 2.5, width = 0.3, height = 0.3) +
      geom_text(aes(label = word), check_overlap = TRUE, vjust = 1.5) +
```

```
      scale_x_log10(labels = percent_format()) +
      scale_y_log10(labels = percent_format()) +
      scale_color_gradient(limits = c(0, 0.001), low = "darkslategray4", high = "gray75") +
      theme(legend.position="none") +
      labs(y = '2023', x = '2013', title = 'Word Frequency Changes Between 2013 & 2023')
```

```
## `summarise()` has grouped output by 'ep_year'. You can override using the
## `.groups` argument.
```