

title: "Party Control of State Government" author: "Daniel Posthumus" date: 12/28/2024 output: pdf\_document — Taken from Andy Greens github

Andy Green

11/12/19

This file contains the code I used to build a dataset aimed at analyzing the relationship between party control of state government and a variety of well-being metrics at the state level. As the dataset encompasses data from a variety of different sources, this document is split up such that each of the primary metrics is given a sub-section. The source of the data, including any relevant instructions on how to query the exact data I used, are included under each sub-section heading.

## Set-up:

### Compiling the state-level party control dataset

Source: I scraped the data from each state's individual "Party Control of State Government" page on Ballotpedia. The URL for each state, as seen in the loop below, is given by: [https://ballotpedia.org/Party\\_control\\_of\\_state\\_government](https://ballotpedia.org/Party_control_of_state_government).

```
#### Looping through each of the states on Ballotpedia ####
## Create initial blank dataset to store everything in
df <- data.frame()

## Setting up the list of state names for looping. Nebraska has to be removed because they have a unique
state.name2 <- as.list(state.name)
state.name2[27] <- NULL
state.name2[28] <- "New_Hampshire"
state.name2[29] <- "New_Jersey"
state.name2[30] <- "New_Mexico"
state.name2[31] <- "New_York"
state.name2[32] <- "North_Carolina"
state.name2[33] <- "North_Dakota"
state.name2[38] <- "Rhode_Island"
state.name2[39] <- "South_Carolina"
state.name2[40] <- "South_Dakota"
state.name2[47] <- "West_Virginia"

## Creating a list to append all the individual state dataframes to
datalist = list()

## Looping through each state
for (state in state.name2)
{
  ## Concatenating the state name to the url
  url <- paste("https://ballotpedia.org/Party_control_of_", state, "_state_government", sep = "", col.

  ## Reading in the webpage
  webpage <- read_html(url)

  ## Extracting the table
  data <- html_nodes(webpage, "table.wikitable")
  data <- html_table(data)

  ## Converting to dataframe
```

```

data <- as.data.frame(data)

## Transposing rows and columns
data <- t(data)

## Fixing column names (currently stored in first row)
colnames(data) <- as.character(unlist(data[1,]))
data <- data[-1, ]

## Pulling out the year from the rownames
data2 <- as.data.frame(rownames(data))
names(data2) <- "year_end"
data <- cbind(data, data2)
data$year_end <- sub('.', '', data$year_end)

## Fixing the column name for the House, as some states call it "Assembly"
names(data)[3] <- "House"

## Filling in the state name for all rows
data$state <- state

## Converting the year values into the full year value
data$year_beg[data$year_end > 50] <- 19
data$year_beg[data$year_end < 50] <- 20
data$year <- paste(data$year_beg, data$year_end, sep = "")

## Getting rid of the underscores in the state names
data$state <- sub('_', ' ', data$state)

## Creating a variable that concatenates state and year
data$stateyear <- paste(data$state, data$year, sep = "")

## take care of potential [] coming along
## just checks if the variable is of length one and then fills it
data <- data %>%
  mutate(
    Governor = if_else(nchar(Governor) > 1, strtrim(Governor, 1), Governor),
    House = if_else(nchar(House) > 1, strtrim(House, 1), House),
    Senate = if_else(nchar(Senate) > 1, strtrim(Senate, 1), Senate)
  )

## Summarizing the total government control
data$total_gov[data$Governor == "D" & data$Senate == "D" & data$House == "D"] <- "D"
data$total_gov[data$Governor == "R" & data$Senate == "R" & data$House == "R"] <- "R"
data$total_gov[data$Governor == "D" & data$Senate == "D" & data$House == "R"] <- "Split"
data$total_gov[data$Governor == "D" & data$Senate == "R" & data$House == "D"] <- "Split"
data$total_gov[data$Governor == "D" & data$Senate == "R" & data$House == "R"] <- "Split"
data$total_gov[data$Governor == "R" & data$Senate == "D" & data$House == "R"] <- "Split"
data$total_gov[data$Governor == "R" & data$Senate == "R" & data$House == "D"] <- "Split"
data$total_gov[data$Governor == "R" & data$Senate == "D" & data$House == "D"] <- "Split"
data$total_gov[data$Governor == "I" & data$Senate == "S" & data$House == "D"] <- "Split"
data$total_gov[data$Governor == "I" & data$Senate == "R" & data$House == "D"] <- "Split"
data$total_gov[data$Governor == "I" & data$Senate == "R" & data$House == "R"] <- "Split"

```

```

data$total_gov[data$Governor == "R" & data$Senate == "R" & data$House == "S"] <- "Split"
data$total_gov[data$Governor == "R" & data$Senate == "S" & data$House == "R"] <- "Split"
data$total_gov[data$Governor == "R" & data$Senate == "D" & data$House == "S"] <- "Split"
data$total_gov[data$Governor == "I" & data$Senate == "D" & data$House == "D"] <- "Split"
data$total_gov[data$Governor == "D" & data$Senate == "S" & data$House == "D"] <- "Split"
data$total_gov[data$Governor == "D" & data$Senate == "S" & data$House == "R"] <- "Split"
data$total_gov[data$Governor == "I" & data$Senate == "D" & data$House == "R"] <- "Split"
data$total_gov[data$Governor == "D" & data$Senate == "D" & data$House == "S"] <- "Split"
data$total_gov[data$Governor == "D" & data$Senate == "R" & data$House == "S"] <- "Split"
data$total_gov[data$Governor == "R" & data$Senate == "S" & data$House == "D"] <- "Split"
## to deal with split chambers
data$total_gov[data$Senate == "S" | data$House == "S"] <- "Split"

## add existing data back to big dataframe
df <- rbind(df, data)
}
unique(df$state)

```

```

## [1] "Alabama"      "Alaska"       "Arizona"      "Arkansas"
## [5] "California"   "Colorado"     "Connecticut"  "Delaware"
## [9] "Florida"     "Georgia"      "Hawaii"       "Idaho"
## [13] "Illinois"    "Indiana"      "Iowa"         "Kansas"
## [17] "Kentucky"    "Louisiana"    "Maine"        "Maryland"
## [21] "Massachusetts" "Michigan"     "Minnesota"    "Mississippi"
## [25] "Missouri"    "Montana"      "Nevada"       "New Hampshire"
## [29] "New Jersey"  "New Mexico"   "New York"     "North Carolina"
## [33] "North Dakota" "Ohio"         "Oklahoma"     "Oregon"
## [37] "Pennsylvania" "Rhode Island" "South Carolina" "South Dakota"
## [41] "Tennessee"   "Texas"        "Utah"         "Vermont"
## [45] "Virginia"    "Washington"   "West Virginia" "Wisconsin"
## [49] "Wyoming"

```

```

## Setting the URL for Nebraska
url <- "https://ballotpedia.org/Party_control_of_Nebraska_state_government"

## Reading in the webpage
webpage <- read_html(url)

## Extracting the table
data <- html_nodes(webpage, "table.wikitable")
data <- html_table(data)

## Converting to dataframe
data <- as.data.frame(data)

## Transposing rows and columns
data <- t(data)

## Fixing column names (currently stored in first row)
colnames(data) <- as.character(unlist(data[1,]))
data <- data[-1, ]

## Converting to dataframe

```

```

data <- as.data.frame(data)

## Adding in House column (needs to match up with other states)
data$House <- "-"

## Pulling out the year from the rownames
data2 <- as.data.frame(rownames(data))
names(data2) <- "year_end"
data <- cbind(data, data2)
data$year_end <- sub('.', '', data$year_end)

## Filling in the state name for all rows
data$state <- "Nebraska"

## Converting the year values into the full year value
data$year_beg[data$year_end > 50] <- 19
data$year_beg[data$year_end < 50] <- 20
data$year <- paste(data$year_beg, data$year_end, sep = "")

## Creating a variable that concatenates state and year
data$stateyear <- paste(data$state, data$year, sep = "")

## Adding the columns for total government control and detailed government control
data$total_gov <- "-"

df <- rbind(df, data)

####

```

Creating the Nebraska dataset

Clean up df for export

```

df2 <- df %>%
  select("state", "year", "Governor", "Senate", "House", "total_gov") %>%
  rename(Year = year, State = state) %>%
  mutate(
    Year = as.numeric(Year)
  ) %>%
  filter(Year >= 2010)

## Warning: There was 1 warning in `mutate()`.
## i In argument: `Year = as.numeric(Year)`.
## Caused by warning:
## ! NAs introduced by coercion

setwd("~/election_inflation_analysis/data/clean")
write.csv(df2, "state_trifectas_ballotpedia_scrape.csv")

```