```
R version 4.0.1 (2020-06-06) -- "See Things Now"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin17.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.72 (7845) x86_64-apple-darwin17.0]

[History restored from /Users/ryanbrutger/.Rapp.history]

>
> # Replication code for Ryan Brutger and Amy Pond's PSRM manuscript (Last updated March 10, 2025)
> # This r script generates the results for the manuscript and appendix for the media analysis
>
> # Code was run on R version R version 3.6.1 (2019-07-05) -- "Action of the Toes"
> # Using MacBook Pro using macOS Ventura, with 2.8 GHz Quad-Core Intel Core i7
>
>
> ################# 1. Load libraries, load and clean data ##############
>
> #setwd("ENTER DIRECTORY") # set working directory (change to your directory)
> setwd()
Error in setwd() : argument "dir" is missing, with no default
>
> rm(list = ls(all = TRUE))
> install.packages("pacman")
trying URL 'https://ftp.osuosl.org/pub/cran/bin/macosx/contrib/4.0/pacman_0.5.1.tgz'
Content type 'application/x-gzip' length 379950 bytes (371 KB)
==================================================
downloaded 371 KB


The downloaded binary packages are in
    /var/folders/b5/py1yktzj52g1mns3yr67jpzr0000gn/T//RtmpuV0vkb/downloaded_packages
> pacman::p_load(dplyr, ggplot2, quanteda, plotly, foreign, interplot, plm, reshape2, lda,
+                countrycode, sandwich, lmtest, MASS, RColorBrewer, states, mice, VIM,
+                margins, clusterSEs, optimx, coefplot, systemfit, randomizr, estimatr, ri2,
+                stargazer, fastDummies, cowplot, egg, cobalt, plotrix, tm, stm, stringr) # load packages
>
>
> #### Coded Media Data ####
> AT_med <- read.csv("Antitrust_Media_Coding.csv") # load RA coded media data from NexusUni
> AT_med$date <-  as.Date(AT_med$Date,'%m/%d/%Y') #ensure date format is correct. The year must be in 4-digit format.
> AT_med$Year <- as.numeric(format(AT_med$date,'%Y')) #isolate the year of each article
>
> # Create a single measure that addressess any Democracy/Political Power articles
> AT_med$DemPwr <- ifelse(AT_med$Democracy==1, 1, ifelse(AT_med$PolPwr==1, 1, 0))
>
> # Percent of articles on each topic for the full dataset, reported in "Analysis of Antitrust Media Coverage" section
> (sum(AT_med$Competition, na.rm = TRUE)/525) # 57%
[1] 0.5733333
> (sum(AT_med$Prices , na.rm = TRUE)/525) #24%
[1] 0.2438095
> (sum(AT_med$Fairness , na.rm = TRUE)/525) #22%
[1] 0.2209524
> (sum(AT_med$DemPwr, na.rm = TRUE)/525) # 16%
```

```
[1] 0.1561905
> (sum(AT_med$PunishFirm, na.rm = TRUE)/525) # 8%
[1] 0.08190476
> # the following percentages of coverage are for informational purposes only and are not reported in the paper
> (sum(AT_med$SmallBus , na.rm = TRUE)/525) #19%
[1] 0.1885714
> (sum(AT_med$Efficiency, na.rm = TRUE)/525) # 21%
[1] 0.2114286
> (sum(AT_med$Democracy, na.rm = TRUE)/525) # 8%
[1] 0.07619048
> (sum(AT_med$PolPwr, na.rm = TRUE)/525) # 11%
[1] 0.1142857
>
> # create a measure of the proportion of coverage for each year that is of X type
> year <-numeric(length = length(1990:2021))
> AT.obs.per.yr <-numeric(length = length(1990:2021))
> AT.punish.per.yr <-numeric(length = length(1990:2021))
> AT.prop.punish <-numeric(length = length(1990:2021))
> AT.fair.per.yr <-numeric(length = length(1990:2021))
> AT.prop.fair <-numeric(length = length(1990:2021))
> AT.smallbus.per.yr <-numeric(length = length(1990:2021))
> AT.prop.smallbus <-numeric(length = length(1990:2021))
> AT.prices.per.yr <-numeric(length = length(1990:2021))
> AT.prop.prices <-numeric(length = length(1990:2021))
> AT.dempwr.per.yr <-numeric(length = length(1990:2021))
> AT.prop.dempwr <-numeric(length = length(1990:2021))
>
> year <- c()
> AT.obs.per.yr <- c()
> AT.punish.per.yr <-c()
> AT.prop.punish <- c()
> AT.fair.per.yr <-c()
> AT.prop.fair <- c()
> AT.smallbus.per.yr <- c()
> AT.prop.smallbus <- c()
> AT.prices.per.yr <- c()
> AT.prop.prices <- c()
> AT.dempwr.per.yr <- c()
> AT.prop.dempwr <- c()
>
> AT_df <- data.frame(matrix(ncol=0, nrow =32))
>
> loop_output <- capture.output({
+ for (i in 1990:2021){
+   AT_df$year[i -1989]<-i
+   AT_df$AT.obs.per.yr[i -1989] <- length(AT_med$Year[AT_med$Year==i])
+   AT_df$AT.punish.per.yr[i -1989] <-length(AT_med$Year[AT_med$PunishFirm ==1 & AT_med$Year==i])
+   AT_df$AT.fair.per.yr[i -1989] <-length(AT_med$Year[AT_med$Fairness ==1 & AT_med$Year==i])
+   AT_df$AT.smallbus.per.yr[i -1989] <-length(AT_med$Year[AT_med$SmallBus ==1 & AT_med$Year==i])
+   AT_df$AT.prices.per.yr[i -1989] <-length(AT_med$Year[AT_med$Prices ==1 & AT_med$Year==i])
+   AT_df$AT.dempwr.per.yr[i -1989] <-length(AT_med$Year[AT_med$DemPwr ==1 & AT_med$Year==i])
+ }})
>
> AT_df$AT.prop.punish <- AT_df$AT.punish.per.yr/AT_df$AT.obs.per.yr
> AT_df$AT.prop.fair <- AT_df$AT.fair.per.yr/AT_df$AT.obs.per.yr
> AT_df$AT.prop.smallbus <- AT_df$AT.smallbus.per.yr/AT_df$AT.obs.per.yr
> AT_df$AT.prop.prices <- AT_df$AT.prices.per.yr/AT_df$AT.obs.per.yr
> AT_df$AT.prop.dempwr <- AT_df$AT.dempwr.per.yr/AT_df$AT.obs.per.yr
>
> # create a measure of the proportion of coverage for each 4-year window that is of X type
> AT_4yr <- data.frame(matrix(ncol=0, nrow =8))
> AT_4yr$period <- c()
> AT_4yr$punish.prop.per <- c()
> punish.prop1 <- c()
> AT_4yr$fair.prop.per <- c()
> fair.prop1 <- c()
> AT_4yr$smallbus.prop.per <- c()
> smallbus.prop1 <- c()
> AT_4yr$prices.prop.per <- c()
```

```
> prices.prop1 <- c()
> AT_4yr$dempwr.prop.per <- c()
> dempwr.prop1 <- c()
>
> loop_output <- capture.output({
+     for(j in c(1990, 1994, 1998, 2002, 2006, 2010, 2014, 2018)){
+       for(k in 0:3){
+     time  <- j+k
+     period <- ifelse(j==1990, 1, ifelse(j==1994, 2, ifelse(j==1998, 3, ifelse(j==2002, 4, ifelse(j==2006, 5,
ifelse(j==2010, 6, ifelse(j==2014, 7, ifelse(j==2018, 8, NA)))))))
+     AT_4yr$period[period]<- period
+     punish.prop1[1] <- ifelse(k==0, AT_df$AT.prop.punish[AT_df$year==time], punish.prop1[1])
+     punish.prop1[2] <- ifelse(k==1, AT_df$AT.prop.punish[AT_df$year==time], punish.prop1[2])
+     punish.prop1[3] <- ifelse(k==2, AT_df$AT.prop.punish[AT_df$year==time], punish.prop1[3])
+     punish.prop1[4] <- ifelse(k==3, AT_df$AT.prop.punish[AT_df$year==time], punish.prop1[4])
+     AT_4yr$punish.prop.per[period]   <- ((punish.prop1[1] + punish.prop1[2] + punish.prop1[3] + punish.prop1[4])/4)
+     fair.prop1[1] <- ifelse(k==0, AT_df$AT.prop.fair[AT_df$year==time], fair.prop1[1])
+     fair.prop1[2] <- ifelse(k==1, AT_df$AT.prop.fair[AT_df$year==time], fair.prop1[2])
+     fair.prop1[3] <- ifelse(k==2, AT_df$AT.prop.fair[AT_df$year==time], fair.prop1[3])
+     fair.prop1[4] <- ifelse(k==3, AT_df$AT.prop.fair[AT_df$year==time], fair.prop1[4])
+     AT_4yr$fair.prop.per[period]   <- ((fair.prop1[1] + fair.prop1[2] + fair.prop1[3] + fair.prop1[4])/4)
+     smallbus.prop1[1] <- ifelse(k==0, AT_df$AT.prop.smallbus[AT_df$year==time], smallbus.prop1[1])
+     smallbus.prop1[2] <- ifelse(k==1, AT_df$AT.prop.smallbus[AT_df$year==time], smallbus.prop1[2])
+     smallbus.prop1[3] <- ifelse(k==2, AT_df$AT.prop.smallbus[AT_df$year==time], smallbus.prop1[3])
+     smallbus.prop1[4] <- ifelse(k==3, AT_df$AT.prop.smallbus[AT_df$year==time], smallbus.prop1[4])
+     AT_4yr$smallbus.prop.per[period]   <- ((smallbus.prop1[1] + smallbus.prop1[2] + smallbus.prop1[3] +
smallbus.prop1[4])/4)
+     prices.prop1[1] <- ifelse(k==0, AT_df$AT.prop.prices[AT_df$year==time], prices.prop1[1])
+     prices.prop1[2] <- ifelse(k==1, AT_df$AT.prop.prices[AT_df$year==time], prices.prop1[2])
+     prices.prop1[3] <- ifelse(k==2, AT_df$AT.prop.prices[AT_df$year==time], prices.prop1[3])
+     prices.prop1[4] <- ifelse(k==3, AT_df$AT.prop.prices[AT_df$year==time], prices.prop1[4])
+     AT_4yr$prices.prop.per[period]   <- ((prices.prop1[1] + prices.prop1[2] + prices.prop1[3] + prices.prop1[4])/4)
+     dempwr.prop1[1] <- ifelse(k==0, AT_df$AT.prop.dempwr[AT_df$year==time], dempwr.prop1[1])
+     dempwr.prop1[2] <- ifelse(k==1, AT_df$AT.prop.dempwr[AT_df$year==time], dempwr.prop1[2])
+     dempwr.prop1[3] <- ifelse(k==2, AT_df$AT.prop.dempwr[AT_df$year==time], dempwr.prop1[3])
+     dempwr.prop1[4] <- ifelse(k==3, AT_df$AT.prop.dempwr[AT_df$year==time], dempwr.prop1[4])
+     AT_4yr$dempwr.prop.per[period]   <- ((dempwr.prop1[1] + dempwr.prop1[2] + dempwr.prop1[3] + dempwr.prop1[4])/4)
+   }}})
>
> # Bar plot, Prices by percent
> prices_perc <- AT_4yr$prices.prop.per * 100
> barplot(prices_perc,
+         #names.arg = AT_4yr$period,
+         xlab = "", cex.axis = 1.5, cex.lab = 1.5, cex.main = 1.5,
+         ylab = "",
+         main = "Percent of `Prices' Articles",
+         ylim = c(0, 40), # Set the y-axis limits to 0-100%
+         col = "black",
+         border = "black",
+         #density = 10, # Add pattern to the bars
+         args.legend = list(x = "topright")
+         # Create a custom function to format the y-axis labels as percentages
+         #axisFUN = function(x, ...) paste0(x, "%")
+ )
>
> # Bar plot, Punish by percent
> punish_perc <- AT_4yr$punish.prop.per * 100
> barplot(punish_perc,
+        # names.arg = AT_4yr$period,
+         xlab = "", cex.axis = 1.5, cex.lab = 1.5, cex.main = 1.5,
+         ylab = "",
+         main = "Percent of `Punish Firms' Articles",
+         ylim = c(0, 40), # Set the y-axis limits to 0-100%
+         col = "black",
+         border = "black",
+         #density = 10, # Add pattern to the bars
+         args.legend = list(x = "topright")
+         # Create a custom function to format the y-axis labels as percentages
+         #axisFUN = function(x, ...) paste0(x, "%")
```

```
+ )
>
> # Bar plot, Small Business by percent
> smallbus_perc <- AT_4yr$smallbus.prop.per * 100
> barplot(smallbus_perc,
+         #names.arg = AT_4yr$period,
+         xlab = "", cex.axis = 1.5, cex.lab = 1.5, cex.main = 1.5,
+         ylab = "",
+         main = "Percent of 'Small Business' Articles",
+         ylim = c(0, 40), # Set the y-axis limits to 0-100%
+         col = "black",
+         border = "black",
+         #density = 10, # Add pattern to the bars
+         args.legend = list(x = "topright")
+         # Create a custom function to format the y-axis labels as percentages
+         #axisFUN = function(x, ...) paste0(x, "%")
+ )
>
> # Bar plot, Fair by percent
> fair_perc <- AT_4yr$fair.prop.per * 100
> barplot(fair_perc,
+         #names.arg = AT_4yr$period,
+         xlab = "", cex.axis = 1.5, cex.lab = 1.5, cex.main = 1.5,
+         ylab = "",
+         main = "Percent of 'Fairness' Articles",
+         ylim = c(0, 40), # Set the y-axis limits to 0-100%
+         col = "black",
+         border = "black",
+         #density = 10, # Add pattern to the bars
+         args.legend = list(x = "topright")
+         # Create a custom function to format the y-axis labels as percentages
+         #axisFUN = function(x, ...) paste0(x, "%")
+ )
>
> # Bar plot, Democracy/Power by percent
> dempwr_perc <- AT_4yr$dempwr.prop.per * 100
> barplot(dempwr_perc,
+         #names.arg = AT_4yr$period,
+         xlab = "Four Year Intervals 1990-2021", cex.axis = 1.5, cex.lab = 1.5, cex.main = 1.5,
+         ylab = "",
+         main = "Percent of `Democracy' \n and `Political Power' Articles",
+         ylim = c(0, 40), # Set the y-axis limits to 0-100%
+         col = "black",
+         border = "black",
+         #density = 10, # Add pattern to the bars
+         args.legend = list(x = "topright")
+         # Create a custom function to format the y-axis labels as percentages
+         #axisFUN = function(x, ...) paste0(x, "%")
+ )
>
> # Figure A1 of the appendix presents the same information as Figure 1 of the manuscript
> # Figure A1 was manually generated using excel
>
>
2025-03-12 09:56:44.013 R[36904:7859856] allowedContentsTypes : UTType pdf does not have a valid
preferredFilenameExtension and will be ignored when validating the file name
>
```