STV4030A: All home assignments

Candidate number 117

2024-09-24

Week 1

Loading necessary packages.

```
library(scales)
library(dplyr)
library(haven)
library(tidyverse)
library(tinytable)
library(gt)
```

Task 1

Loading dataset.

```
name_mapping <- list(
  "merged_r3_data" = "data_round_3",
  "merged_r4_data" = "data_round_4",
  "merged-round-5-data-34-countries-2011-2013-last-update-july-2015_0" =
    "data_round_5",
  "merged_r6_data_2016_36countries2" = "data_round_6",
  "r7_merged_data_34ctry.release" = "data_round_7"
)

files <- list.files(pattern = "*.sav*")

for (i in 1:length(files)) {
   file_name <- files[i]
   data_name <- sub("*.sav*", "", files[i])
```

```
  temp_data <- read_sav(file_name, encoding = "latin1")
  if (data_name %in% names(name_mapping)) {
    assign(name_mapping[[data_name]], temp_data, envir = .GlobalEnv)}
  else {
    warning(paste("No mapping found for", data_name))}
  }
```

Task 2

Renaming variables.

```
data_round_3 <- data_round_3 %>%
  rename(horizontal_judicial_power = q41,
         close_to_party = q86)

data_round_4 <- data_round_4 %>%
  rename(horizontal_judicial_power = Q37,
         close_to_party = Q86)

data_round_5 <- data_round_5 %>%
  rename(horizontal_judicial_power = Q40,
         close_to_party = Q89B)

data_round_6 <- data_round_6 %>%
  rename(horizontal_judicial_power = Q38,
         close_to_party = Q90B)

data_round_7 <- data_round_7 %>%
  rename(horizontal_judicial_power = Q33,
         close_to_party = Q88B )
```

Task 3

Filtering out Ghana-observations. Selecting only two variables, and creating a new one.

```
data_round_3 <- data_round_3 %>%
  filter(country == 4) %>%
  select(horizontal_judicial_power, close_to_party) %>%
  mutate(horizontal_judicial_power = as.numeric(horizontal_judicial_power),
         close_to_party = as.numeric(close_to_party)) %>%
```

```
  mutate(round = 3)

data_round_4 <- data_round_4 %>%
  filter(COUNTRY == 5) %>%
  select(horizontal_judicial_power, close_to_party) %>%
  mutate(horizontal_judicial_power = as.numeric(horizontal_judicial_power),
         close_to_party = as.numeric (close_to_party)) %>%
  mutate(round = 4)


data_round_5 <- data_round_5 %>%
  filter(COUNTRY_ALPHA == 11) %>%
  select(horizontal_judicial_power, close_to_party) %>%
  mutate(horizontal_judicial_power = as.numeric(horizontal_judicial_power),
         close_to_party = as.numeric(close_to_party)) %>%
  mutate(round = 5)


data_round_6 <- data_round_6 %>%
  filter(COUNTRY == 11) %>%
  select(horizontal_judicial_power, close_to_party) %>%
  mutate(horizontal_judicial_power = as.numeric(horizontal_judicial_power),
         close_to_party = as.numeric(close_to_party)) %>%
  mutate(round = 6)


data_round_7 <- data_round_7 %>%
  filter(COUNTRY == 10) %>%
  select(horizontal_judicial_power, close_to_party) %>%
  mutate(horizontal_judicial_power = as.numeric(horizontal_judicial_power),
         close_to_party = as.numeric(close_to_party)) %>%
  mutate(round = 7)


data_round_3_7 <- rbind(data_round_3, data_round_4, data_round_5,
                        data_round_6, data_round_7)
```

Task 4

Datawrangling.

```r
data_round_3_7 <- data_round_3_7 %>%
  mutate(party_supporters = case_when(
    round == 3 & close_to_party == 181 ~ "NDC supporters",
    round == 3 & close_to_party == 182 ~ "NPP supporters",
    round >= 4 & round <= 7 & close_to_party == 261 ~ "NDC supporters",
    round >= 4 & round <= 7 & close_to_party == 262 ~ "NPP supporters")) %>%
  mutate(year = case_when(
    round == 3 ~ 2005,
    round == 4 ~ 2008,
    round == 5 ~ 2012,
    round == 6 ~ 2014,
    round == 7 ~ 2017)) %>%
  mutate(partisan = case_when(
      (party_supporters == "NDC supporters" &
         year %in% c(2005, 2008, 2017)) |
      (party_supporters == "NPP supporters" &
         year %in% c(2012, 2014)) ~ "Out-partisan",
      (party_supporters == "NPP supporters" &
         year %in% c(2005, 2008, 2017)) |
      (party_supporters == "NDC supporters" &
         year %in% c(2012, 2014)) ~ "Co-partisan")) %>%
  mutate(horizontal_judicial_power_binary = case_when(
      horizontal_judicial_power %in% c(3, 4) ~ 1,
      horizontal_judicial_power %in% c(1, 2, 5) ~ 0))
```

Task 5

```r
mean_data_round_3_7 <- data_round_3_7 %>%
  group_by(party_supporters, year) %>%
  summarize(avarage = mean(horizontal_judicial_power_binary, na.rm = TRUE)) %>%
  drop_na()
```

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

```r
mean_data_round_3_7 %>%
  mutate(avarage = avarage*100) %>%
  gt() %>%
  cols_label(avarage = "Percentage avarage",
```

```
            year = "Year") %>%
  fmt_number(decimals = 0, use_seps = FALSE) %>%
  tab_header(title = md("Public views on judicial power in Ghana, by year and party loyalty")
             subtitle = md("Percentage of positive responses to the statement: *The Presid
```

Public views on judicial power in Ghana, by year and party loyalty

Percentage of positive responses to the statement: *The President must always obey the laws and the courts, even if he thinks they are wrong.*

Year	Percentage avarage
NDC supporters	
2005	82
2008	70
2012	71
2014	65
2017	77
NPP supporters	
2005	74
2008	65
2012	79
2014	74
2017	76

Task 6

```
label1 <- c("Out-partisan","Out-partisan",
            "Co-partisan","Co-partisan", "Out-partisan",
            "Co-partisan","Co-partisan",
            "Out-partisan","Out-partisan","Co-partisan")

mean_data_round_3_7 <- data.frame(mean_data_round_3_7, label1)

ggplot(data = mean_data_round_3_7 %>%
         drop_na(),
       aes(x = year, y =  avarage,
           shape = party_supporters, linetype = party_supporters)) +
  geom_line() +
```

```
geom_point() +
geom_vline(xintercept = 2008.25,
           linetype = "dashed",
            alpha = 0.8)+
geom_vline(xintercept = 2016.25,
            linetype = "dashed",
            alpha = 0.8)+
geom_text(label = label1,
           nudge_y = 0.01,
           size = 4,
           check_overlap = TRUE)+
scale_linetype_manual(values=c(4,2)) +
scale_shape_manual(values = c(24,16))+
scale_x_continuous(limits = c(2004, 2019),
                   breaks = seq(2004,2019, by = 1))+
scale_y_continuous(limits = c(0.55, 0.85),
                   breaks = seq(0.55,0.85, by = 0.05))+
labs(y = "Proportion that Support Horizontal Power\n",
     x = "\nYear")+
theme_classic()+
theme(legend.position = c(0.094, 0.094))+
theme(legend.background = element_rect(
  size=0.5, color = "black",linetype="solid"))+
guides(shape = guide_legend(title = NULL),
       linetype = guide_legend(title = NULL)) +
annotate("text", label = "NPP president", x = 2006, y = 0.85)+
annotate("text", label = "NDC president", x = 2012, y = 0.85)+
annotate("text", label = "NPP president", x = 2018, y = 0.85)
```

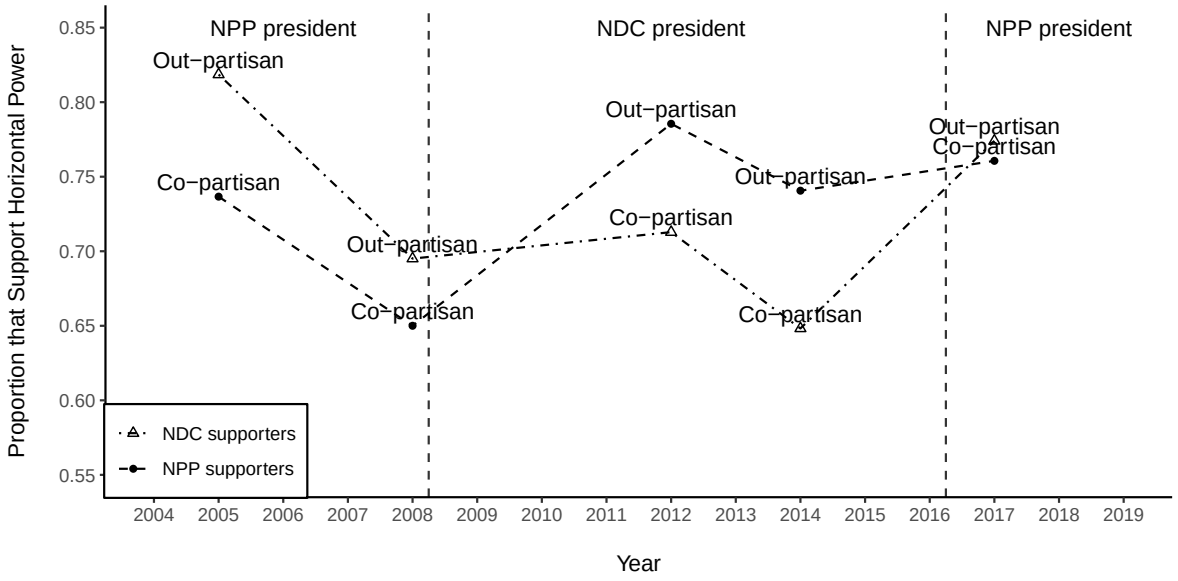

Figure 1: Does Public Support for Judicial Power Depend on Who Is in Political Power? Testing a Theory of Partisan Alignment in Africa. American Political Science Review 114 (1): 144–63.

Task 7

Figure 1 replicates the figure from the article "Does Public Support for Judicial Power Depend on Who Is in Political Power? (Bartels and Kramon 2020).

Task 8

```
year_of_survey <- c("Round 1", "Round 2", "Round 3", "Round 4",
                    "Round 5", "Round 6", "Round 7")

year <- c(1999, 2002, 2005, 2008,
          2012, 2014, 2017)

president <- c("Jerry Rawlings", "John Kufuor", "John Kufuor", "John Kufuor",
               "John Atta Mills", "John Mahama*", "Nana Akufo-Addo")

president_party <- c("NDC", "NPP", "NPP", "NPP",
                     "NDC", "NDC", "NPP")
```

Table 2: Presidential Turnover and the Afrobarometer Surveys in Ghana

	Year of survey	President	President party
Round 1	1999	Jerry Rawlings	NDC
Round 2	2002	John Kufuor	NPP
Round 3	2005	John Kufuor	NPP
Round 4	2008	John Kufuor	NPP
Round 5	2012	John Atta Mills	NDC
Round 6	2014	John Mahama*	NDC
Round 7	2017	Nana Akufo-Addo	NPP

Note: NDC = National Democratic Congress; NPP = New Patriotic Party.*President Atta Mills passed away in July 2012. The survey was conducted in May 2012. Vice President John Mahama assumed the presidency and was the NDC candidate in the December 2012 election.

```
table <- tibble(
  ` ` = year_of_survey,
  `Year of survey` = year,
  `President` = president,
  `President party` = president_party) %>%
  tinytable::tt(notes = "Note: NDC =  National Democratic Congress; NPP = New Patriotic Party
table
```

Week 2

Loading necessary packages.

```
library(lubridate)
library(janitor)
library(knitr)
library(kableExtra)
library(scales)
library(dplyr)
library(haven)
library(stringr)
library(tidyverse)
```

Task 1

```
load("grand_chamber_texts.RData")
```

Task 2

```
grand_chamber_texts <- grand_chamber_texts %>%
  mutate(itemid = sub("http://hudoc.echr.coe.int/eng\\?i=", "", url))
```

Task 3

```
load("case_details.RData")
```

Task 4

```
case_details <- janitor::clean_names(case_details)
names(case_details)
```

```
[1] "itemid"             "document_type"      "respondent_state_s"
[4] "judgment_date"      "conclusion_s"
```

Task 5

```
case_details <- case_details %>%
  mutate_all(~ stringr::str_squish(str_trim(.)))
```

Task 6

To verify that the merged data has the expected number of rows, I first checked the number of rows in both case_details and grand_chamber_texts. I then performed the inner join on the itemid column since it uniquely identifies each judgment. Then I checked the number of rows in the resulting merged_dataset. The number of rows in should be less than or equal to the number of rows in both of the original datasets, because an inner join only keeps records with matching item IDs. By printing out the number of rows, I observed that merged_dataset has 511 rows, the same as grand_chamber_texts, and less than case-details, indicating that only the common rows that exist in both datasets were kept,.

```
merged_dataset <- inner_join(case_details, grand_chamber_texts, by = "itemid")

nrow(merged_dataset)
```

```
[1] 511
```

```
nrow(case_details)
```

```
[1] 27246
```

```
nrow(grand_chamber_texts)
```

```
[1] 511
```

Task 7

The bar chart in Figure Figure 2 shows how many times each country has been a respondent in a Grand Chamber case. The data reveals significant variation with some countries appearing more frequently than others in important cases. This visualization provides a clear and informative overview of the distribution of cases by respondent state.

```
country_counts <- merged_dataset %>%
  separate_rows(respondent_state_s, sep = ",") %>%
  group_by(respondent_state_s) %>%
  summarise(value = n()) %>%
  arrange(desc(value)) %>%
  mutate(value = if_else(str_detect(respondent_state_s,
"Cyprus|Türkiye|Moldova|Russia|Bosnia and Herzegovina|Croatia|North Macedonia|Serbia|Sloveni
```

```
              value + 1, value))

country_counts <- country_counts[-c(34,39,40,41),]

ggplot(country_counts, aes(x = reorder(respondent_state_s, -value), y = value)) +
  geom_bar(stat = "identity", fill = "#0073C2FF", color = "white") +
  labs(x = "Respondent State",
       y = "Number of Cases",
       caption = "Data from the Grand Chamber judgments from the European Court of Human Rigl
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        panel.grid.minor = element_blank())
```

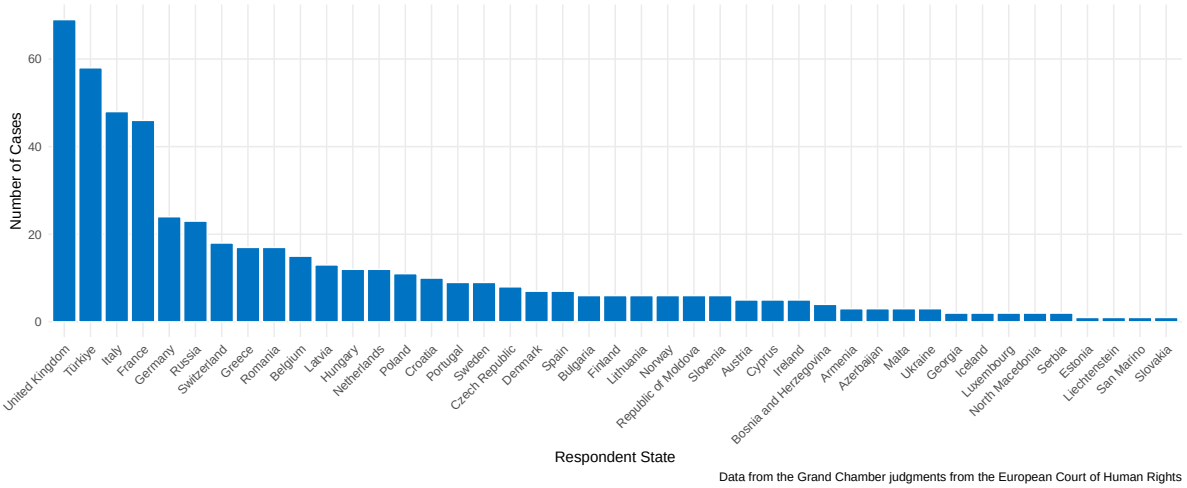

Figure 2: Count of Grand Chamber Cases by Respondent State.

Task 8

I used the package called lubridate.

```
merged_dataset$judgment_date <- as.character(merged_dataset$judgment_date)
merged_dataset$date <- dmy(merged_dataset$judgment_date)
merged_dataset$judgment_year <- as.character(year(merged_dataset$date))
```

Task 9

```
merged_dataset <- merged_dataset %>%
  mutate(margin_of_appreciation_count =
           str_count(judgment_text, fixed("margin of appreciation")),
         judgment_year = as.numeric(judgment_year))


ggplot(data = merged_dataset,
       aes(x = judgment_year, y = margin_of_appreciation_count)) +
  geom_point(color = "#0073C2FF", size = 2) +
  geom_vline(xintercept = 2012, color = "black")+
  geom_smooth(method = "loess",
              linetype = "dashed",
              size = 2,
              color = "black")+
  labs(title = "References to the 'Margin of Appreciation' Doctrine in Grand Chamber Judgment
       x = "\nJudgment Year",
       y = "Count of 'Margin of Appreciation' References") +
  theme_minimal()
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```

```
`geom_smooth()` using formula = 'y ~ x'
```

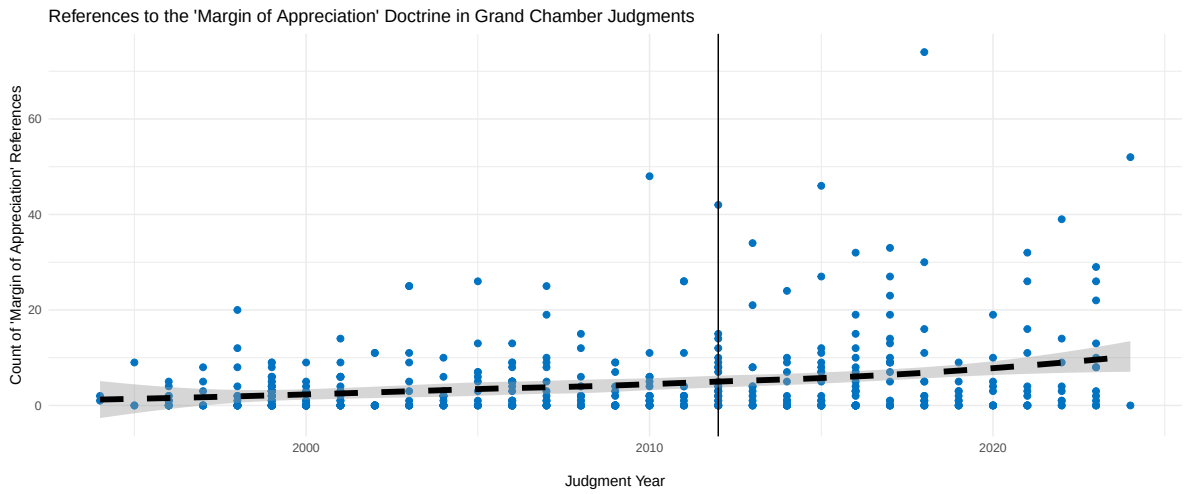

Figure 3: Figure: Trends in mentions of the 'margin of appreciation' over time, highlighting the year 2012.

Figure 3 counts the number of times "margin of appreciation" is mentioned in each judgment_text. If the European Court has become more deferential towards (some) member states over the last decade or so, I expect this number to go up. This is excactly what I see, but the increase is not visible until the late 2010s.

Task 10

Based on the feedback, I slightly modified the code on line so that the start_marker string, is just "composed". This seemeed to do the trick, and the answer in task 11 should now be correct.

```
extract_judges <- function(text) {
  start_marker <- "composed"
  end_marker <- "deliberated|the following judgment"
  pattern <- paste0("(?s)", start_marker, "(.*?)", end_marker)
  judges_section <- str_extract(text, pattern)
  if (is.na(judges_section)) return(NA)
  judges_section <- str_remove(judges_section, start_marker)
  judges_section <- str_remove(judges_section, end_marker)
  judges <- str_split(judges_section, "[,\\n]+")[[1]]
  judges <- str_trim(judges)
  judges <- judges[
    judges != "" & !grepl("registrar|president|President|ad hoc judge|Having|judges|juriscons
  return(judges)}
```

```r
judge_list <- merged_dataset %>%
  rowwise() %>%
  mutate(judges = list(extract_judges(judgment_text))) %>%
  select(itemid, judgment_year, judges) %>%
  unnest(judges) %>%
  filter(!grepl("and ", judges))

judge_summary <- judge_list %>%
  group_by(itemid, judgment_year) %>%
  summarise(num_judges = n())
```

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

```r
print(judge_summary, n = 10)
```

```
# A tibble: 508 x 3
# Groups:   itemid [508]
   itemid     judgment_year num_judges
   <chr>              <dbl>      <int>
 1 001-100413          2010         17
 2 001-100448          2010         17
 3 001-100686          2010         17
 4 001-101568          2010         17
 5 001-101579          2010         17
 6 001-101739          2010         17
 7 001-101740          2010         17
 8 001-102332          2010         17
 9 001-102617          2011         17
10 001-103050          2011         17
# i 498 more rows
```

Task 11

```r
table_data <- judge_list %>%
  group_by(judges) %>%
  summarise(count = n()) %>%
  drop_na() %>%
```

Table 3: Top 10 Judges by Number of Cases in the The European Court of Human Rights

Judge	Number of cases
Mr L. Wildhaber	166
Mr J.-P. Costa	151
Josep Casadevall	135
Mrs E. Palm	123
Guido Raimondi	115
Nicolas Bratza	115
Dean Spielmann	110
Jean-Paul Costa	104
Angelika Nußberger	97
Mr M. Fischbach	97

```
  arrange(desc(count)) %>%
  head(10)

table <- tibble(
  `Judge` = table_data$judges,
  `Number of cases` = table_data$count)

table %>%
  tinytable::tt()
```

Task 12

```
task_12_data <- case_details %>%
  select(itemid, conclusion_s) %>%
  unique()

extract_first_number <- function(text) {
  numbers <- str_extract_all(text, "\\d+")
  flat_numbers <- unlist(numbers)
  first_number <- if (length(flat_numbers) > 0) as.numeric(flat_numbers[1]) else NA
  return(first_number)}
```

```r
task_12_data$article <- sapply(task_12_data$conclusion_s, extract_first_number)

task_12_data <- task_12_data %>%
  filter(!str_detect(conclusion_s, "^Violation of P") &
           !str_detect(conclusion_s, "^No violation of P")) %>%
  mutate(violated = str_starts(conclusion_s, "Violation"),
         not_violated = str_starts(conclusion_s, "No violation")) %>%
  filter(violated == TRUE | not_violated == TRUE)

table_data <- task_12_data %>%
  group_by(article) %>%
  filter(violated == FALSE) %>%
  summarise(count = n()) %>%
  drop_na() %>%
  arrange(desc(count)) %>%
  head(10)

table <- tibble(
  `Article` = table_data$article,
  `Number of non-violations` = table_data$count) %>%
  tinytable::tt()

table_data2 <- task_12_data %>%
  group_by(article) %>%
  filter(violated == TRUE) %>%
  summarise(count = n()) %>%
  drop_na() %>%
  arrange(desc(count)) %>%
  head(10)

table2 <- tibble(
  `Article` = table_data2$article,
  `Number of violations` = table_data2$count) %>%
  tinytable::tt()
```

```r
table
table2
```

Article	Number of non-violations
6	697
3	373
8	315
2	213
5	213
10	153
1	72
14	43
11	34
7	26

Article	Number of violations
6	9692
3	2588
5	1960
8	1112
1	933
10	816
2	753
11	431
14	163
13	81

Top 10 most and least violated articles in The European Court of Human RightsS

Week 3

Loading necessary packages.

```
library(lubridate)
library(zoo)
library(rvest)
library(knitr)
library(kableExtra)
library(scales)
library(dplyr)
library(haven)
library(stringr)
library(tidyverse)
```

Task 1

Save a local copy of the website https://www.pollofpolls.no/?cmd=Stortinget&do=visallesnittLenker til en ekstern side. as an `.html` file. Change the execution options for the code chunk that downloads the file so that the file is not downloaded again every time you render your document. This website shows average support for each major political party in Norway based on public opinion polls for each month from 2008.

```
url <- "https://www.pollofpolls.no/?cmd=Stortinget&do=visallesnitt"
download.file(url, destfile = "poll_of_polls.html")
```

Task 2

```
html <- read_html("poll_of_polls.html")

table <- html %>%
  html_element("table") %>%
  html_table()

monthly_averages <- as.data.frame(table)
names(monthly_averages)=make.names(names(monthly_averages))
head(monthly_averages)
```

```
           X        Ap    HÃ.yre       Frp        SV       Sp     KrF  Venstre
1 August '24 20,7 (41) 25,2 (46) 16,4 (30)  9,1 (17) 6,2 (11) 3,9 (3)  5,2 (9)
2   Juli '24 21,1 (41) 24,9 (44) 16,5 (33)  9,5 (17)  4,7 (8) 3,9 (3) 5,6 (10)
3   Juni '24 19,7 (36) 24,2 (43) 17,2 (35)  9,7 (17) 6,3 (11) 3,5 (2) 5,9 (11)
4    Mai '24 19,9 (40) 25,2 (44) 17,1 (32)  9,3 (16) 6,1 (10) 4,1 (7) 5,9 (10)
5  April '24 19,5 (39) 26,4 (48) 15,2 (28) 10,0 (18) 6,2 (11) 3,8 (3) 5,5 (10)
6   Mars '24 18,2 (35) 26,5 (49) 14,8 (26)  9,9 (17) 6,5 (12) 3,8 (3) 5,9 (10)
      MDG    RÃ.dt   Andre
1 3,6 (2) 5,7 (10) 4,2 (0)
2 3,7 (3) 5,4 (10) 4,7 (0)
3 3,4 (2) 6,6 (12) 3,5 (0)
4 3,7 (2)  5,0 (8) 3,8 (0)
5 3,6 (2) 5,5 (10) 4,2 (0)
6 4,0 (7) 5,8 (10) 4,6 (0)
```

Task 3

```
data <- monthly_averages %>%
  pivot_longer(cols = c(Ap:Andre),
               names_to = "party",
               values_to = "value") %>%
  mutate(party = recode(party, "HÃ.yre" = "Høyre", "RÃ.dt" = "Rødt")) %>%
  rename(date = X)
```

Task 4

```
data <- data %>%
  separate(value, into = c("support", "seats"),
           sep = " \\(", convert = TRUE) %>%
  mutate(seats = as.numeric(gsub("\\)", "", seats)),
         support = as.numeric(gsub(",", ".", support)))
```

Task 5

```
data <- data %>%
  mutate(date = gsub("Januar", "January", date),
         date = gsub("Februar", "January", date),
         date = gsub("Mars", "March", date),
         date = gsub("April", "April", date),
         date = gsub("Mai", "May", date),
         date = gsub("Juni", "June", date),
         date = gsub("Juli", "July", date),
         date = gsub("August", "August", date),
         date = gsub("September", "September", date),
         date = gsub("Oktober", "October", date),
         date = gsub("November", "November", date),
         date = gsub("Desember", "December", date)) %>%
  mutate(date = as.yearmon(date, "%B '%y"))
```

Task 6

```
party_colors <- c(
  "Høyre" = "#1E90FF",
  "Ap" = "#FF0000",
  "Sp" = "#006400",
  "Frp" = "#87CEEB",
  "SV" = "#FF69B4",
  "Venstre" = "#32CD32",
  "KrF" = "#FFD700",
  "MDG" = "#2E8B57",
  "Andre" = "#808080")
```

```r
ggplot(data, aes(x = date, y = support, color = party)) +
  geom_line(size = 1) +
  scale_color_manual(values = party_colors) +
  labs(title = "Support for Norwegian political parties over time",
    x = "Date",
    y = "Support (%)\n",
    caption = "Source: pollofpolls.no",
    color = "Party") +
  theme_minimal() +
  theme(
    panel.grid.minor = element_blank(),
    legend.position = "top",
    plot.title = element_text(hjust = 0.5, size = 24),
    axis.title = element_text(size = 16),
    axis.text = element_text(size = 14),
    legend.title = element_text(size = 14),
    legend.text = element_text(size = 12))
```

```
Warning: The `trans` argument of `continuous_scale()` is deprecated as of ggplot2 3.5.0.
i Please use the `transform` argument instead.
```

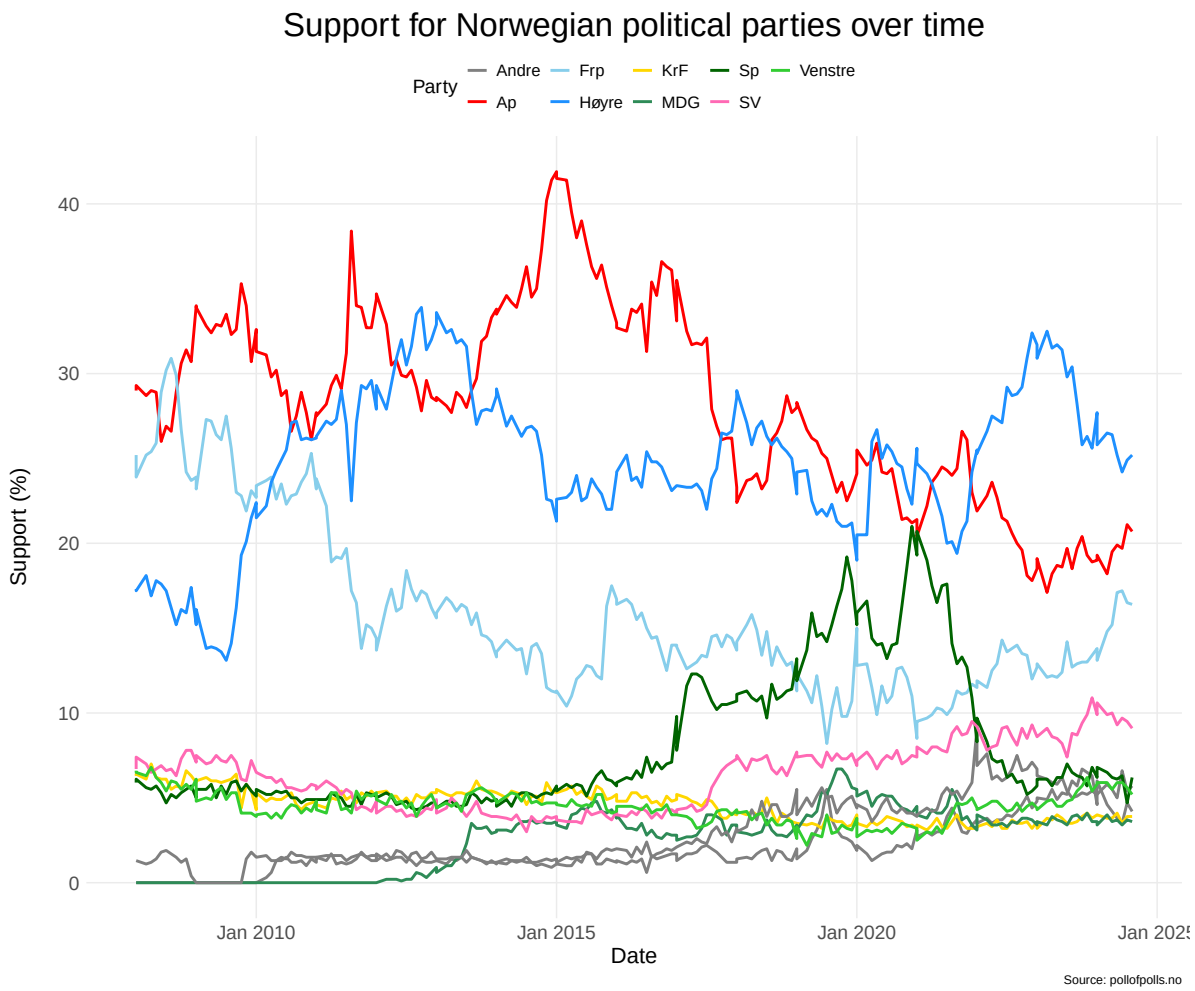

Figure 4: Trends in support for political parties in Norway.

Task 7

```
urls <- html %>%
  html_element("table") %>%
  html_elements("a") %>%
  html_attr("href")

month_links <- grep("\\?cmd=Stortinget&do=snitt&yw=\\d+", urls, value = TRUE)
```

Task 8

```
base_url <- "https://www.pollofpolls.no/"
complete_urls <- paste0(base_url, month_links)

if(length(complete_urls) > 0) {browseURL(complete_urls[1]) }
```

Task 9

```
start_date <- ymd("2024-08-01")
end_date <- ymd("2008-01-01")
date_sequence <- seq(start_date, end_date, by = "-1 month")

months <- format(date_sequence, "%B_%Y")

if (!dir.exists("monthly")) { dir.create("monthly") }

download_pages <- function(complete_urls, output_folder, months) {
  for (i in seq_along(complete_urls)) {
    output_file <- file.path(output_folder, paste0("month_", months[i], ".html"))
    if (!file.exists(output_file)) {
      download.file(complete_urls[i], output_file, quiet = TRUE)
      Sys.sleep(1)}
    }
  }

if (!file.exists(file.path("monthly", "download_complete.txt"))) {
  download_pages(complete_urls, "monthly", months)
  file.create(file.path("monthly", "download_complete.txt"))
  }
```

Task 10

```
extract_polls_table <- function(file_path) {
  html <- read_html(file_path)
  table <- html %>% html_element("table") %>% html_table()
  table$file_name <- basename(file_path)
```

```
  return(table)
}

files <- list.files(path = "monthly", pattern = "*.html", full.names = TRUE)

all_polls <- files %>%
  lapply(extract_polls_table) %>%
  bind_rows()

writexl::write_xlsx(all_polls,"all_polls.xlsx")
```

Task 11

```
all_polls <- read_excel("all_polls.xlsx")
```

```
Error in read_excel("all_polls.xlsx"): could not find function "read_excel"
```

```
final_polls <- all_polls %>%
  rename_with(~c("poll","Ap","Høyre","Frp","SV","Sp",
                 "KrF","Venstre","MDG","Rødt","Andre","month")) %>%
  mutate(month_year = str_remove_all(month, "month_|\\.html"),
    yearmon = as.yearmon(month_year, "%B_%Y")) %>%
  select(-month, -month_year)
```

```
Error: object 'all_polls' not found
```

```
avarage_polls <- final_polls %>%
  filter((str_detect(poll, "Gjennomsnitt|Feilmargin"))) %>%
  pivot_longer(
    cols = c(2:11),
    names_to = c("party"),
    values_to = "value") %>%
  pivot_wider(names_from = poll,
              values_from = value) %>%
  rename(error = Feilmargin,
         avarage = Gjennomsnitt)
```

```
Error: object 'final_polls' not found
```

```r
final_polls <- final_polls %>%
  filter(!(str_detect(poll, "Gjennomsnitt|Feilmargin"))) %>%
    pivot_longer(
    cols = c(2:11),
    names_to = c("party"),
    values_to = "value") %>%
  left_join(avarage_polls, by = c("party", "yearmon")) %>%
  separate(value, into = c("support", "seats"),
           sep = " \\(", convert = TRUE) %>%
  separate(avarage, into = c("support_avarage", "seats_avarage"),
           sep = " \\(", convert = TRUE) %>%
  mutate(support = as.numeric(gsub(",", ".", support)),
         support_avarage = as.numeric(gsub(",", ".", support_avarage)),
         error = gsub("±", "", error),
         error = as.numeric(gsub(",", ".", error)),) %>%
  select(-c(seats, seats_avarage))
```

```
Error: object 'final_polls' not found
```

Task 12

```r
labor_polls <- final_polls %>%
  filter(party == "Ap")
```

```
Error: object 'final_polls' not found
```

```r
ggplot(labor_polls, aes(x = yearmon)) +
  geom_point(aes(y = support), color = "#FF0000", alpha = 0.3) +
  geom_line(aes(y = support_avarage), color = "#FF0000", size = 1.2) +
  geom_ribbon(aes(ymin = support_avarage - error, ymax = support_avarage + error),
              fill = "#FF0000", alpha = 0.2) +
  labs(title = "Labor Party Support Over Time",
       x = "Month-Year",
       y = "Percentage Support",
       caption = "Source: Your Data Source") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold", hjust = 0.5),
        axis.title = element_text(size = 14),
        axis.text = element_text(size = 12),
        plot.caption = element_text(size = 10, hjust = 1))
```

```
Error: object 'labor_polls' not found
```

Week 4

Loading necessary packages.

```
library(rjson)
library(jsonlite)
library(readxl)
library(rvest)
library(knitr)
library(kableExtra)
library(scales)
library(dplyr)
library(haven)
library(stringr)
library(tidyverse)
```

Task 1

```
if (!requireNamespace("PxWebApiData", quietly = TRUE)) {
  install.packages("PxWebApiData")}

library(PxWebApiData)
```

Task 2

The number of the table about "Election campaign contributions" to different Norwegian political parties is 10198. The election years 2013, 2015, 2017, 2019, 2021 and 2023 are included. For more information, please go to Statistics Norway.[1]

Task 3

```
ApiData("https://data.ssb.no/api/v0/en/table/10198",  returnMetaFrames = FALSE)
```

[1] https://www.ssb.no/en/statbank/table/10198

```
Error in ApiData("https://data.ssb.no/api/v0/en/table/10198", returnMetaFrames = FALSE): cou
```

```
party_contributions <- ApiData12(
  "https://data.ssb.no/api/v0/en/table/10198",
  PolitPartiLag = c(
    "9010000", "9020000", "9030000","9040000", "9050000",
    "9060000", "9070000", "9080000", "9130000"),
  ContentsCode = c(
    "BidragIalt","Privatpersoner","OrganisasjonArbLiv",
    "BidragKommersielle","BidragAndre"),
  Tid = c("2013","2015","2017","2019","2021","2023")) %>%
  select(cols = c(1:4,7)) %>%
  mutate(cols1 = recode(
    cols1,
    "Christian Democratic party, total" = "KrF",
    "Liberal Party, total" = "Venstre",
    "Socialist Left Party, total" = "SV",
    "Labour Party, total" = "Ap",
    "Progress Party, total" = "Frp",
    "Conservative Party, total" = "Høyre",
    "Centre Party, total" = "Sp",
    "Red Party, total" = "Rødt",
    "Green Party, total" = "MDG"))
```

```
Error in ApiData12("https://data.ssb.no/api/v0/en/table/10198", PolitPartiLag = c("9010000",
```

```
colnames(party_contributions) <- c(
  "party","contribution_type","year", "party_code","value")
```

```
Error: object 'party_contributions' not found
```

```
writexl::write_xlsx(party_contributions,
                    "party_contributions.xlsx")
```

```
Error: object 'party_contributions' not found
```

Task 4

```r
party_contributions <- read_excel("party_contributions.xlsx") %>%
  mutate(year = as.numeric(year))
```

Task 5

Figure 5 shows donations to the Norwegian political parties over time.

```r
party_colors <- c(
  "Høyre" = "#1E90FF",
  "Ap" = "#FF0000",
  "Sp" = "#006400",
  "Frp" = "#87CEEB",
  "SV" = "#FF69B4",
  "Venstre" = "#32CD32",
  "KrF" = "#FFD700",
  "MDG" = "#2E8B57")

task4_data <- party_contributions %>%
  filter(contribution_type == "Total contributions")

ggplot(
  data = task4_data,
  aes(x = year, y = value, color = party)) +
  geom_line(size = 1) +
  scale_x_continuous(limits = c(2013, 2023),
                     breaks = seq(2013, 2023, by = 2))+
  scale_color_manual(values = party_colors,
                     na.value = "gray40",
                     labs(label = "Party"))+
  scale_y_continuous(labels = scales::number_format(suffix = " kr"))+
  labs(title = "Donations to Norwegian political parties",
       x = "Election year",
       y = "Total donations\n",
       color = "Party") +
  theme_minimal() +
  theme(legend.position = "right",
        strip.text = element_text(size = 14),
        axis.title = element_text(size = 25),
        axis.text.x = element_text(size = 12),
        axis.text.y = element_text(size = 16),
        legend.title = element_text(size = 18),
```

```
        legend.text = element_text(size = 15),
        legend.key.height = unit(1, 'cm'),
        panel.grid.minor = element_blank(),
        plot.title = element_text(hjust = 0.5,
                                  size = 25,
                                  margin=margin(20,0,20,0)),
        plot.subtitle = element_text(hjust = 0.5,
                                     size = 20,
                                     margin=margin(10,0,10,0)),
        plot.caption = element_text(hjust = 1))
```

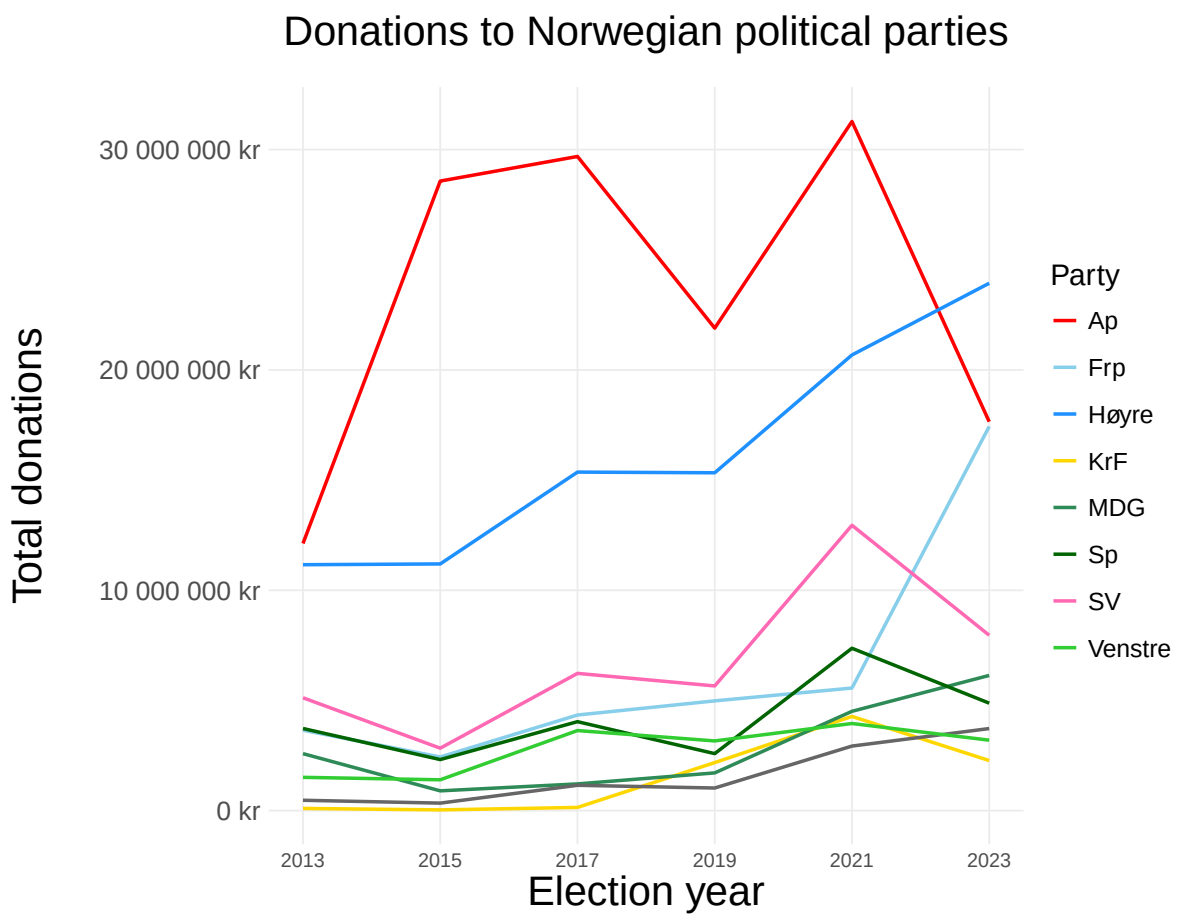

Figure 5: Trends in donations to Norwegian political parties

Task 6

Figure 6 shows donations to the Norwegian political blocks over time.

```
party_colors <- c(
  "Right-wing block" = "#1E90FF",
  "Left-wing block" = "#FF0000")

total_contributions_blocks <- party_contributions %>%
  filter(contribution_type == "Total contributions") %>%
  mutate(block = case_when(
    party %in% c("KrF", "Venstre","Høyre","Frp") ~ "Right-wing block",
    party %in% c("Rødt","SV","Ap","Sp") ~ "Left-wing block")) %>%
    group_by(block, year) %>%
    summarise(value = sum(value, na.rm = TRUE)) %>%
  drop_na()
```

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

```
print(total_contributions_blocks)
```

```
# A tibble: 12 x 3
# Groups:   block [2]
   block             year    value
   <chr>            <dbl>    <dbl>
 1 Left-wing block   2013 21444541
 2 Left-wing block   2015 34065114
 3 Left-wing block   2017 41105149
 4 Left-wing block   2019 31170075
 5 Left-wing block   2021 54526795
 6 Left-wing block   2023 34212032
 7 Right-wing block  2013 16421400
 8 Right-wing block  2015 15064616
 9 Right-wing block  2017 23482021
10 Right-wing block  2019 25656281
11 Right-wing block  2021 34473198
12 Right-wing block  2023 46841884
```

```r
types_contributions_blocks <- party_contributions %>%
  mutate(contribution_type = str_replace(contribution_type, "Contributions from (.)", functi
      paste0(toupper(substr(match, 20, 20)), substr(match, 21, nchar(match)))}),
    block = case_when(
      party %in% c("KrF", "Venstre", "Høyre", "Frp") ~ "Right-wing block",
      party %in% c("Rødt", "SV", "Ap", "Sp") ~ "Left-wing block")) %>%
  group_by(block, year, contribution_type) %>%
  summarise(value = sum(value, na.rm = TRUE), .groups = 'drop') %>%
  drop_na()

plot <- ggplot(
  data = types_contributions_blocks,
  aes(x = year, y = value, color = block)) +
  geom_line(size = 1) +
  scale_x_continuous(limits = c(2013, 2023),
                     breaks = seq(2013, 2023, by = 2))+
  scale_color_manual(values = party_colors,
                     na.value = "gray40",
                     labs(label = "Party"))+
  scale_y_continuous(labels = scales::number_format(suffix = " kr"))+
  facet_wrap(~contribution_type, scales = "free",  ncol = 2)+
  labs(title = "Donations to the political blocks from donors",
       x = "\nElection year",
       y = "Total donations\n",
       color = "Party") +
  theme_minimal() +
  theme(legend.position = "top",
        strip.text = element_text(size = 14),
        axis.title = element_text(size = 25),
        axis.text.x = element_text(size = 12),
        axis.text.y = element_text(size = 16),
        legend.title = element_text(size = 18),
        legend.text = element_text(size = 15),
        legend.key.height = unit(1, 'cm'),
        panel.grid.minor = element_blank(),
        plot.title = element_text(hjust = 0.5,
                                  size = 25,
                                  margin=margin(20,0,20,0)),
        plot.subtitle = element_text(hjust = 0.5,
                                     size = 20,
                                     margin=margin(10,0,10,0)),
        plot.caption = element_text(hjust = 1))
```

```
plot
```

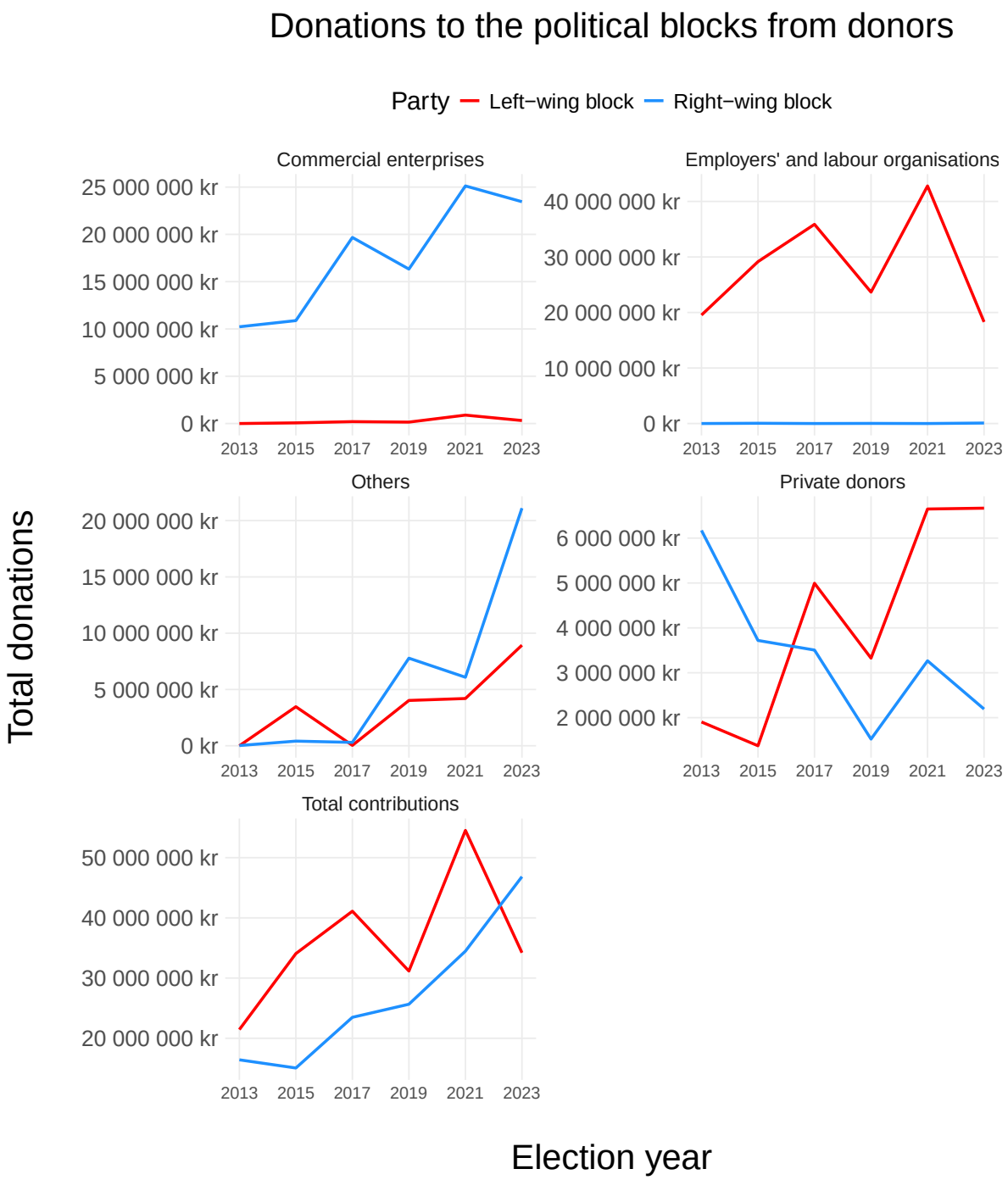

Figure 6: Trends in donations to Norwegian political blocks

Task 7

```
subjects <- read_json("https://data.uio.no/studies/v1/courses")
subjects <- toJSON(subjects, pretty = TRUE)
write(subjects, file = "subjects_data.json")
```

Task 8

```
json_data <- as.data.frame(fromJSON("subjects_data.json")) %>%
  select(courses.code, courses.name, courses.url)
```

Task 9

```
task_9_data <- json_data %>%
  filter(grepl("STV", courses.code)) %>%
  mutate(level = case_when(
    substr(courses.code, 4, 4) == "1" ~ "Introductory BA",
    substr(courses.code, 4, 4) %in% c("2", "3") ~ "Advanced BA",
    substr(courses.code, 4, 4) == "4" ~ "MA",
    substr(courses.code, 4, 4) == "9" ~ "PhD"),
  subfield = case_when(
    substr(courses.code, 5, 5) == "0" ~ "Methods",
    substr(courses.code, 5, 5) == "1" ~ "Political theory",
    substr(courses.code, 5, 5) == "2" ~ "International relations",
    substr(courses.code, 5, 5) == "3" ~ "Comparative politics",
    substr(courses.code, 5, 5) %in% c("4", "8") ~ "Public administration",
    substr(courses.code, 5, 5) == "5" ~ "Cross-cutting",
    substr(courses.code, 5, 5) %in% c("6", "9") ~  "MA Thesis"))
```

Task 10

```
# Check if the directory "semesters" exists; if not, create it
if (!dir.exists("semesters")) {
  dir.create("semesters")}
```

```
# Define a function to extract the last two digits of a year
get_last_two_digits <- function(year) {return(substr(year, 3, 4))}

years <- 2004:2024 # Create a sequence of years from 2004 to 2024
semesters <- c("v", "h") # Defining semesters: 'v' for spring and 'h' for autumn

for (year in years) {
  last_two_digits <- get_last_two_digits(year) # Last two digits identifies year
  for (semester in semesters) { # Loop through semesters for the current year
    url <- paste0("https://data.uio.no/studies/v1/semester/",
                  last_two_digits, semester, "/courses")
    file_name <- file.path("semesters", # Define the file name and path
                           paste0(year, "_", semester,
                                  "_courses.json"))
    if (!file.exists(file_name)) {  # Check if the file already exists
      tryCatch({ # Download and write the JSON data to a file
        write_json(fromJSON(url), path = file_name)
        cat("Downloaded:", file_name, "\n")
        Sys.sleep(1) # Sleep for 1 second to avoid overloading the server
      }, error = function(e) { # Print an error message if the download fails
        cat("Failed to download:", url, "\n")
      })
    } else {  # Inform that the file already exists
      cat("File already exists:", file_name, "\n")
    }
  }
}
```

Task 11

```
process_json_file <- function(file) {
  semester <- sub("^.*/", "", file)
  semester <- substr(semester, 1, 6)
  data <- fromJSON(file) %>%
    as.data.frame() %>%
    select(courses.code, courses.url)
  data <- data %>%
    mutate(semester = semester)
  return(data)
  }
```

```
json_files <- list.files("semesters", pattern = "\\.json$", full.names = TRUE)
task_11_data <- do.call(rbind, lapply(json_files, process_json_file))
```

Task 12

For some reason, spring semesters in 2013, 2015 and 2016 are unavailable for download. Therefore, I have dropped these years from the figure. From what I can observe, methods classes have exploded in popularity. Comparative politics and political theory are less and less widespread at the Department of Political Science at UiO.

```
task_12_data <- task_11_data %>%
  mutate(level = case_when(
        substr(courses.code, 4, 4) == "1" ~ "Introductory BA",
        substr(courses.code, 4, 4) %in% c("2", "3") ~ "Advanced BA",
        substr(courses.code, 4, 4) == "4" ~ "MA",
        substr(courses.code, 4, 4) == "9" ~ "PhD"),
      subfield = case_when(
        substr(courses.code, 5, 5) == "0" ~ "Methods",
        substr(courses.code, 5, 5) == "1" ~ "Political theory",
        substr(courses.code, 5, 5) == "2" ~ "International relations",
        substr(courses.code, 5, 5) == "3" ~ "Comparative politics",
        substr(courses.code, 5, 5) %in% c("4", "8") ~ "Public administration",
        substr(courses.code, 5, 5) == "5" ~ "Cross-cutting",
        substr(courses.code, 5, 5) %in% c("6", "9") ~ "MA Thesis"),
      ba_or_ma = case_when(
        substr(courses.code, 4, 4) %in% c("1", "2", "3") ~ "BA",
        substr(courses.code, 4, 4) == "4" ~ "MA"),
      semester = semester,
      year = as.numeric(substring(semester, 1, 4))) %>%
  drop_na()

course_counts <- task_12_data %>%
  group_by(year, ba_or_ma, subfield) %>%
  summarise(course_count = n(), .groups = "drop") %>%
  filter(!year %in% c(2013,2015,2016))

ggplot(course_counts,
       aes(x = year, y = course_count,
           linetype = ba_or_ma)) +
  geom_line(size = 1.2) +
  facet_wrap(~subfield, scales = "free")+
```

```
  labs(title = "Trends in different courses over time",
    x = "Year",
    y = "Number of courses",
    color = "Subfield",
    linetype = "Course Level") +
  theme_minimal() +
  theme(legend.position = "top",
        strip.text = element_text(size = 18),
        axis.title = element_text(size = 25),
        axis.text.x = element_text(size = 12),
        axis.text.y = element_text(size = 16),
        legend.title = element_text(size = 18),
        legend.text = element_text(size = 15),
        legend.key.height = unit(1, 'cm'),
        panel.grid.minor = element_blank(),
        plot.title = element_text(hjust = 0.5,
                                  size = 25,
                                  margin=margin(20,0,20,0)),
        plot.subtitle = element_text(hjust = 0.5,
                                     size = 20,
                                     margin=margin(10,0,10,0)),
        plot.caption = element_text(hjust = 1))
```

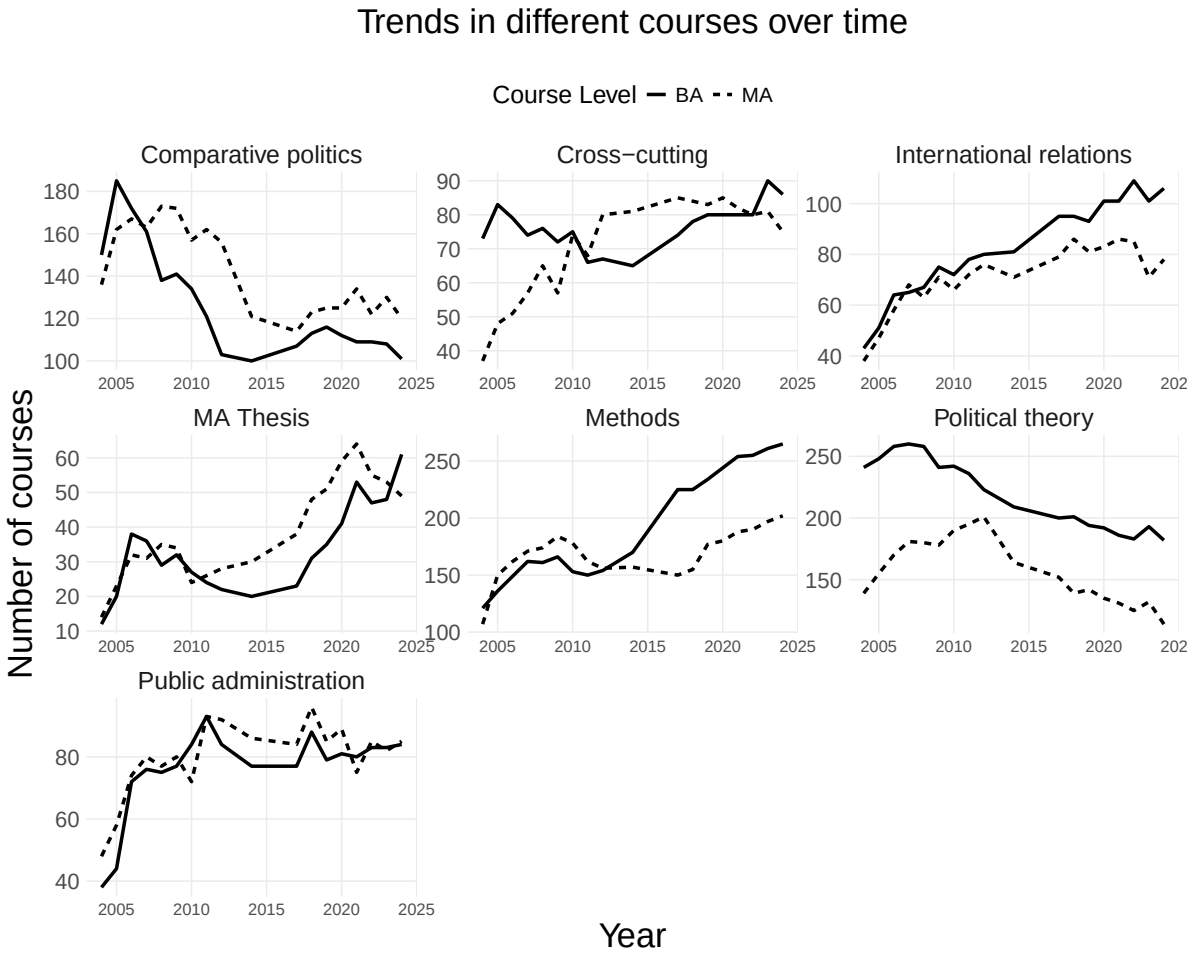

Figure 7: Trends in differenc courses offered in the Department of Political Science.

Week 5

Loading necessary packages.

```
library(sf)
library(readxl)
library(rvest)
library(knitr)
library(kableExtra)
library(scales)
library(dplyr)
```

```
library(haven)
library(stringr)
library(tidyverse)
```

Task 1

```
scad <- read_csv("SCAD2018Africa_Final.csv")
zems  <- read_dta("zems.dta")
constituencies <- st_read("constituencies2016.shp") %>%
         mutate(pr_2016 = as.factor(pr_2016))
```

```
Reading layer `constituencies2016' from data source
  `G:\My Drive\Mastergrad\STV4030A\Obligatoriske oppgaver\constituencies2016.shp'
  using driver `ESRI Shapefile'
Simple feature collection with 156 features and 3 fields
Geometry type: POLYGON
Dimension:     XY
Bounding box:  xmin: -42851.31 ymin: 8001227 xmax: 1235744 ymax: 9091128
Projected CRS: WGS 84 / UTM zone 35S
```

Task 2

```
ggplot(data = constituencies %>%
         mutate(pr_2016 = str_to_title(pr_2016))) +
     geom_sf(aes(fill = pr_2016))+
  scale_fill_discrete(name = "Provinces")+
  theme_minimal()+
  theme(legend.text = element_text(size = 16),
        legend.title = element_text(size = 20),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
```

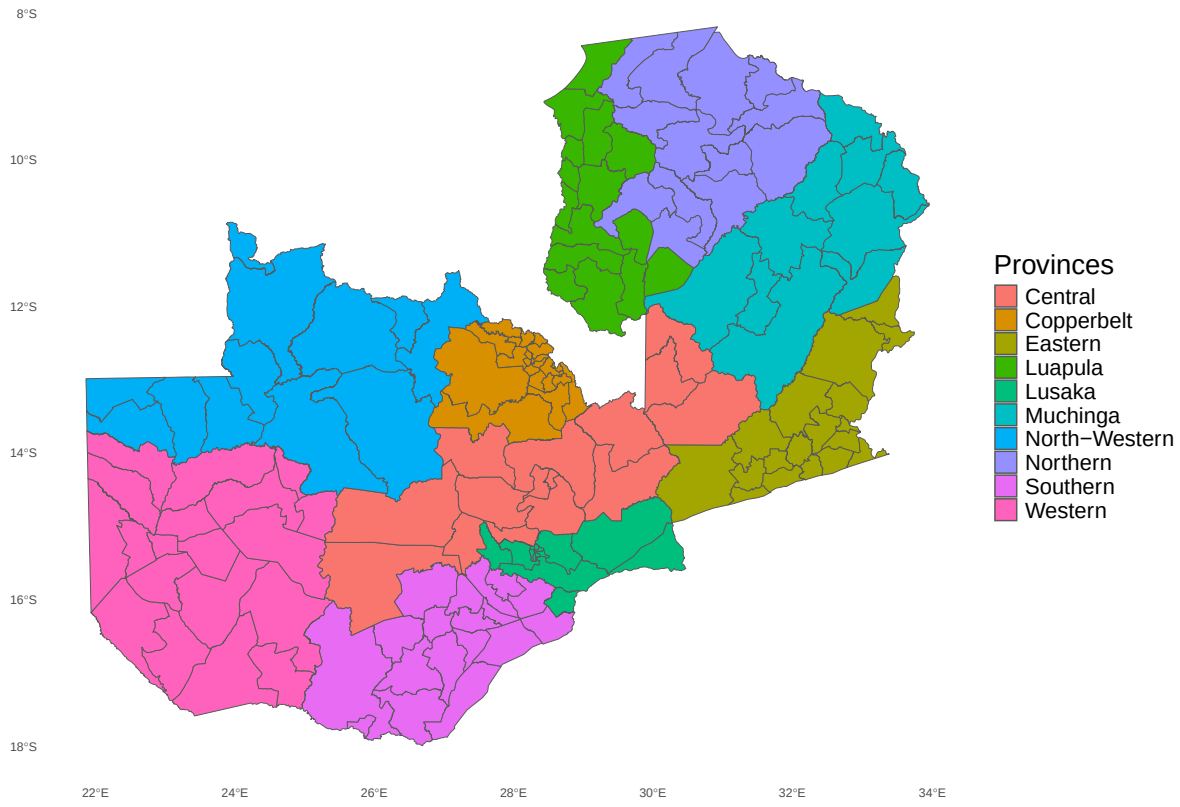

Figure 8: Constituencies and provinces in Zambia

Task 3

```
constituencies <- constituencies %>%
  mutate(cnsttnc = ifelse(cnsttnc == "mapatizya", "Zimba", cnsttnc))

constituencies$cnsttnc <- str_to_title(constituencies$cnsttnc)

unique_cnsttnc <- unique(constituencies$cnsttnc)
unique_CNAME <- unique(zems$CNAME)
mismatches <- unique_CNAME[!unique_CNAME %in% unique_cnsttnc]
print(mismatches)
```

```
[1] "Itezhitezhi"  "Mufilira"     "Mwembezhi"    "Isoka"        "Mpika"
[6] "Shiwan'gandu" "Ikeleng'i"    "Mwinilunga"
```

```
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Itezhi Tezhi"] <- "Itezhitezhi"
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Mufulira"] <- "Mufilira"
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Mwembeshi"] <- "Mwembezhi"
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Isoka West"] <- "Isoka"
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Mpika Central"] <- "Mpika"
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Shiwangandu"] <- "Shiwan'gandu"
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Ikelengi"] <- "Ikeleng'i"
constituencies$cnsttnc[constituencies$cnsttnc ==
                         "Mwinilunga East"] <- "Mwinilunga"


final_unique_cnsttnc <- unique(constituencies$cnsttnc)
final_mismatches <-
  final_unique_cnsttnc[!final_unique_cnsttnc %in% unique_CNAME]
print(final_mismatches)
```

```
character(0)
```

Task 4

I join zems to constituencies using left_join(), which is a regular join, because I have done this before, and therefore can make sure that it is done correctly.

```
zems <- zems %>%
  rename(cnsttnc = CNAME)
constituencies <- constituencies %>%
  left_join(zems, by = "cnsttnc")
```

Task 5

```
constituencies <- constituencies %>%
  mutate(electoral_violence = case_when(
    preelection_violence == 1 &
      postelection_violence == 0 ~ "Pre-election violence",
    preelection_violence == 0 &
      postelection_violence == 1 ~ "Post-election violence",
    preelection_violence == 1 &
      postelection_violence == 1 ~ "Pre- and post-election violence",
    TRUE ~ "No reported violence"))
```

Task 6

The figure tells us that violence, especially post-election violence, was mostly concentrated in the south-eastern corner of Zambia.

```
colors <- c("Pre- and post-election violence" = "#b2182b",
                "Pre-election violence" = "#fddbc7",
                "Post-election violence" = "pink",
                "No reported violence" = "white")

ggplot(data = constituencies %>%
         mutate(pr_2016 = str_to_title(pr_2016))) +
     geom_sf(aes(fill = electoral_violence))+
    scale_fill_manual(values = colors,
                      na.value = "gray40",
                     labs(label = ""))+
  theme_minimal()+
  theme(legend.text = element_text(size = 16),
        legend.title = element_text(size = 20),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
```

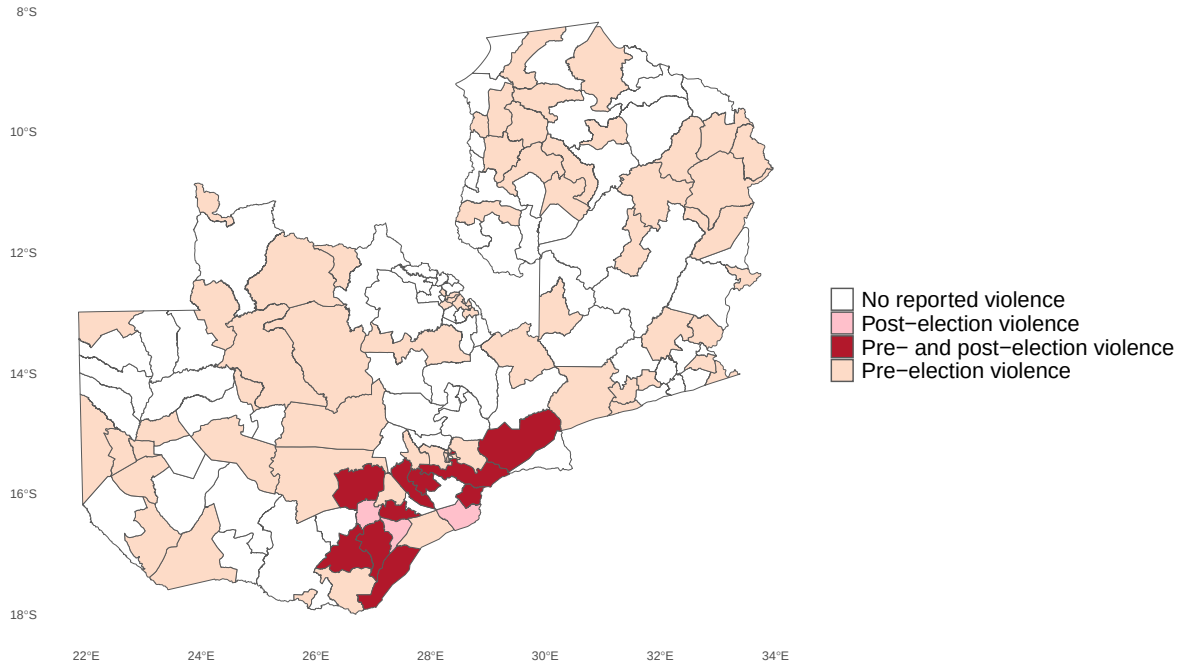

Figure 9: Electoral violence in Zambia

Task 7

```
scad_filtered <- scad %>%
  filter(countryname == "Zambia") %>%
  filter(styr == 2016) %>%
  mutate(electoral_violence_event = case_when(
    id %in% c(143,145,151) ~ 0,
    id %in% c(144,146,147,148,149,150,152) ~ 1))
```

```r
table <- tibble(
  `ID` = scad_filtered$id,
  `Date` = scad_filtered$startdate,
  `Event description` = scad_filtered$issuenote)

table %>%
  tinytable::tt(width = 2)
```

Task 8

```r
scad_filtered <- scad_filtered %>%
  st_as_sf(coords = c("longitude", "latitude"),
           crs = 4326)
```

Task 9

```r
colors <- c("Pre- and post-election violence" = "#b2182b",
                "Pre-election violence" = "#fddbc7",
                "Post-election violence" = "pink",
                "No reported violence" = "white")

ggplot(data = constituencies %>%
         mutate(pr_2016 = str_to_title(pr_2016))) +
  geom_sf(aes(fill = electoral_violence))+
  geom_sf(data = st_jitter(scad_filtered %>%
                             filter(electoral_violence_event == 1),
                           factor = 0.04), alpha = 0.5, size = 7) +
  scale_fill_manual(values = colors,
                    na.value = "gray40",
                   labs(label = ""))+
  theme_minimal()+
  theme(legend.text = element_text(size = 16),
        legend.title = element_text(size = 20),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
```

Table 4: Violence events in Zambia during 2016

ID	Date	Event description
143	2-Feb-16	Students at two universities rioted over non-payment of food and book allowances, causing the government to shut down the universities and police to disperse the rioters
143	2-Feb-16	Students at two universities rioted over non-payment of food and book allowances, causing the government to shut down the universities and police to disperse the rioters
144	2-Mar-16	Police arrested an opposition leader and 21 of his supporters
145	18-Apr-16	Rioters attacked foreign-owned shops and burned two people alive before being dispersed by police
146	16-May-16	Members of the ruling and opposition parties clashed frequently during the election campaign period, occasionally resulting in fatalities
147	8-Jul-16	Police opened fire on an opposition party campaign meeting
148	8-Aug-16	Ruling party supporters attacked bus filled with opposition party supporters
149	10-Aug-16	Unidentified assailants fought in the streets and overturned cars during an opposition leader's rally
150	15-Aug-16	Opposition supporters blocked roads with logs and burning tires following the announcement of election results before being dispersed by police
151	5-Oct-16	UPND supporters protested the arrest of two of their leaders, before being dispersed by the police
152	5-Oct-16	Two opposition leaders were

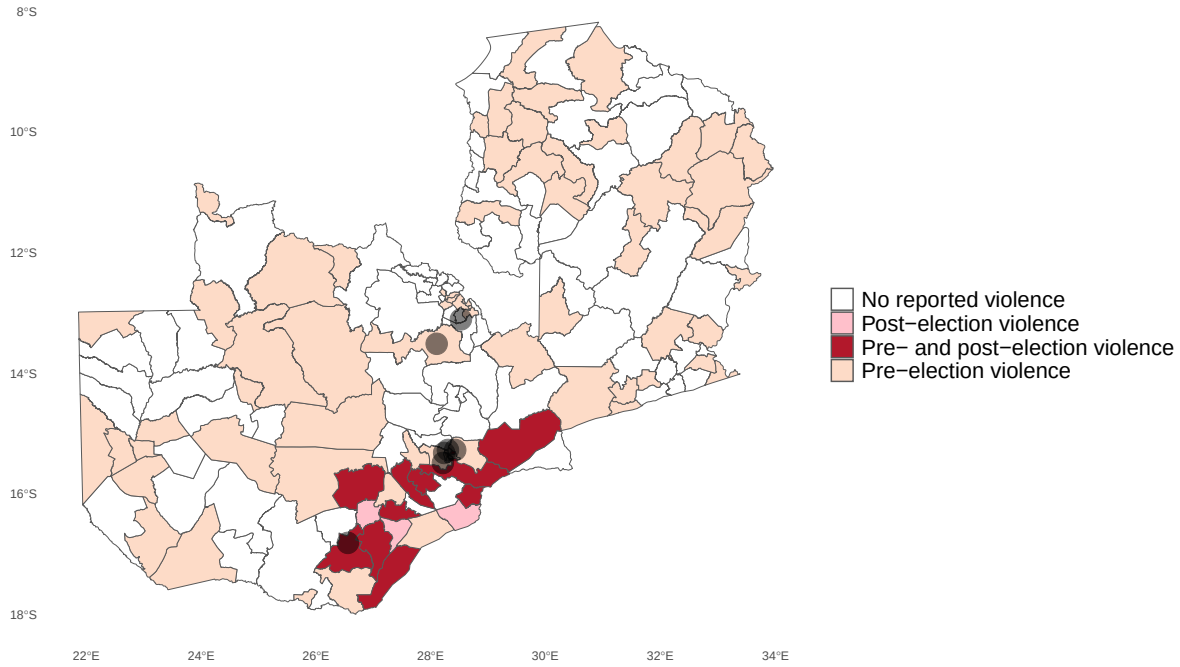

Figure 10: Electoral violence in Zambia, 2016

Task 10

```
scad_points <- scad_filtered %>%
  select(elocal) %>%
  filter(elocal != "Nationwide")

scad_points <- st_transform(scad_points, st_crs(constituencies))
joined_data <- st_join(scad_points, constituencies, join = st_within)

counts <- joined_data %>%
```

```
  group_by(pr_2016) %>%
  summarise(violence_count = n()) %>%
  as.data.frame() %>%
  mutate(pr_2016 = as.character(pr_2016)) %>%
  select(pr_2016, violence_count)

constituencies <- constituencies %>%
  left_join(counts, by = "pr_2016")

constituencies[is.na(constituencies)] <- 0
```

Task 11

The results are not consistent for the variable "Nightlight", which predicts violence outcomes using the SCAD dataset, but not using the ZEMS dataset. The results are, on the other hand, consistent for population and competition. The ZEMS violence are based on more observations of election-related-violence - 83 - compared to the the SCAD dataset's meager 54 values. Violence is measured differently in the two models, the ZEMS violence notably include a lot more incidents of pre-election violence. Overall, I do trust model 2 based on the ZEMS dataset more. It includes more violence events that occur before the election, making it more robust.

```
constituencies_task_11 <- constituencies %>%
  mutate(scad_violence = case_when(violence_count >= 1 ~ 1,
                                   violence_count == 0 ~ 0),
         zems_violence = case_when(
    electoral_violence != "No reported violence" ~ 1,
    electoral_violence == "No reported violence" ~ 0)) %>%
  as.data.frame()

constituencies_task_11 %>%
  count(zems_violence)
```

```
  zems_violence  n
1             0 73
2             1 83
```

```
constituencies_task_11 %>%
  count(scad_violence)
```

```
  scad_violence   n
1             0 102
2             1  54
```

```
model1 <- glm(
  scad_violence ~ log(PopulationDensity) +
    Nightlightlog + Competition,
  data = constituencies_task_11,
  family=binomial(link="logit"))
```

```
Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
model2 <- glm(
  zems_violence ~ log(PopulationDensity) +
    Nightlightlog + Competition,
  data = constituencies_task_11,
  family=binomial(link="logit"))

vcov_model1 <- sandwich::vcovCL(model1)
vcov_model2 <- sandwich::vcovCL(model2)

modelsummary::modelsummary(
  list("Model 1 (SCAD)" = model1,
       "Model 2 (ZEMS)" = model2),
  vcov = list(model1 = vcov_model1,
              model2 = vcov_model2),
  cluster = "pr_2016",
  gof_map = c("nobs"),
  fmt = 2)
```

I was informed in the feedback on this task, that the intercept for the scad model should be -4.789 with SE 1.102, and 0.466 with 0.339 SE for the zems model. My zems model seems to be correct. My scad model, however, does not line up with what the feedback says. I have tried my best to find the reason why this is the case, but I have come up short. I did not receive any feedback or comment on the previous steps of datawrangling. This makes me unsure of whether the discrepancy between my scad model and the feedback is due to a coding error I have failed to discover, or a different reason.

Task 12

Based on this graph, I say that the relationship between electoral competition and electoral violence in Zambia is negative. Greater electoral competition is associated with less electoral

Table 5: Results from regression analysis

	Model 1 (SCAD)	Model 2 (ZEMS)
(Intercept)	−0.84	0.47
	(0.48)	(0.33)
log(PopulationDensity)	0.59	0.16
	(0.24)	(0.13)
Nightlightlog	537.57	0.68
	(244.95)	(1.58)
Competition	−2.64	−1.28
	(1.53)	(0.73)
Num.Obs.	156	156

violence.

```
scenarios <- cbind(
  1,
  median(constituencies_task_11$PopulationDensity, na.rm = TRUE),
  median(constituencies_task_11$Nightlightlog, na.rm = TRUE),
  seq(min(constituencies_task_11$Competition, na.rm = TRUE),
      max(constituencies_task_11$Competition, na.rm = TRUE),
      length.out = 20))

sim_betas <- MASS::mvrnorm(
  n = 1000,
  mu = coefficients(model2),
  Sigma = sandwich::vcovCL(model2, cluster = constituencies_task_11$pr_2016))

sim_linear_predictors <- sim_betas %*% t(scenarios)

sim_predicted_probabilities <- pnorm(sim_linear_predictors)

plot_values <- apply(sim_predicted_probabilities,
                             MARGIN = 2,
                             FUN = quantile,
                             probs = c(.025,.5,.975))

plot_values <- plot_values%>%
  t() %>%
```

```
  as.data.frame() %>%
  mutate(Competition = as.vector(scenarios[,4]))

ggplot(plot_values,
  aes(x = Competition,
           y = `50%`,
           ymin = `2.5%`,
           ymax = `97.5%`)) +
  geom_ribbon(alpha = 0.2)+
  geom_line(linewidth = 1)+
  theme_classic()+
    labs(y = "Predicted probability of violence\n",
      x = "\nElectoral competition")
```

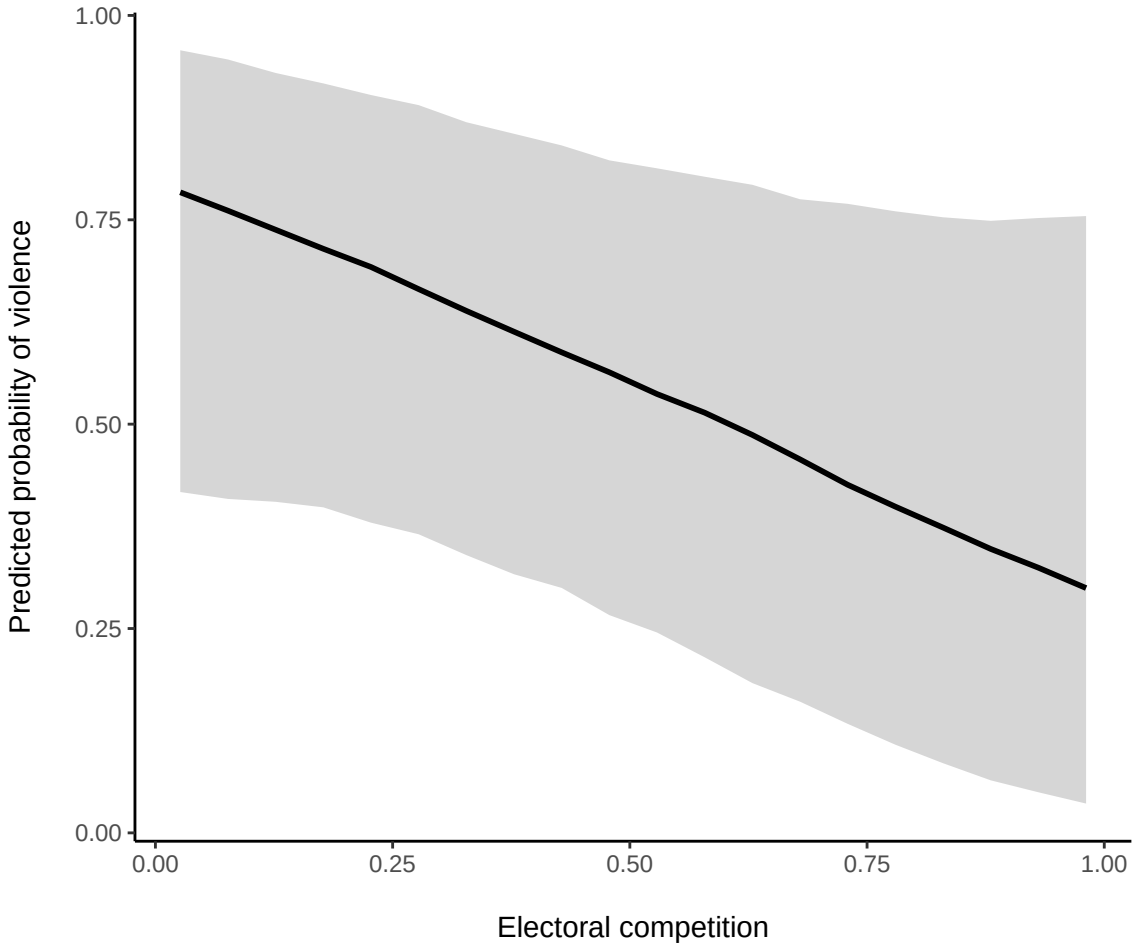

Figure 11: Violence and electoral competition in Zambia

Bartels, Brandon L, and Eric Kramon. 2020. "Does Public Support for Judicial Power Depend on Who Is in Political Power? Testing a Theory of Partisan Alignment in Africa." *American Political Science Review* 114 (1): 144–63.