Replication_File

Alex Newhouse

2024-06-22

```
library(tidyverse)
posts <- read_csv("posts.csv")
comments <- read_csv("comments.csv")
metadata <- read_csv("metadata.csv")
```

Figure 1a

Calculate monthly number of posts using the daily date variable in the posts.csv dataset, grouping data by month and filtering the data to posts after 01-01-2013. Plot monthly posts over time as a line graph.

```
posts %>%
  filter(date >= "2013-01-01" & date < "2018-09-01") %>%
  mutate(month_ = floor_date(date, "months")) %>%
  count(month_) %>%
  ggplot(aes(x = month_, y = n)) +
  geom_line() +
  labs(x = "Date",
       y = "Monthly Number of Posts",
       title = "Posts")
```

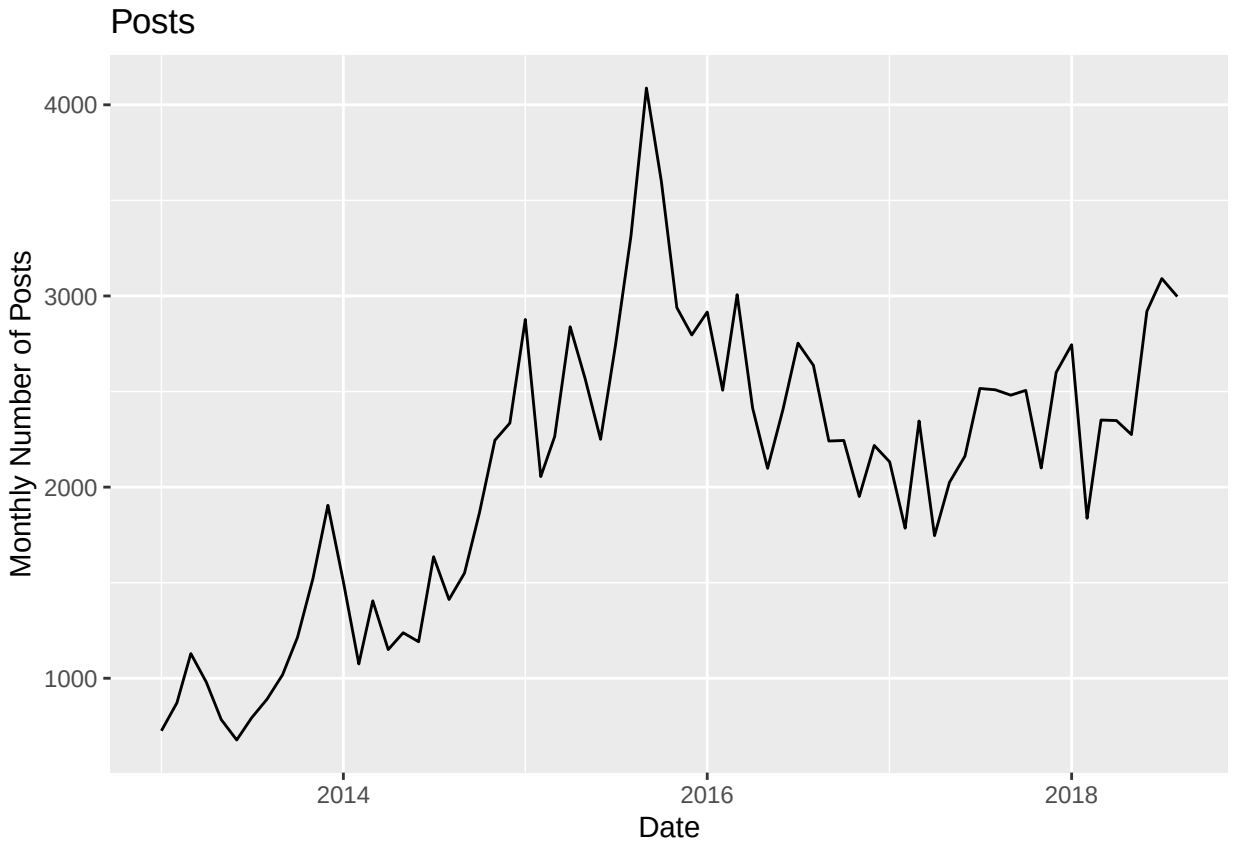


Figure 1b

Calculate monthly number of comments using the daily date variable in the comments.csv dataset, grouping the data by month and filtering the data to posts after 01-01-2013. Plot monthly comments over time as a line graph.

```
comments %>%
  filter(date >= "2013-01-01" & date < "2018-09-01") %>%
  mutate(month_ = floor_date(date, "months")) %>%
  count(month_) %>%
  ggplot(aes(x = month_, y = n)) +
  geom_line() +
  labs(x = "Date",
       y = "Monthly Number of Comments",
       title = "Comments")
```

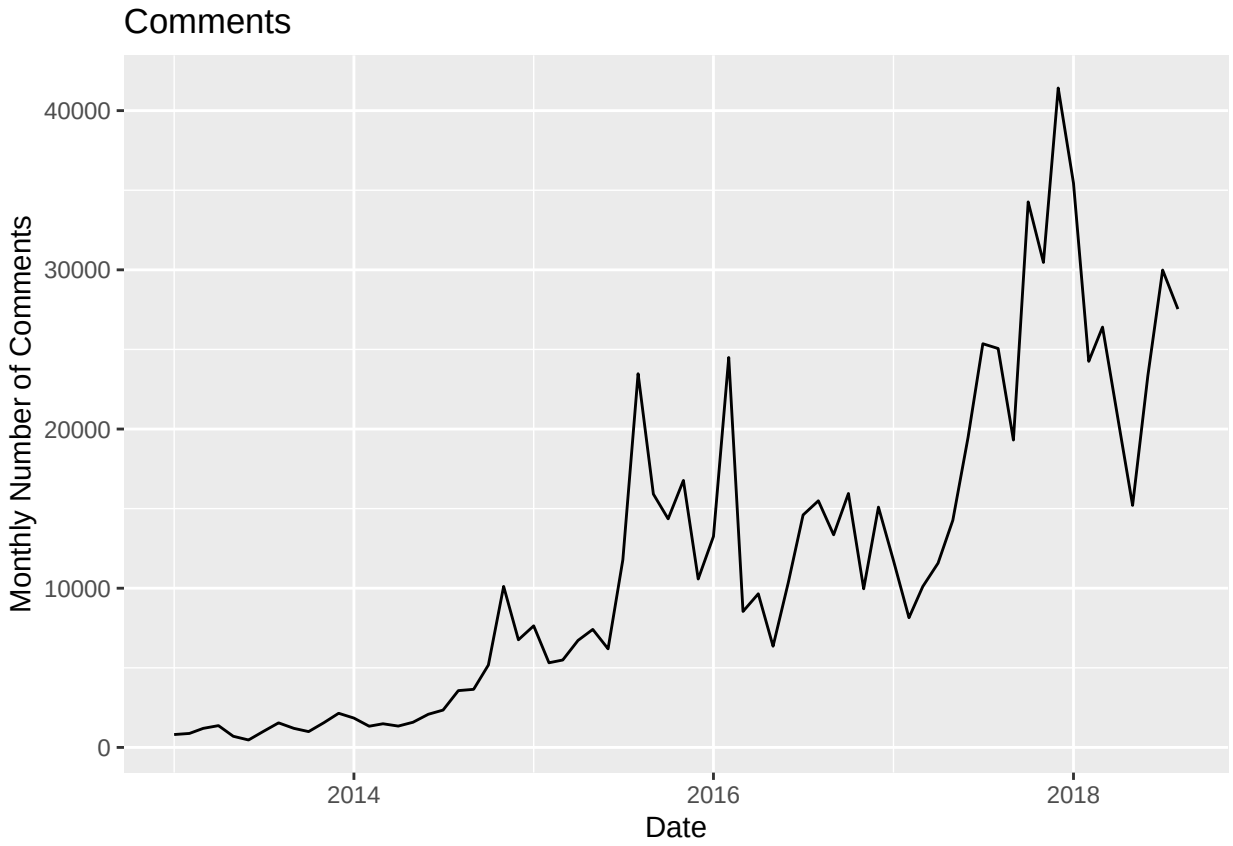


Table 1

Use the page_host_country variable in metadata.csv (a page-level dataset) to calculate how many pages are associated with each host country.Create a table with each country name and the number of pages ordered by most frequent host countries

```
metadata %>%
  count(page_host_country) %>%
  arrange(desc(n)) %>%
  top_n(n=20) %>%
  rename("Number of Pages" = n,  "Host Country" = page_host_country) %>%
  knitr::kable(format = "latex")
```

Host Country	Number of Pages
NA	53
Jordan	42
Germany	38
Turkey	32
Europe	28
Canada	14
Lebanon	14
Egypt	12
Sweden	11
West	10
Greece	9
Netherlands	9
Saudi Arabia	6
Cyprus	4
Norway	4
US	4
Australia	3
Austria	3
Belgium	3
Denmark	3
Sudan	3
Switzerland	3

Figure 2a

Group posts.csv data by post_type (source) variable and calculate the total number of posts of each source type (official, unofficial, news). Use grouped data to plot a histogram of the total volume of posts by source (post_type)

```
posts %>%
  ggplot(aes(x = post_type)) +
  geom_histogram(stat = "count") +
  labs(x = "Source",
       y = "Total Volume of Posts")
```

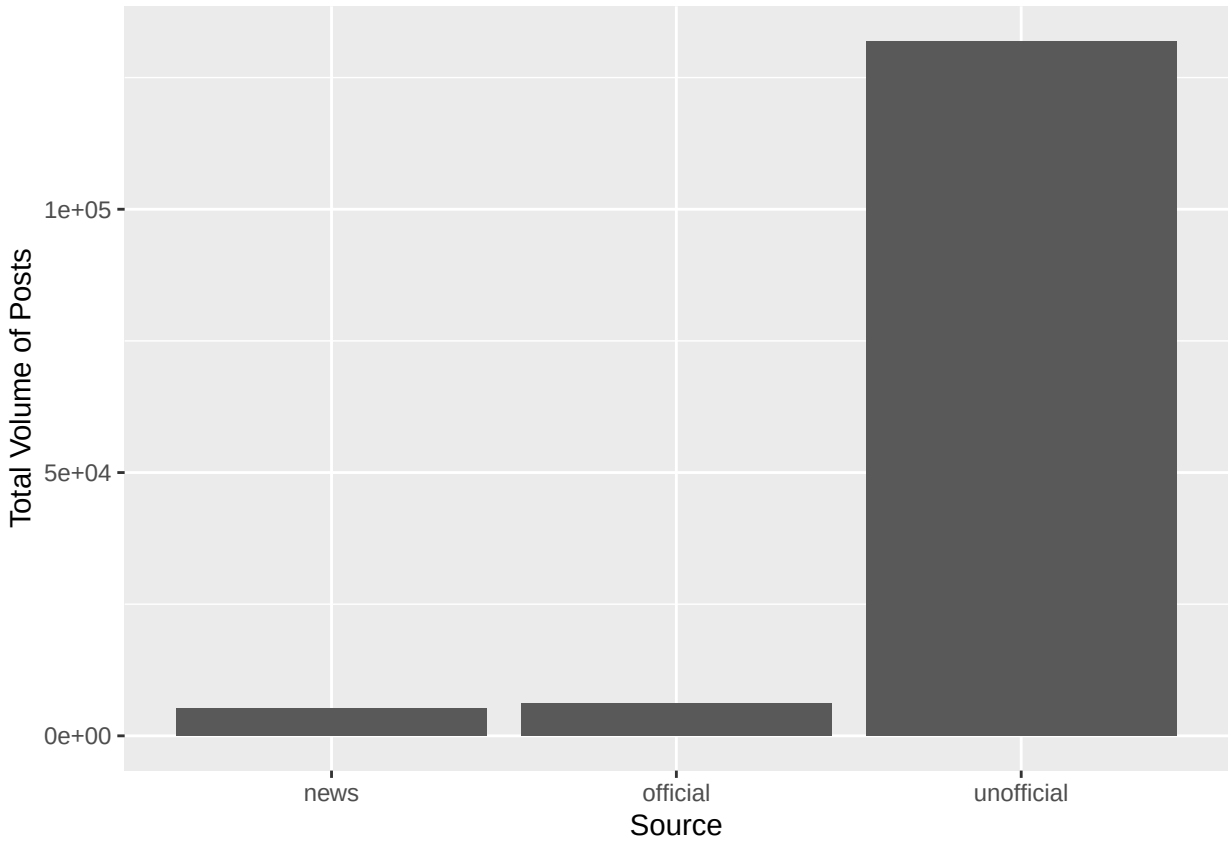


Figure 2b

Group posts.csv data by post_type (source) variable and month using the daily date variable and calculate the total number of posts of each source type in each month (official, unofficial, news). Use grouped data (by source and month) to plot the monthly volume of posts by source over time (plotting the raw data points and using loess smoothing)

```
posts %>%
  group_by(post_type) %>%
  mutate(month_ = floor_date(date, "months")) %>%
  count(month_) %>%
  ggplot(aes(x = month_, y = n)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  facet_wrap(vars(post_type), scales = "free_y") +
  labs(x = "Date", y = "Monthly Volume of Posts")
```

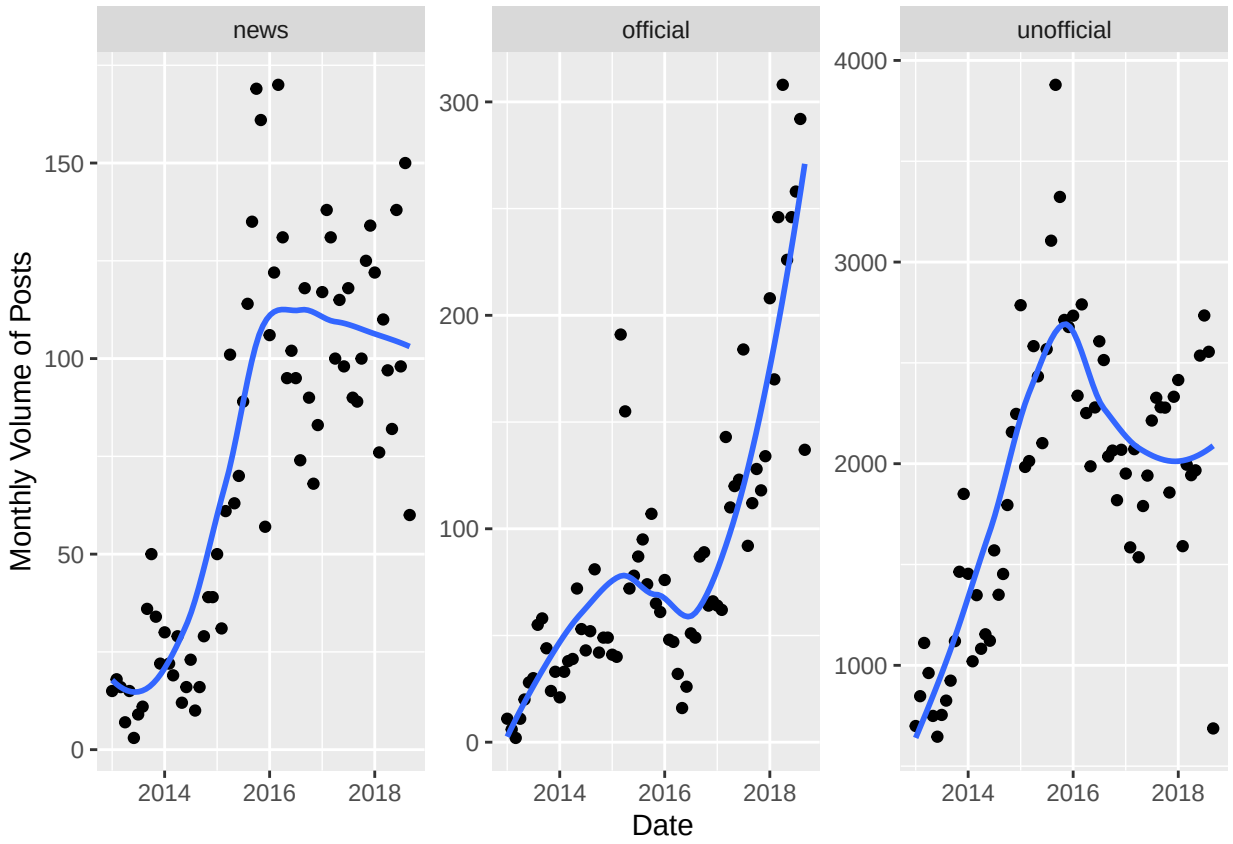


Figure 3a

Group posts.csv by post_type (source) variable and calculate the total engagement for each source (post_type) using the engagement_fb variable. Divide total engagement by number of posts for each source type (eg official posts total engagement / number of official posts) to calculate mean engagement per post by source type. Plot histogram of mean engagement per post by source type

```
sum_eng <- posts %>%
  group_by(post_type) %>%
  summarize(sum_eng = sum(engagement_fb))

count_type <- posts %>%
  count(post_type)

mean_eng <- sum_eng %>%
  left_join(count_type, by = "post_type") %>%
  mutate(mean = round(sum_eng/n))

mean_eng %>%
  ggplot(aes(x = post_type, y = mean, fill = post_type)) +
  geom_col() +
  labs(title = "Average Engagement per Post",
       y = "Mean Engagement per Post",
       x = "Source")
```

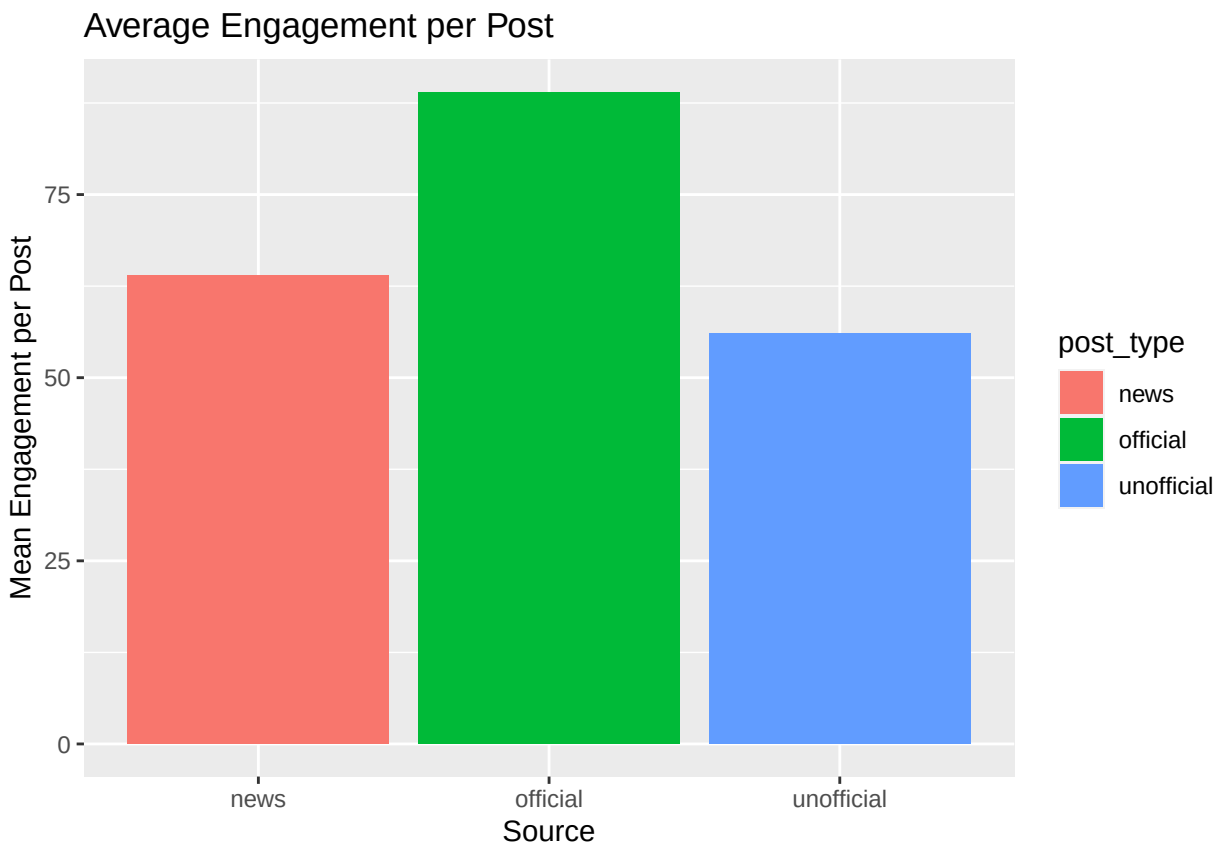


Figure 3b

Group posts.csv by month (using the date variable) and source (using the post_type) variable and calculate posts per month by source and total engagement per month by source. For each source, divide total engagement per month by posts per month to calculate mean monthly engagement by source type. Plot mean monthly engagement per post by source type over time (plotting the raw data points and using loess smoothing).

```
posts <- posts %>%
  mutate(month_ = floor_date(date, "months"))

sum_eng_month <- posts %>%
  group_by(post_type, month_) %>%
  summarize(sum_eng = sum(engagement_fb))

count_type_month <- posts %>%
  count(post_type, month_)

mean_eng_month <- sum_eng_month %>%
  left_join(count_type_month, by = c("post_type", "month_")) %>%
  mutate(mean = sum_eng/n)

mean_eng_month %>%
  ggplot(aes(x = month_, y = mean)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  facet_wrap(vars(post_type)) +
```

```
  labs(x = "Date", y = "Mean Monthly Engagement Per Post")
```

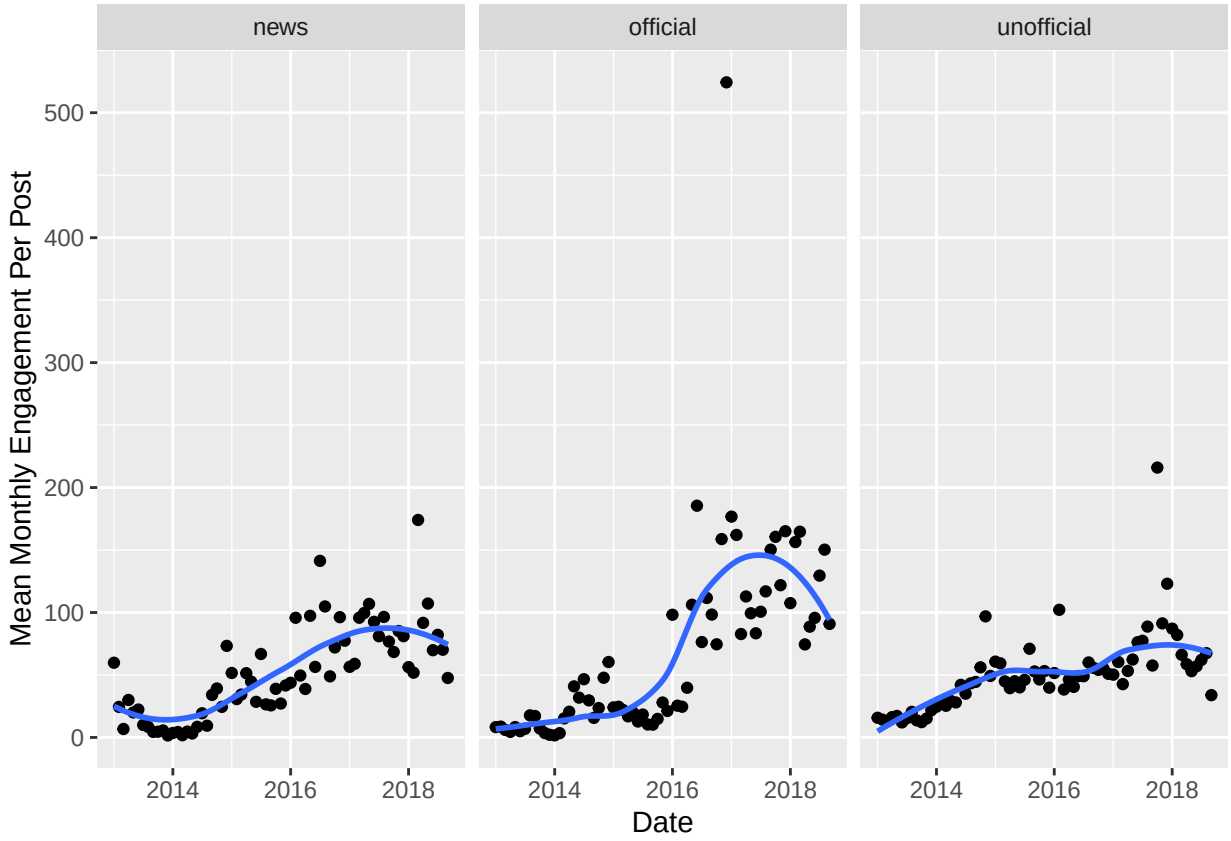


Figure 3c (left)

Subset posts.csv to "official" posts only using the post_type (source) variable. Group the subsetted official data by the pre_2016_source variable, and calculate the total engagement for each type. Also calculate the total number of posts of each type (pre vs post 2016 sources). Divide total engagement for each type by total posts to calculate mean engagement per post for pre vs. post 2016 sources. Plot a histogram of mean engagement per post for pre vs. post 2016 sources.

```
posts %>%
  filter(post_type == "official") %>%
  group_by(pre_2016_source) %>%
  summarize(sum_eng = sum(engagement_fb)) -> sum_eng_official

posts %>%
  filter(post_type == "official") %>%
  count(pre_2016_source) -> count_official

sum_eng_official %>%
  left_join(count_official, by = "pre_2016_source") %>%
  mutate(mean_eng = sum_eng / n) -> mean_eng_official

mean_eng_official %>%
  ggplot(aes(x = pre_2016_source, y = mean_eng)) +
  geom_col(aes(fill = pre_2016_source)) +
  labs(x = "Source",
```

```
       y = "Mean Engagement Per Post",
       title = "Average Engagement with Official Sources Pre and Post 2016")
```

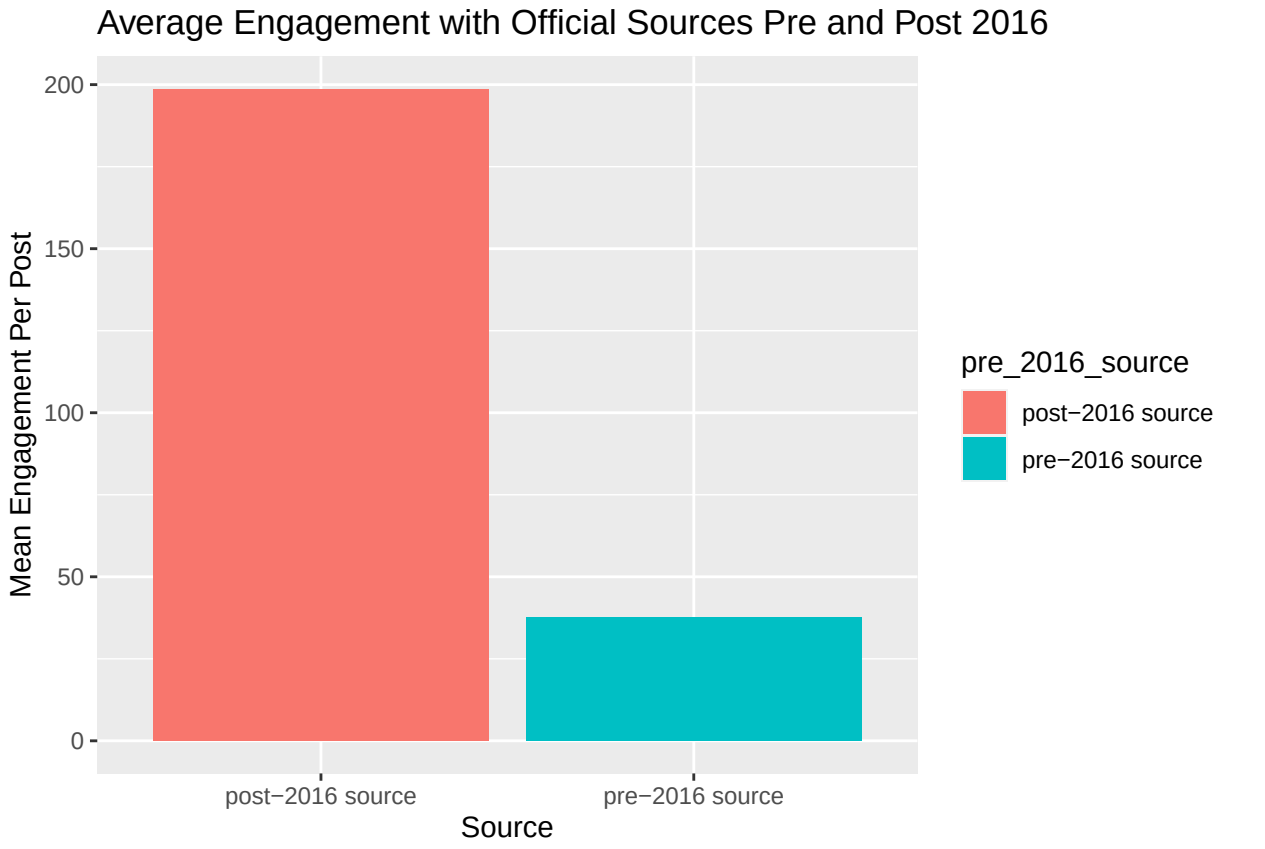


Figure 3c (right)

Group posts.csv by month (using the date variable) and source (using the pre_post_2016 variable) and calculate posts per month by source and total engagement per month by source. Divide engagement per month by source type (pre vs. post 2016) by posts per month by source to calculate mean monthly engagement by source type. Plot mean monthly engagement per post by source type (pre vs. post 2016) over time (plotting the raw data points and using loess smoothing).

```
sum_eng_month <- posts %>%
  filter(post_type == "official") %>%
  group_by(pre_2016_source, month_) %>%
  summarize(sum_eng = sum(engagement_fb))

count_type_month <- posts %>%
  filter(post_type == "official") %>%
  count(pre_2016_source, month_)

mean_eng_month <- sum_eng_month %>%
  left_join(count_type_month, by = c("pre_2016_source", "month_")) %>%
  mutate(mean = sum_eng/n)

mean_eng_month %>%
  ggplot(aes(x = month_, y = mean)) +
  geom_point() +
```

```
  geom_smooth(se = FALSE) +
  facet_wrap(vars(pre_2016_source)) +
  labs(x = "Date", y = "Mean Monthly Engagement Per Post")
```

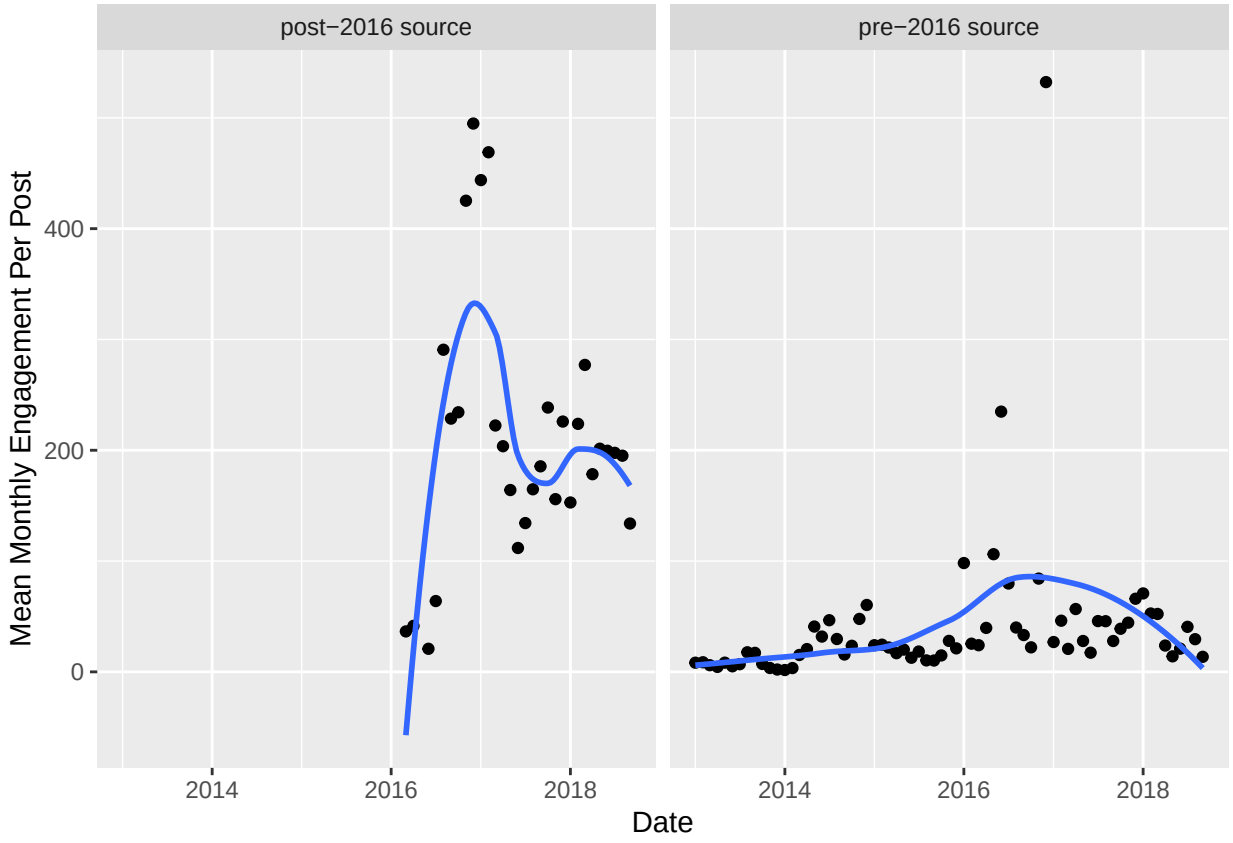


Figure 4a

Group posts.csv by post_type (source), as well as the following dummy variables, which indicate whether or not a post references each topic: land_travel, sea_travel, smuggling, asylum, return, camp, employment, scholarship, housing, healthcare to calculate the proportion of posts of each source type (official, unofficial, news) referencing each topic. Plot a histogram of the proportion of posts of each source type referencing each topic.

```
posts_prop <- posts %>%
  group_by(post_type) %>%
  summarize(sum_asylum = sum(asylum) / length(asylum),
            sum_employment = sum(employment) / length(employment),
            sum_housing = sum(housing) / length(housing),
            sum_healthcare = sum(healthcare) / length(healthcare),
            sum_camp = sum(camp) / length(camp),
            sum_return = sum(return) / length(return),
            sum_land_travel = sum(land_travel) / length(land_travel),
            sum_sea_travel = sum(sea_travel) / length(sea_travel),
            sum_scholarship = sum(scholarship) / length(scholarship),
            sum_smuggling = sum(smuggling) / length(smuggling))

posts_prop %>%
  pivot_longer(cols=sum_asylum:sum_smuggling) %>%
```

```
  ggplot(aes(x = name, y = value, fill = post_type))+
  geom_col(position = position_dodge()) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = "Topic",
       y = "Proportion of Posts",
       title = "Salience",
       fill = "")
```

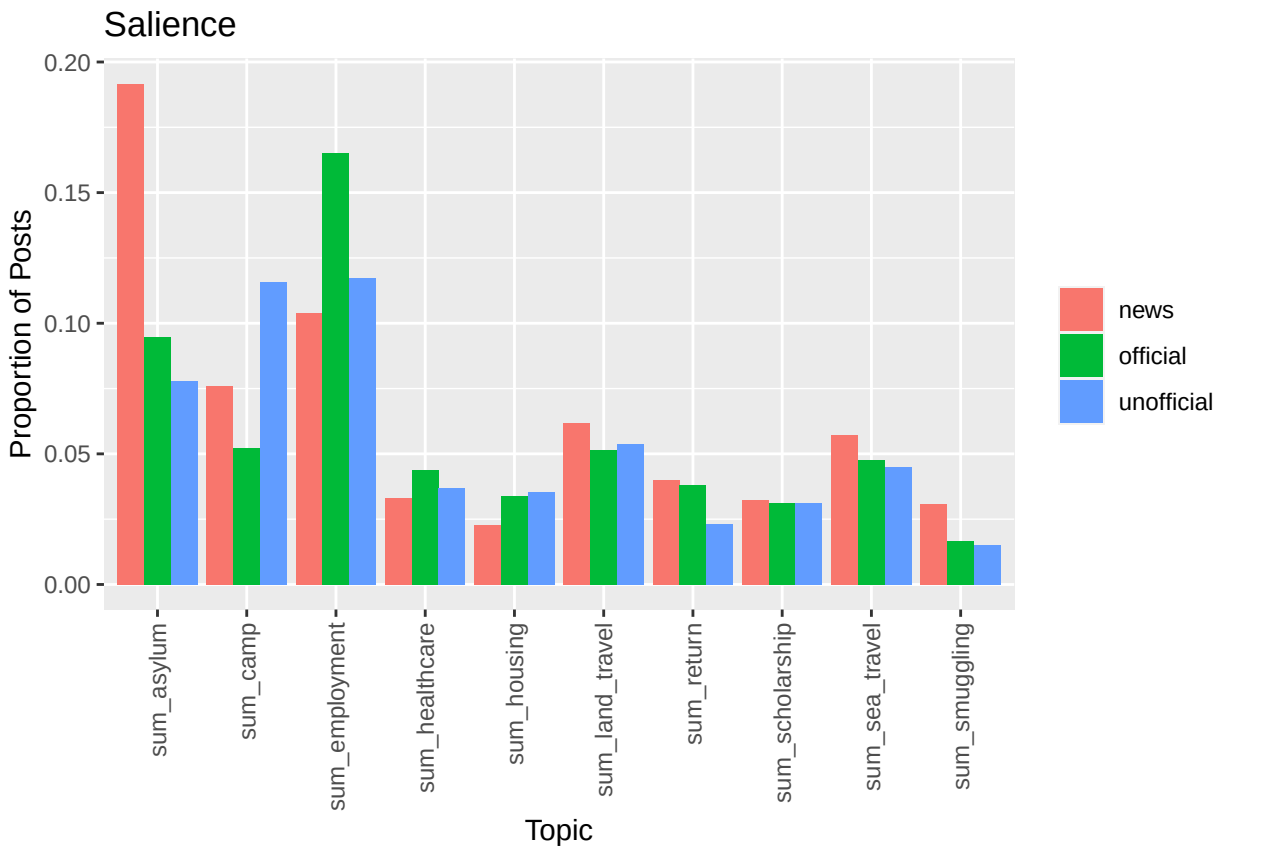


Figure 4b

For each topic (using the land_travel, sea_travel, smuggling, asylum, return, camp, employment, scholarship, housing, healthcare dummy variables) calculate average engagement per post by post_type (source). Plot a histogram of average engagement per post for each source type referencing each topic

```
asylum_eng <- posts %>%
  group_by(post_type) %>%
  filter(asylum == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(asylum)) %>%
  mutate(topic = "asylum")

education_eng <- posts %>%
  group_by(post_type) %>%
  filter(scholarship == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(scholarship)) %>%
  mutate(topic = "education")

employment_eng <- posts %>%
```

```
  group_by(post_type) %>%
  filter(employment == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(employment)) %>%
  mutate(topic = "employment")

healthcare_eng <- posts %>%
  group_by(post_type) %>%
  filter(healthcare == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(healthcare)) %>%
  mutate(topic = "healthcare")

housing_eng <- posts %>%
  group_by(post_type) %>%
  filter(housing == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(housing)) %>%
  mutate(topic = "housing")

land_travel_eng <- posts %>%
  group_by(post_type) %>%
  filter(land_travel == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(land_travel)) %>%
  mutate(topic = "land_travel")

refugee_camps_eng <- posts %>%
  group_by(post_type) %>%
  filter(camp == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(camp)) %>%
  mutate(topic = "refugee camps")

return_eng <- posts %>%
  group_by(post_type) %>%
  filter(return == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(return)) %>%
  mutate(topic = "return")

sea_travel_eng <- posts %>%
  group_by(post_type) %>%
  filter(sea_travel == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(sea_travel)) %>%
  mutate(topic = "sea travel")

smuggling_eng <- posts %>%
  group_by(post_type) %>%
  filter(smuggling == TRUE) %>%
  summarize(mean_eng = sum(engagement_fb)/length(smuggling)) %>%
  mutate(topic = "smuggling")

asylum_eng %>%
  rbind(education_eng) %>%
  rbind(employment_eng) %>%
  rbind(healthcare_eng) %>%
  rbind(housing_eng) %>%
  rbind(land_travel_eng) %>%
```

```
  rbind(refugee_camps_eng) %>%
  rbind(return_eng) %>%
  rbind(sea_travel_eng) %>%
  rbind(smuggling_eng) %>%
  ggplot(aes(x = topic, y = mean_eng, fill = post_type)) +
  geom_col(position = position_dodge()) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(title = "Average Engagement per Post",
       x = "Topic",
       y = "Average Engagement per Post")
```

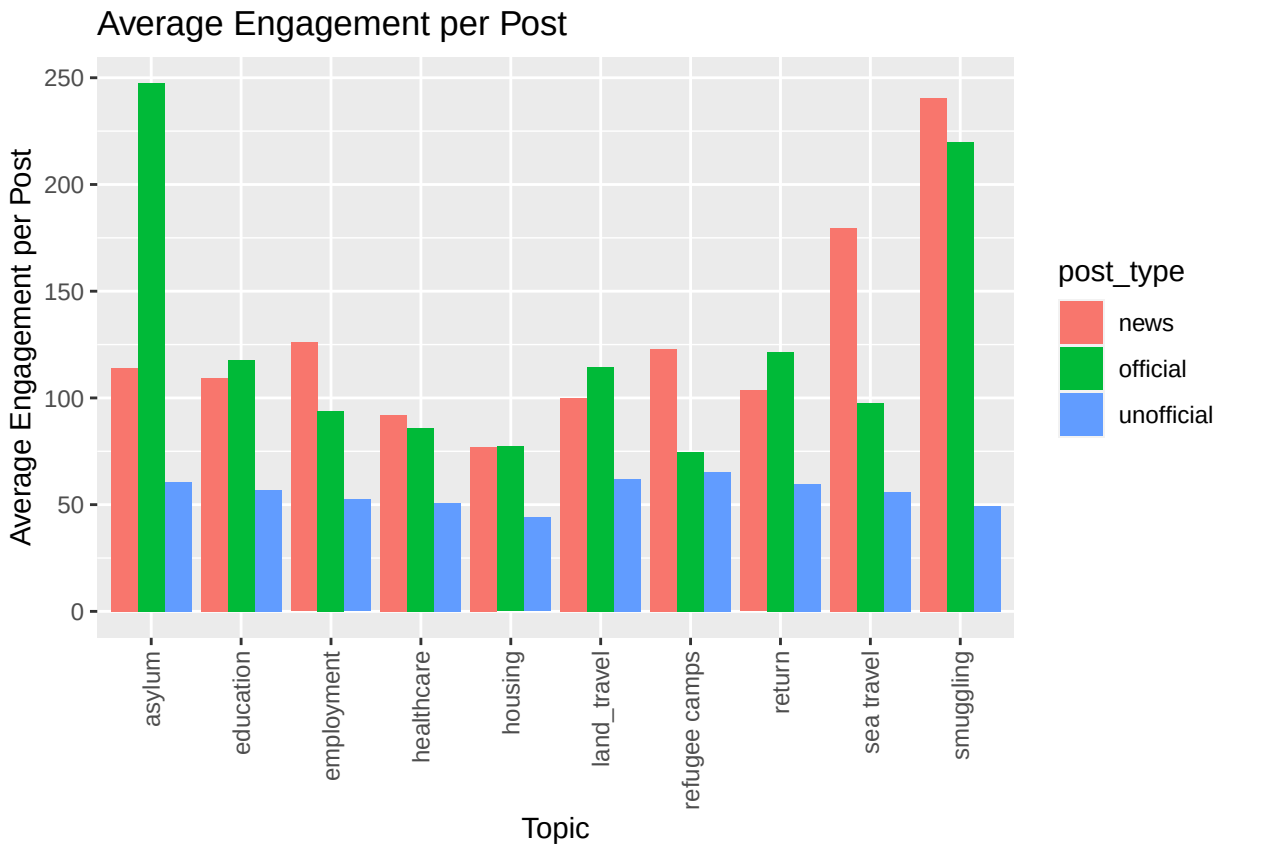


Figure 5

For each topic (using the land_travel, sea_travel, smuggling, asylum, return, camp, employment, scholarship, housing, healthcare dummy variables) calculate the monthly (using the daily date variable) proportion of posts referencing the topic by post_type (source). Exclude news from the analysis to compare just official vs. unofficial sources. Plot the monthly proportion of posts referencing each topic over time comparing official vs. unofficial sources. Create a new plot for each topic (plotting the raw data points and using loess smoothing).

```
monthly_topics <- posts %>%
  filter(post_type != "news") %>%
  group_by(post_type, month_) %>%
  summarize(sum_asylum = sum(asylum) / length(asylum),
            sum_employment = sum(employment) / length(employment),
            sum_housing = sum(housing) / length(housing),
            sum_healthcare = sum(healthcare) / length(healthcare),
```

```
            sum_camp = sum(camp) / length(camp),
            sum_return = sum(return) / length(return),
            sum_land_travel = sum(land_travel) / length(land_travel),
            sum_sea_travel = sum(sea_travel) / length(sea_travel),
            sum_scholarship = sum(scholarship) / length(scholarship),
            sum_smuggling = sum(smuggling) / length(smuggling))

monthly_topics %>%
  pivot_longer(cols=sum_asylum:sum_smuggling) %>%
  ggplot(aes(x=month_, y = value, group = post_type)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  facet_wrap(vars(name), ncol=2, scales = "free_y") +
  theme_minimal() +
  labs(
    x = "Date",
    y = "Monthly Proportion of Posts",

  )
```

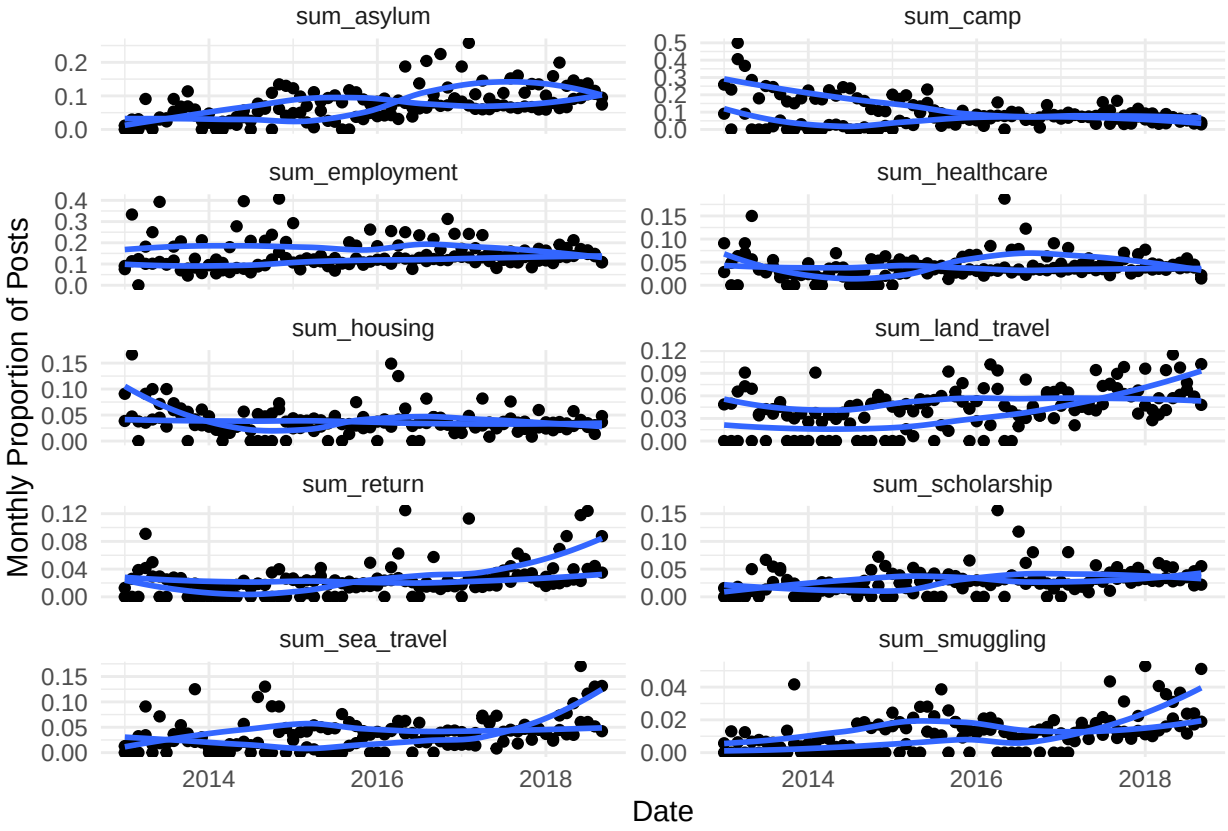


Figure 6

Subset data to official posts only using the post_type variable. For each topic (using the land_travel, sea_travel, smuggling, asylum, return, camp, employment, scholarship, housing, healthcare dummy variables) calculate average engagement per post by pre_2016_source variable. Plot a histogram of average engagement per post for each source type (pre vs. post 2016) referencing each topic.

```r
posts_prop <- posts %>%
  filter(post_type == "official") %>%
  group_by(pre_2016_source) %>%
  summarize(sum_asylum = sum(asylum) / length(asylum),
            sum_employment = sum(employment) / length(employment),
            sum_housing = sum(housing) / length(housing),
            sum_healthcare = sum(healthcare) / length(healthcare),
            sum_camp = sum(camp) / length(camp),
            sum_return = sum(return) / length(return),
            sum_land_travel = sum(land_travel) / length(land_travel),
            sum_sea_travel = sum(sea_travel) / length(sea_travel),
            sum_scholarship = sum(scholarship) / length(scholarship),
            sum_smuggling = sum(smuggling) / length(smuggling))

posts_prop %>%
  pivot_longer(cols=sum_asylum:sum_smuggling) %>%
  ggplot(aes(x = name, y = value, fill = pre_2016_source))+
  geom_col(position = position_dodge()) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(x = "Topic",
       y = "Proportion of Posts",
       fill = "")
```

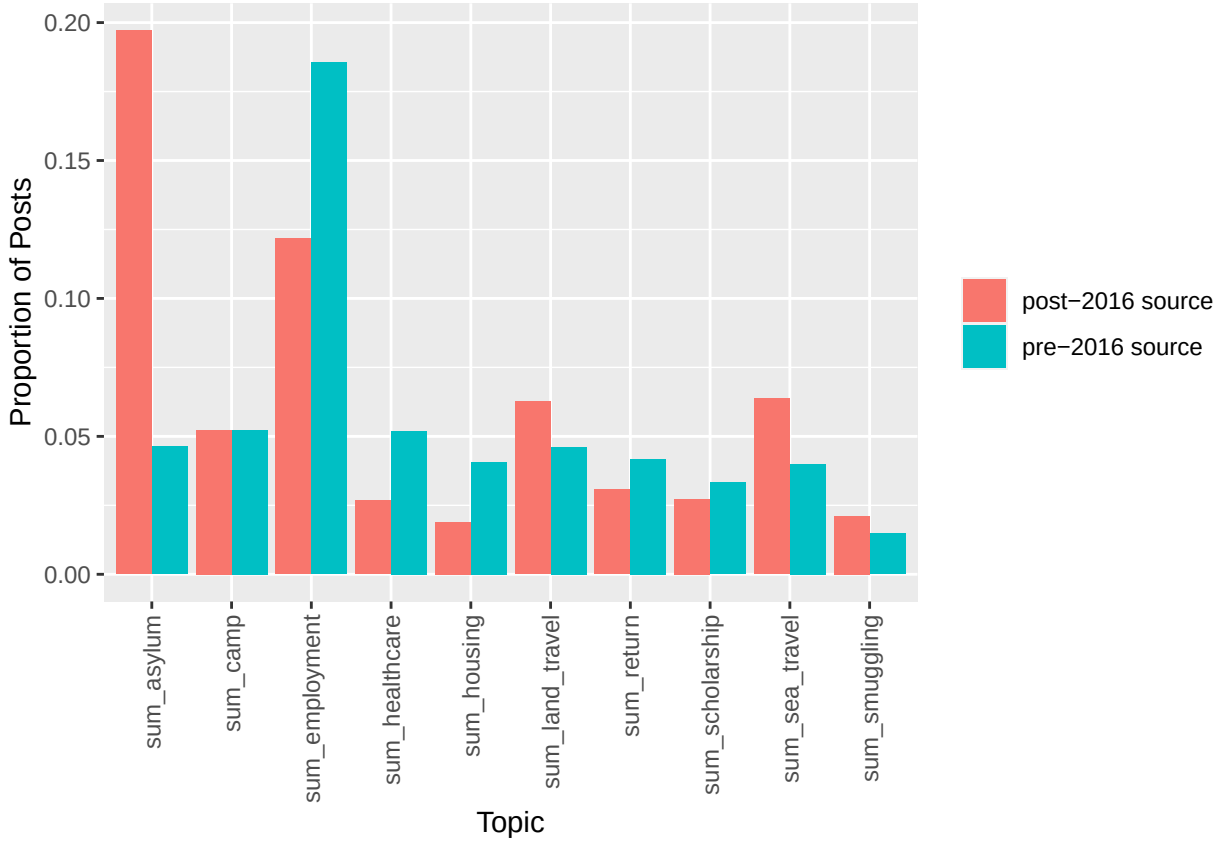


Figure 7a

Using the post_tone.csv dataset, group by source (using the post_type variable) and calculate the average cosine similarity (encouraging – discouraging) by taking the mean of the diff_ar variable by source (post_type).

Note you will have to remove NAs in order to calculate the mean. There are NAs in posts that do not contain the target words related to migration. Plot a histogram of mean cosine similarity by source (post_type).

```
post_tone <- read_csv("post_tone.csv")
monthly_tone <- read_csv("monthly_tone.csv")

post_tone %>%
  group_by(post_type) %>%
  filter(!is.na(diff_ar)) %>%
  summarize(mean_cos = mean(diff_ar)) %>%
  ggplot(aes(x = post_type, y = mean_cos, fill = post_type)) +
  geom_col() +
  labs(x = "Source",
       y = "Cosine Similarity",
       title = "Tone by Source",
       legend = "")
```

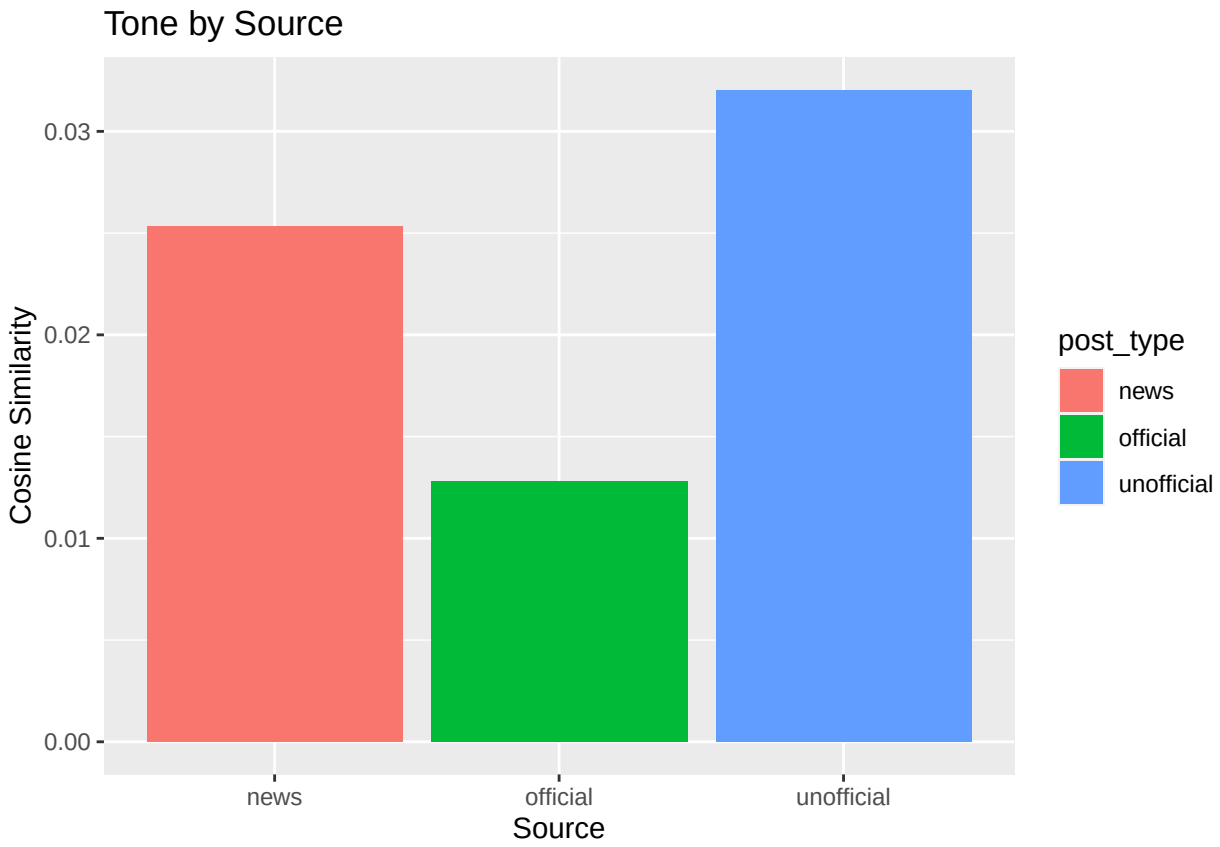


Figure 7b

Subset the monthly_tone.csv dataset to only official and unofficial posts using the post_type variable (i.e. exclude news). Plot the date (seqvar, a monthly date varaible) on the x axis and the monthly cosine similarity (encouraging- discouraging) on the y axis (using the dif_ar_combined) variable. Group by post_type (source) and subset to remove news data to plot official vs. unofficial sources on the same plot (plotting the raw data points and using loess smoothing).

```
monthly_tone %>%
  filter(source != "news") %>%
  ggplot(aes(seqvar, diff_ar_combined, color=source)) +
```

```
  geom_point() +
  geom_smooth(se=FALSE) +
  labs(x = "Date",
       y = "Monthly Cosine Similarity",
       title = "Tone by Month")
```

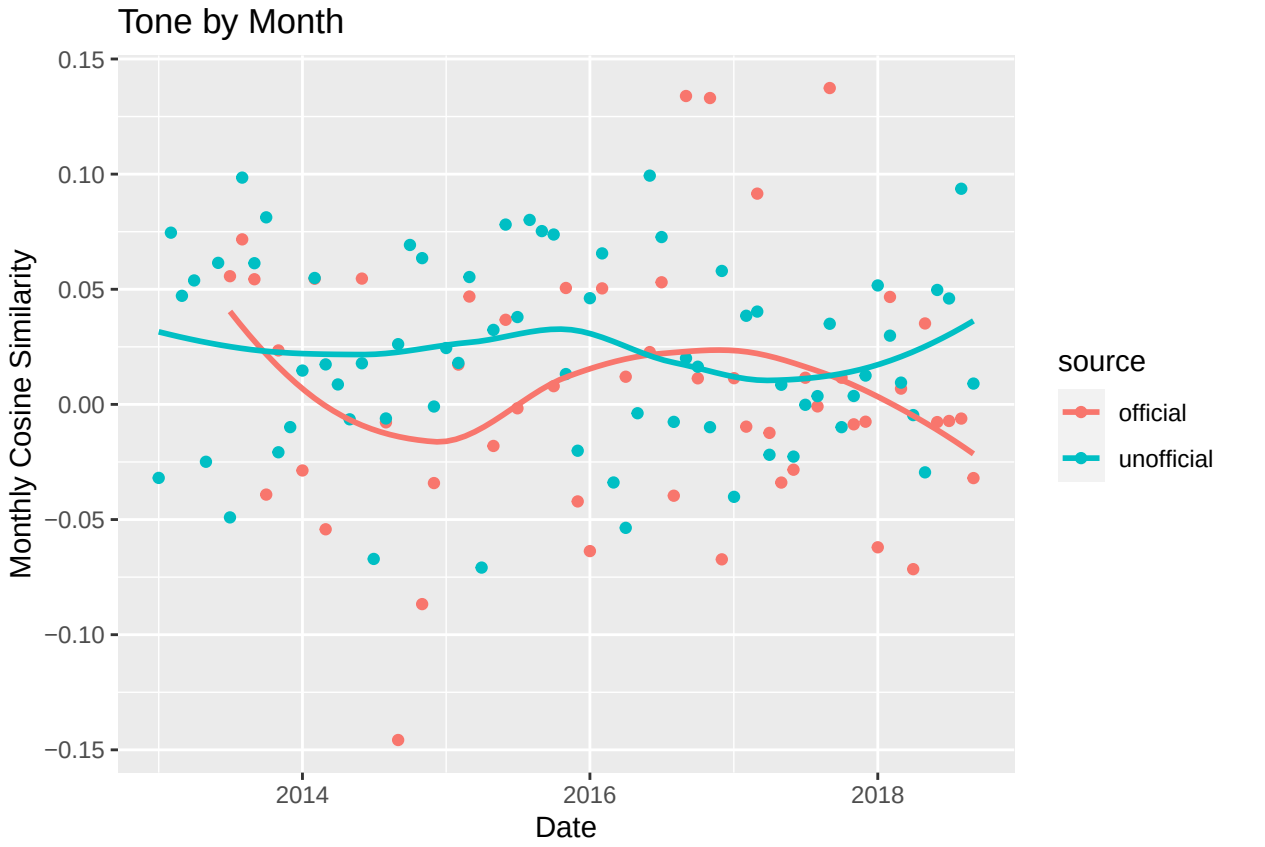


Figure 8

Using post_tone.csv and the dif_ar variable, subset data to the top quartile and bottom quartile of posts (most encouraging and most discouraging). First summarize the dif_ar variable to identify posts in the top and bottom quartile. Calculate mean engagement per post for posts in the top quartile (encouraging posts) and posts in the bottom quartile (discouraging posts) grouped by source (post_type variable).

```
upper_quart <- quantile(post_tone$diff_ar, 0.75, na.rm = TRUE)
lower_quart <- quantile(post_tone$diff_ar, 0.25, na.rm = TRUE)

post_tone %>%
  group_by(post_type) %>%
  mutate(quart = case_when(diff_ar >= upper_quart ~ "most_encouraging",
                           diff_ar <= lower_quart ~ "most_discouraging",
                           TRUE ~ NA)) -> post_tone_quart

post_tone_quart %>%
  filter(!is.na(quart)) %>%
  group_by(post_type, quart) %>%
  summarize(mean_eng = mean(engagement_fb)) %>%
  ggplot(aes(x = post_type, y = mean_eng, fill = quart)) +
```

```
  geom_col(position=position_dodge()) +
  labs(x = "Source",
       y = "Average Engagement Per Post",
       fill = "")
```

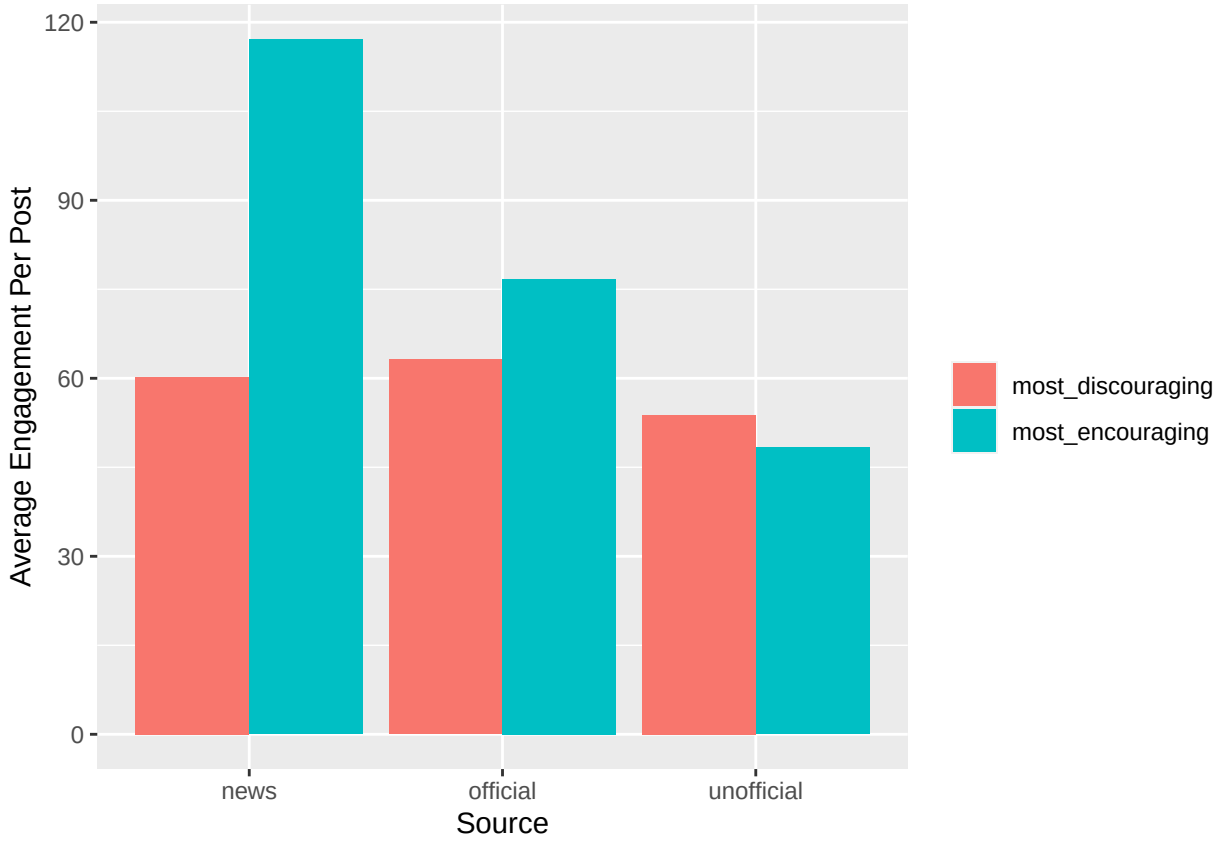
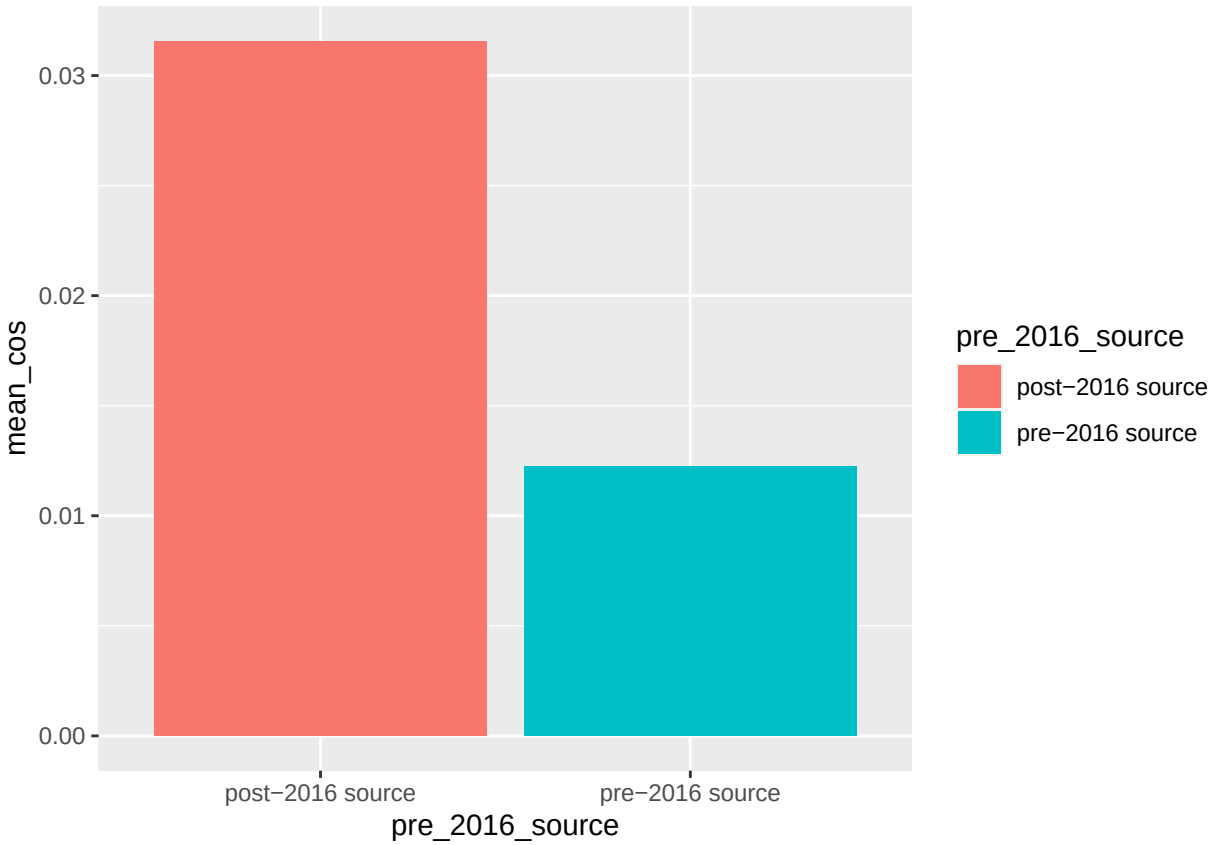


Figure 9

Using the post_tone.csv dataset, group by source (using the pre_2016_source variable) and calculate the average cosine similarity (encouraging – discouraging) by taking the mean of the diff_ar variable by source (pre_2016_source). Note you will have to remove NAs from diff_ar in order to calculate the mean. There are NAs in posts that do not contain the target words related to migration. Plot a histogram of mean cosine similarity by source (pre_2016_source).

```
post_tone %>%
  group_by(pre_2016_source) %>%
  filter(!is.na(diff_ar)) %>%
  summarize(mean_cos = mean(diff_ar)) %>%
  ggplot(aes(x = pre_2016_source, y = mean_cos, fill = pre_2016_source)) +
  geom_col()
```


0.03
0.02
0.01
0.00
mean_cos
post−2016 source
pre−2016 source
pre_2016_source
pre_2016_source
post−2016 source
pre−2016 source