Twitter reproducible methods

Dr .Mario Antonio Martinez

11/29/2021

Introduction to procedurals

The following code below shows procedures to make the initial data frame. It is used to inform the data, but it not run. In the grouped code below, we pick up the code from the storage device on the computer

```
library(magrittr)

a <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter_shiny\\twittter\\data\\tweetschool.rds")

o <- readxl::read_excel("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\latest data 11-28-21\\teaching online 12-3-21 - Copy.xlsx")

a1 <- setdiff(a$content, o$content) %>% data.frame() # n = 7701
o1 <- setdiff(o$content,a$content) %>% data.frame() # n = 4392




saveRDS(to1, "c:/users/mario/downloads/to1.rds")
saveRDS(ts1, "c:/users/mario/downloads/ts1.rds")
```

Here, we pick up the code with the data from the storage device on the computer. Quants methods wilcox tests

```
ts <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\Latest-data-12-4-21\\ts1.rds") # n = 7701
to <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\Latest-data-12-4-21\\to1.rds") # n = 4392

library(WRS2)

## Warning: package 'WRS2' was built under R version 4.1.3

library(plotrix)

ts1 <- syuzhet::get_nrc_sentiment(ts)

data <- data.frame(cbind(ts1,ts))
```

```r
to1 <- syuzhet::get_nrc_sentiment(to)

data1 <- data.frame(cbind(to1,to))

#data$kind <- rep("ts", 2000)
#data1$kind <- rep("to", 2000)

data$kind <- rep("ts", 7701)
data1$kind <- rep("to", 4392)


options(scipen=999)

#

dataTeachschoolpos <-data[,c("kind", "positive")]

# shapiro.test(dataTeachschoolpos$positive)

mean(dataTeachschoolpos$positive)

## [1] 2.756655

plotrix::std.error(dataTeachschoolpos$positive)

## [1] 0.01567726

dataTeachonlinepos <- data1[,c("kind", "positive")]

# shapiro.test(dataTeachonlinepos$positive)

mean(dataTeachonlinepos$positive)

## [1] 2.889117

plotrix::std.error(dataTeachonlinepos$positive)

## [1] 0.02142073

now_data_pos <- data.frame(rbind(dataTeachonlinepos,dataTeachschoolpos))



(rpositive <- wilcox.test(now_data_pos$positive~now_data_pos$kind))

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_pos$positive by now_data_pos$kind
```

```
## W = 17792915, p-value = 0.0000009277
## alternative hypothesis: true location shift is not equal to 0

(ypositive <- WRS2::yuenbt(now_data_pos$positive~now_data_pos$kind))

## Call:
## WRS2::yuenbt(formula = now_data_pos$positive ~ now_data_pos$kind)
##
## Test statistic: 4.949 (df = NA), p-value = 0
##
## Trimmed mean difference:  0.13222
## 95 percent confidence interval:
## 0.0763     0.1882

(d1 <- effectsize::cohens_d(now_data_pos$positive~now_data_pos$kind,  pooled_
sd = FALSE))

## Registered S3 method overwritten by 'parameters':
##   method                          from
##   format.parameters_distribution datawizard

## Cohen's d |       95% CI
## ------------------------
## 0.09      | [0.06, 0.13]
##
## - Estimated using un-pooled SD.

effectsize::d_to_r(d1)

##     Cohens_d        CI     CI_low    CI_high
## 1 0.04732731 0.4290568 0.02874464 0.06585832

#

dataTeachschoolAnger <- data[, c("kind", "anger")]

# shapiro.test(dataTeachschoolAnger$anger)

mean(dataTeachschoolAnger$anger)

## [1] 0.3898195

plotrix::std.error(dataTeachschoolAnger$anger)

## [1] 0.007596367

dataTeachonlineAnger <- data1[,c("kind", "anger")]

# shapiro.test(dataTeachonlineAnger$anger)


mean(dataTeachonlineAnger$anger)
```

```
## [1] 0.2556922

plotrix::std.error(dataTeachonlineAnger$anger)

## [1] 0.008095361

now_data_anger <- data.frame(rbind(dataTeachschoolAnger,dataTeachonlineAnger)
)

(ranger <- wilcox.test(now_data_anger$anger~now_data_anger$kind))

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_anger$anger by now_data_anger$kind
## W = 15287565, p-value < 0.00000000000000022
## alternative hypothesis: true location shift is not equal to 0

(ypositive <- WRS2::yuenbt(now_data_anger$anger~now_data_anger$kind))

## Call:
## WRS2::yuenbt(formula = now_data_anger$anger ~ now_data_anger$kind)
##
## Test statistic: -11.3755 (df = NA), p-value = 0
##
## Trimmed mean difference:  -0.15382
## 95 percent confidence interval:
## -0.1814     -0.1262

(d2 <- effectsize::cohens_d(now_data_anger$anger~now_data_anger$kind,  pooled
_sd = FALSE))

## Cohen's d |         95% CI
## -------------------------
## -0.22     | [-0.26, -0.19]
##
## - Estimated using un-pooled SD.

effectsize::d_to_r(d2)

##    Cohens_d        CI     CI_low     CI_high
## 1 -0.110162 0.4290568 -0.1278178 -0.09239632

#

dataTeachschoolAnticipation <- data[, c("kind", "anticipation")]

# shapiro.test(dataTeachschoolAnticipation$anticipation)

mean(dataTeachschoolAnticipation$anticipation)

## [1] 0.7852227
```

```
plotrix::std.error(dataTeachschoolAnticipation$anticipation)

## [1] 0.0102917

dataTeachonlineAnticipation <- data1[,c("kind", "anticipation")]

# shapiro.test(dataTeachonlineAnticipation$anticipation)

mean(dataTeachonlineAnticipation$anticipation)

## [1] 0.8422131

plotrix::std.error(dataTeachonlineAnticipation$anticipation)

## [1] 0.01403281

now_data_anticipation <- data.frame(rbind(dataTeachschoolAnticipation,dataTea
chonlineAnticipation))

(ranticipation <- wilcox.test(now_data_anticipation$anticipation~now_data_ant
icipation$kind))

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_anticipation$anticipation by now_data_anticipation$kind
## W = 17459771, p-value = 0.001367
## alternative hypothesis: true location shift is not equal to 0

(yanticipation <- WRS2::yuenbt(now_data_anticipation$anticipation~now_data_an
ticipation$kind))

## Call:
## WRS2::yuenbt(formula = now_data_anticipation$anticipation ~ now_data_antic
ipation$kind)
##
## Test statistic: 2.8434 (df = NA), p-value = 0.00835
##
## Trimmed mean difference:  0.062
## 95 percent confidence interval:
## 0.0167     0.1073

(d3 <- effectsize::cohens_d(now_data_anticipation$anticipation~now_data_antic
ipation$kind,  pooled_sd = FALSE))

## Cohen's d |       95% CI
## ------------------------
## 0.06      | [0.02, 0.10]
##
## - Estimated using un-pooled SD.

effectsize::d_to_r(d3)
```

```
##     Cohens_d        CI     CI_low    CI_high
## 1 0.03107068 0.4290568 0.01247398 0.04963343

#


dataTeachschoolDisgust <- data[, c("kind", "disgust")]

# shapiro.test(dataTeachschoolDisgust$disgust)


mean(dataTeachschoolDisgust$disgust)

## [1] 0.3068433

plotrix::std.error(dataTeachschoolDisgust$disgust)

## [1] 0.006705102

dataTeachonlineDisgust <- data1[,c("kind", "disgust")]

# shapiro.test(dataTeachonlineDisgust$disgust)


mean(dataTeachonlineDisgust$disgust)

## [1] 0.1773679

plotrix::std.error(dataTeachonlineDisgust$disgust)

## [1] 0.006673303

now_data_disgust <- data.frame(rbind(dataTeachschoolDisgust,dataTeachonlineDi
sgust))

(rdisgust <- wilcox.test(now_data_disgust$disgust~now_data_disgust$kind))

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_disgust$disgust by now_data_disgust$kind
## W = 15259224, p-value < 0.00000000000000022
## alternative hypothesis: true location shift is not equal to 0

(ydisgust <- WRS2::yuenbt(now_data_disgust$disgust~now_data_disgust$kind))

## Call:
## WRS2::yuenbt(formula = now_data_disgust$disgust ~ now_data_disgust$kind)
##
## Test statistic: -10.1553 (df = NA), p-value = 0
##
## Trimmed mean difference:  -0.08353
```

```
## 95 percent confidence interval:
## -0.0995     -0.0675

(d4 <- effectsize::cohens_d(now_data_disgust$disgust~now_data_disgust$kind,
pooled_sd = FALSE))

## Cohen's d |         95% CI
## -------------------------
## -0.25     | [-0.28, -0.21]
##
## - Estimated using un-pooled SD.

effectsize::d_to_r(d4)

##    Cohens_d        CI     CI_low    CI_high
## 1 -0.123428 0.4290568 -0.1408438 -0.1058919

#

dataTeachschoolFear <- data[, c("kind", "fear")]

# shapiro.test(dataTeachschoolFear$fear)


mean(dataTeachschoolFear$fear)

## [1] 0.7058824

plotrix::std.error(dataTeachschoolFear$fear)

## [1] 0.01013822

dataTeachonlineFear <- data1[,c("kind", "fear")]

# shapiro.test(dataTeachonlineFear$fear)


mean(dataTeachonlineFear$fear)

## [1] 0.4913479

plotrix::std.error(dataTeachonlineFear$fear)

## [1] 0.01122434

now_data_fear <- data.frame(rbind(dataTeachschoolFear,dataTeachonlineFear))

(rdisgust <- wilcox.test(now_data_fear$fear~now_data_fear$kind))

##
##  Wilcoxon rank sum test with continuity correction
##
```

```
## data:  now_data_fear$fear by now_data_fear$kind
## W = 14689235, p-value < 0.00000000000000022
## alternative hypothesis: true location shift is not equal to 0
```

```
(ydisgust <- WRS2::yuenbt(now_data_fear$fear~now_data_fear$kind))
```

```
## Call:
## WRS2::yuenbt(formula = now_data_fear$fear ~ now_data_fear$kind)
##
## Test statistic: -12.7037 (df = NA), p-value = 0
##
## Trimmed mean difference:  -0.19594
## 95 percent confidence interval:
## -0.2252     -0.1666
```

```
(d5 <- effectsize::cohens_d(now_data_fear$fear~now_data_fear$kind,  pooled_sd
= FALSE))
```

```
## Cohen's d |         95% CI
## -------------------------
## -0.26     | [-0.30, -0.23]
##
## - Estimated using un-pooled SD.
```

```
effectsize::d_to_r(d5)
```

```
##     Cohens_d        CI     CI_low    CI_high
## 1 -0.1297055 0.4290568 -0.1473435 -0.1119375
```

```
#



dataTeachschoolJoy <- data[, c("kind", "joy")]

# shapiro.test(dataTeachschoolJoy$joy)


mean(dataTeachschoolJoy$joy)
```

```
## [1] 1.527334
```

```
plotrix::std.error(dataTeachschoolJoy$joy)
```

```
## [1] 0.008828204
```

```
dataTeachonlineJoy <- data1[,c("kind", "joy")]

# shapiro.test(dataTeachonlineJoy$joy)
```

```r
mean(dataTeachonlineJoy$joy)
```

```
## [1] 1.51867
```

```r
plotrix::std.error(dataTeachonlineJoy$joy)
```

```
## [1] 0.01146838
```

```r
now_data_joy <- data.frame(rbind(dataTeachschoolJoy,dataTeachonlineJoy))

(rjoy <- wilcox.test(now_data_joy$joy~now_data_joy$kind))
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_joy$joy by now_data_joy$kind
## W = 16722630, p-value = 0.2418
## alternative hypothesis: true location shift is not equal to 0
```

```r
(yjoy <- WRS2::yuenbt(now_data_joy$joy~now_data_joy$kind))
```

```
## Call:
## WRS2::yuenbt(formula = now_data_joy$joy ~ now_data_joy$kind)
##
## Test statistic: -2.3335 (df = NA), p-value = 0.01836
##
## Trimmed mean difference:  -0.03588
## 95 percent confidence interval:
## -0.0631     -0.0087
```

```r
(d6 <- effectsize::cohens_d(now_data_joy$joy~now_data_joy$kind,  pooled_sd =
FALSE))
```

```
## Cohen's d |        95% CI
## -------------------------
## -0.01     | [-0.05, 0.03]
##
## - Estimated using un-pooled SD.
```

```r
effectsize::d_to_r(d6)
```

```
##      Cohens_d        CI      CI_low    CI_high
## 1 -0.00564472 0.4290568 -0.02411945 0.01283609
```

```r
#

dataTeachschoolSadness <- data[, c("kind", "sadness")]

# shapiro.test(dataTeachschoolSadness$sadness)
```

```r
mean(dataTeachschoolSadness$sadness)
```

```
## [1] 0.5810934
```

```r
plotrix::std.error(dataTeachschoolSadness$sadness)
```

```
## [1] 0.009232616
```

```r
dataTeachonlineSadness <- data1[,c("kind", "sadness")]

# shapiro.test(dataTeachonlineSadness$sadness)


mean(dataTeachonlineSadness$sadness)
```

```
## [1] 0.4385246
```

```r
plotrix::std.error(dataTeachonlineSadness$sadness)
```

```
## [1] 0.01060124
```

```r
now_data_sadness <- data.frame(rbind(dataTeachschoolSadness,dataTeachonlineSa
dness))

(rsadness <- wilcox.test(now_data_sadness$sadness~now_data_sadness$kind))
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_sadness$sadness by now_data_sadness$kind
## W = 15384127, p-value < 0.00000000000000022
## alternative hypothesis: true location shift is not equal to 0
```

```r
(ysadness <- WRS2::yuenbt(now_data_sadness$sadness~now_data_sadness$kind))
```

```
## Call:
## WRS2::yuenbt(formula = now_data_sadness$sadness ~ now_data_sadness$kind)
##
## Test statistic: -8.9918 (df = NA), p-value = 0
##
## Trimmed mean difference:  -0.13636
## 95 percent confidence interval:
## -0.1649     -0.1079
```

```r
(d7 <- effectsize::cohens_d(now_data_sadness$sadness~now_data_sadness$kind,
pooled_sd = FALSE))
```

```
## Cohen's d |         95% CI
## -------------------------
## -0.19     | [-0.22, -0.15]
```

```
##
## - Estimated using un-pooled SD.

effectsize::d_to_r(d7)

##      Cohens_d        CI     CI_low     CI_high
## 1 -0.09359266 0.4290568 -0.1115164 -0.07557295

#

dataTeachschoolSurprise <- data[, c("kind", "surprise")]

# shapiro.test(dataTeachschoolSurprise$surprise)

mean(dataTeachschoolSurprise$surprise)

## [1] 1.26555

plotrix::std.error(dataTeachschoolSurprise$surprise)

## [1] 0.006321945

dataTeachonlineSurprise <- data1[,c("kind", "surprise")]

# shapiro.test(dataTeachonlineSurprise$surprise)


mean(dataTeachonlineSurprise$surprise)

## [1] 1.245446

plotrix::std.error(dataTeachonlineSurprise$surprise)

## [1] 0.007767

now_data_surprise <- data.frame(rbind(dataTeachschoolSurprise,dataTeachonline
Surprise))

(rsurprise <- wilcox.test(now_data_surprise$surprise~now_data_surprise$kind))

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_surprise$surprise by now_data_surprise$kind
## W = 16618384, p-value = 0.03103
## alternative hypothesis: true location shift is not equal to 0

(ysurprise <- WRS2::yuenbt(now_data_surprise$surprise~now_data_surprise$kind)
)

## Call:
## WRS2::yuenbt(formula = now_data_surprise$surprise ~ now_data_surprise$kind
```

```
)
##
## Test statistic: -3.2359 (df = NA), p-value = 0.00835
##
## Trimmed mean difference:  -0.04217
## 95 percent confidence interval:
## -0.0682     -0.0162

(d8 <- effectsize::cohens_d(now_data_surprise$surprise~now_data_surprise$kind
,  pooled_sd = FALSE))

## Cohen's d |         95% CI
## -------------------------
## -0.04     | [-0.07,  0.00]
##
## - Estimated using un-pooled SD.

effectsize::d_to_r(d8)

##      Cohens_d        CI      CI_low       CI_high
## 1 -0.01878042 0.4290568 -0.03709923 -0.0004417097

#

dataTeachschoolNegative <- data[, c("kind", "negative")]

# shapiro.test(dataTeachschoolNegative$negative)


mean(dataTeachschoolNegative$negative)

## [1] 1.181665

plotrix::std.error(dataTeachschoolNegative$negative)

## [1] 0.01367244

dataTeachonlineNegative <- data1[,c("kind", "negative")]

# shapiro.test(dataTeachonlineNegative$negative)


mean(dataTeachonlineNegative$negative)

## [1] 0.8239982

plotrix::std.error(dataTeachonlineNegative$negative)

## [1] 0.015294

now_data_negative <- data.frame(rbind(dataTeachschoolNegative,dataTeachonline
Negative))
```

```r
(rnegative <- wilcox.test(now_data_negative$negative~now_data_negative$kind))
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_negative$negative by now_data_negative$kind
## W = 13995819, p-value < 0.00000000000000022
## alternative hypothesis: true location shift is not equal to 0
```

```r
(ynegative <- WRS2::yuenbt(now_data_negative$negative~now_data_negative$kind)
)
```

```
## Call:
## WRS2::yuenbt(formula = now_data_negative$negative ~ now_data_negative$kind
)
##
## Test statistic: -17.1123 (df = NA), p-value = 0
##
## Trimmed mean difference:  -0.42863
## 95 percent confidence interval:
## -0.4758     -0.3815
```

```r
(d9 <- effectsize::cohens_d(now_data_negative$negative~now_data_negative$kind
,  pooled_sd = FALSE))
```

```
## Cohen's d |         95% CI
## -------------------------
## -0.32     | [-0.36, -0.29]
##
## - Estimated using un-pooled SD.
```

```r
effectsize::d_to_r(d9)
```

```
##     Cohens_d        CI     CI_low   CI_high
## 1 -0.1589748 0.4290568 -0.1764397 -0.141352
```

```r
#

dataTeachschoolTrust <- data[, c("kind", "trust")]

# shapiro.test(dataTeachschoolTrust$trust)


mean(dataTeachschoolTrust$trust)
```

```
## [1] 2.79665
```

```r
plotrix::std.error(dataTeachschoolTrust$trust)
```

```
## [1] 0.01332051
```

```r
dataTeachonlineTrust <- data1[,c("kind", "trust")]

# shapiro.test(dataTeachonlineTrust$trust)


mean(dataTeachonlineTrust$trust)

## [1] 2.133197

plotrix::std.error(dataTeachonlineTrust$trust)

## [1] 0.01598062

now_data_trust <- data.frame(rbind(dataTeachschoolTrust,dataTeachonlineTrust)
)

(rtrust <- wilcox.test(now_data_trust$trust~now_data_trust$kind))

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  now_data_trust$trust by now_data_trust$kind
## W = 11273781, p-value < 0.00000000000000022
## alternative hypothesis: true location shift is not equal to 0

(ytrust <- WRS2::yuenbt(now_data_trust$trust~now_data_trust$kind))

## Call:
## WRS2::yuenbt(formula = now_data_trust$trust ~ now_data_trust$kind)
##
## Test statistic: -27.2579 (df = NA), p-value = 0
##
## Trimmed mean difference:  -0.68186
## 95 percent confidence interval:
## -0.7309     -0.6328

(d10 <- effectsize::cohens_d(now_data_trust$trust~now_data_trust$kind,  poole
d_sd = FALSE))

## Cohen's d |         95% CI
## -------------------------
## -0.59     | [-0.63, -0.56]
##
## - Estimated using un-pooled SD.

effectsize::d_to_r(d10)

##     Cohens_d        CI     CI_low    CI_high
## 1 -0.2850748 0.4290568 -0.3014455 -0.2684378
```

This part harnesses the qualitative part of the data:

get the visual wheel with term frequencies at 10 minimum.

```
library(magrittr)
library(quanteda)
ts <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\
Latest-data-12-4-21\\ts.rds")
to <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\
Latest-data-12-4-21\\to.rds")
ts1 <- syuzhet::get_nrc_sentiment(ts)

data <- data.frame(cbind(ts1,ts))

to1 <- syuzhet::get_nrc_sentiment(to)


data1 <- data.frame(cbind(to1,to))
data$kind <- rep("ts", 2000)
data1$kind <- rep("to", 2000)

names(data1)[11] <- "content"
names(data)[11] <- "content"

bigdata <- data.frame(rbind(data,data1))

bigdata$content <- gsub("@[A-Za-z0-9]+", "", bigdata$content)


a <- quanteda::corpus(bigdata, text_field = "content")

library(quanteda.textplots)
this <- quanteda::corpus_subset(a,
                kind %in% c("to", "ts")) %>%
  quanteda::tokens(remove_punct = TRUE) %>%
  quanteda::tokens_remove(stopwords("english")) %>%
  quanteda::dfm() %>%
  quanteda::dfm_group(groups = kind) %>%
  quanteda::dfm_trim(min_termfreq = 10, verbose = FALSE) %>%
  quanteda.textplots::textplot_wordcloud(comparison = TRUE,
color = c("blue", "purple"))
```

```
library(magrittr)
ts <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\
Latest-data-12-4-21\\ts.rds")
to <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\
Latest-data-12-4-21\\to.rds")

ts1 <- syuzhet::get_nrc_sentiment(ts)
to1 <- syuzhet::get_nrc_sentiment(to)
```

```
data_to <- data.frame(cbind(to1,to))
data_ts <- data.frame(cbind(ts1,ts))

# Read in data once it has been saved and stored
to <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\
Latest-data-12-4-21\\qualitative tables\\data_to.rds")
ts <- readRDS("C:\\Users\\mario\\OneDrive\\Desktop\\twitter -11-25-21\\data\\
Latest-data-12-4-21\\qualitative tables\\data_ts.rds")

library(magrittr)

tsc <- data.frame(ts) %>% quanteda::corpus(., text_field = "Content", docid_f
ield = "Num")

toc <- data.frame(to) %>% quanteda::corpus(., text_field = "Content", docid_f
ield = "Num")

# save the dfm matrix to the text by copy pasting

tstoks1 <- quanteda::tokens(tsc, remove_numbers = TRUE, remove_punct = TRUE)
c1 <- quanteda::dfm(tstoks1, remove = stopwords::stopwords("english"))

totoks1 <- quanteda::tokens(toc, remove_numbers = TRUE, remove_punct = TRUE)
c2 <- quanteda::dfm(totoks1, remove = stopwords::stopwords("english"))




result <- ldatuning::FindTopicsNumber(
  c1,
  topics = seq(from = 2, to = 200, by = 5),
  metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010", "Deveaud2014"),
  method = "Gibbs",
  control = list(seed = 77),
  mc.cores = 2L,
  verbose = TRUE
)


#=== NOTE adding topic modeling

# Note that for each of the topics you must change the K and first argument o
f "stm" function


ldatuning::FindTopicsNumber_plot(result)

this <- topicmodels::LDA(c1,k =22) # use the c1 and topic number for best res
ults
```

```
that <- topicmodels::LDA(c2, k = 57)
(ap_topics <- tidytext::tidy(this, matrix = "beta"))

(ap_topics <- tidytext::tidy(that, matrix = "beta"))



library(ggplot2)
library(dplyr)
library(tidytext)

(ap_top_terms <- ap_topics %>%
    group_by(topic) %>%
    top_n(10, beta) %>%
    ungroup() %>%
    arrange(topic, -beta))

(ap_top_terms %>%
  mutate(term = tidytext::reorder_within(term, beta, topic)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip() +
  tidytext::scale_x_reordered())
```