Data Science Final Project

Bilge Deniz Kocak

11/18/2021

* Datasets: General Election Results:

# reading in the election results  
election\_results <- "https://tr.wikipedia.org/wiki/T%C3%BCrkiye%27de\_se%C3%A7imler" %>%   
 read\_html() %>%   
 html\_nodes("table")  
  
# the election results were parts of different tables, so I am reading them all in seperately  
election\_results\_Jun\_2018 <- election\_results %>% .[[5]] %>% html\_table() %>% dplyr::select(c(2, 5, 7)) %>% mutate(Sandalye = paste(Sandalye, " / 600", sep="")) %>% rename ("Sandalye Dağılımı" = "Sandalye") %>% mutate(Year = 2018, Month = 6)  
election\_results\_Nov\_2015 <- election\_results %>% .[[6]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(4) %>% mutate(Year = 2015, Month = 11) %>% rename("Sandalye Dağılımı" = "Sandalye dağılımı")  
election\_results\_Jun\_2015 <- election\_results %>% .[[6]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(4) %>% mutate(Year = 2015, Month = 6) %>% rename("Sandalye Dağılımı" = "Sandalye dağılımı")  
election\_results\_Jun\_2011 <- election\_results %>% .[[7]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(4) %>% mutate(Year = 2011, Month = 6)  
election\_results\_Jul\_2007 <- election\_results %>% .[[7]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(4) %>% mutate(Year = 2007, Month = 7)  
election\_results\_Nov\_2002 <- election\_results %>% .[[8]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(3) %>% mutate(Year = 2002, Month = 11)  
election\_results\_Apr\_1999 <- election\_results %>% .[[8]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(6) %>% mutate(Year = 1999, Month = 4)  
election\_results\_Dec\_1995 <- election\_results %>% .[[9]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(5) %>% mutate(Year = 1995, Month = 12)  
election\_results\_Oct\_1991 <- election\_results %>% .[[9]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(5) %>% mutate(Year = 1991, Month = 10)  
election\_results\_Oct\_1987 <- election\_results %>% .[[10]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(4) %>% mutate(Year = 1987, Month = 10)  
election\_results\_Nov\_1983 <- election\_results %>% .[[10]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(3) %>% mutate(Year = 1983, Month = 11)  
election\_results\_Jun\_1977 <- election\_results %>% .[[11]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(7) %>% mutate(Year = 1977, Month = 6)  
election\_results\_Oct\_1973 <- election\_results %>% .[[11]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(8) %>% mutate(Year = 1973, Month = 10)  
election\_results\_Oct\_1969 <- election\_results %>% .[[12]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(9) %>% mutate(Year = 1969, Month = 10)  
election\_results\_Oct\_1965 <- election\_results %>% .[[12]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(7) %>% mutate(Year = 1965, Month = 10)  
election\_results\_Oct\_1961 <- election\_results %>% .[[13]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(4) %>% mutate(Year = 1961, Month = 10)  
election\_results\_Oct\_1957 <- election\_results %>% .[[13]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(4) %>% mutate(Year = 1957, Month = 10)  
election\_results\_May\_1954 <- election\_results %>% .[[14]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(1:3)) %>% head(4) %>% mutate(Year = 1954, Month = 5)  
election\_results\_May\_1950 <- election\_results %>% .[[14]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(6:8)) %>% head(4) %>% mutate(Year = 1950, Month = 5)  
election\_results\_Jul\_1946 <- election\_results %>% .[[14]] %>% html\_table() %>% row\_to\_names(row\_number = 1) %>% dplyr::select(c(11:13)) %>% head(3) %>% mutate(Year = 1946, Month = 7)  
  
# I am combining all those election results I got for different years as general\_election\_results, translating it to English, and using abbreviations for party names  
general\_election\_results <- do.call("rbind",   
 list(election\_results\_Jun\_2018, election\_results\_Nov\_2015,  
 election\_results\_Jun\_2015, election\_results\_Jun\_2011,  
 election\_results\_Jul\_2007, election\_results\_Nov\_2002,  
 election\_results\_Apr\_1999, election\_results\_Dec\_1995,  
 election\_results\_Oct\_1991, election\_results\_Oct\_1987,  
 election\_results\_Nov\_1983, election\_results\_Jun\_1977,  
 election\_results\_Oct\_1973, election\_results\_Oct\_1969,  
 election\_results\_Oct\_1965, election\_results\_Oct\_1961,  
 election\_results\_Oct\_1957, election\_results\_May\_1954,  
 election\_results\_May\_1950, election\_results\_Jul\_1946)) %>%  
 rename("Seats" = "Sandalye Dağılımı") %>%  
 separate(Seats, into = c("MP", "TotalMP"), sep = "/") %>%  
 mutate(Parti = case\_when(  
 Parti == "Adalet ve Kalkınma Partisi" ~ "AKP",  
 Parti == "Cumhuriyet Halk Partisi" ~ "CHP",  
 Parti == "Halkların Demokratik Partisi" ~ "HDP",  
 Parti == "Milliyetçi Hareket Partisi" ~ "MHP",  
 Parti == "İYİ Parti" ~ "İYİ",  
 Parti == "Saadet Partisi" ~ "SP",  
 Parti == "Hür Dava Partisi" ~ "HÜDAPAR",  
 Parti == "Vatan Partisi" ~ "VATAN",  
 Parti == "Bağımsızlar" ~ "Bağımsız",  
 Parti == "Demokratik Sol Parti" ~ "DSP",  
 Parti == "Fazilet Partisi" ~ "FP",  
 Parti == "Anavatan Partisi" ~ "ANAP",  
 Parti == "Doğru Yol Partisi" ~ "DYP",  
 Parti == "Refah Partisi" ~ "RP",  
 Parti == "Türkiye İşçi Partisi" ~ "TİP",   
 Parti == "Türkiye Birlik Partisi" ~ "TBP",   
 Parti == "Birlik Partisi" ~ "TBP",  
 Parti == "Milli Selamet Partisi" ~ "MSP",  
 Parti == "Cumhuriyetçi Güven Partisi" ~ "CGP",  
 Parti == "Sosyaldemokrat Halkçı Parti" ~ "SHP",  
 Parti == "Milliyetçi Demokrasi Partisi" ~ "MDP",  
 TRUE ~ Parti)) %>%  
 mutate\_all(.funs = funs(str\_trim)) %>%  
 mutate(Proportion = as.numeric(MP)/as.numeric(TotalMP)) # calculating the proportion of MPs the parties got as a result of the election

* Datasets: Survey Data:

#reading in the data gotten from different files that were a part of the project on Kaggle  
data1 <- read\_csv("data.csv")

## Rows: 168 Columns: 16

## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (16): Timestamp, Cinsiyet, Yas, Bolge, Egitim, soru1, soru2, soru3, soru...

##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

data2 <- read\_csv("datav2.csv")

## Rows: 308 Columns: 16

## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (16): Timestamp, Cinsiyet, Yas, Bolge, Egitim, soru1, soru2, soru3, soru...

##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

data3 <- read\_csv("yonelimfinal.csv")

## Rows: 885 Columns: 16

## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (16): Timestamp, Cinsiyet, Yas, Bolge, Egitim, soru1, soru2, soru3, soru...

##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# combining the datasets as survey\_data and translating it to English  
survey\_data <- data1 %>%   
 rbind(data2) %>%   
 rbind(data3) %>%   
 unique() %>%   
 rename("gender" = "Cinsiyet",  
 "age" = "Yas",  
 "region" = "Bolge",  
 "education" = "Egitim",  
 "economicalSituation" = "soru1",  
 "educationalReform" = "soru2",  
 "againstPrivatization" = "soru3",  
 "execution" = "soru4",  
 "objectiveMedia" = "soru5",  
 "alcoholBan" = "soru6",   
 "secularState" = "soru7",  
 "abortionBan" = "soru8",  
 "stateOfEmergency" = "soru9",  
 "newParty" = "soru10",  
 "party" = "parti") %>%  
 mutate(gender = ifelse(gender == "Erkek", "Male", gender),  
 gender = ifelse(gender == "Kadın", "Female", gender),  
 region = case\_when(  
 region == "Akdeniz" ~ "Mediterranean",  
 region == "Güneydoğu" ~ "Southeastern Anatolia",  
 region == "Doğu Anadolu" ~ "Eastern Anatolia",  
 region == "Ege" ~ "Aegean",  
 region == "İç Anadolu" ~ "Central Anatolia",  
 region == "Karadeniz" ~ "Black Sea",  
 TRUE ~ region),  
 education = case\_when(  
 education == "İlkokul" ~ "Primary School",  
 education == "Ortaokul" ~ "Middle School",  
 education == "Lise" ~ "High School",   
 education == "Ön Lisans" ~ "2-year College Degree",  
 education == "Lisans" ~ "4-year College Degree",  
 education == "Lisans Üstü" ~ "Graduate School"),  
 economicalSituation = ifelse(economicalSituation == "Hayır", "No", economicalSituation),  
 economicalSituation = ifelse(economicalSituation == "Evet", "Yes", economicalSituation),  
 educationalReform = ifelse(educationalReform == "Hayır", "No", educationalReform),  
 educationalReform = ifelse(educationalReform == "Evet", "Yes", educationalReform),  
 againstPrivatization = ifelse(againstPrivatization == "Hayır", "No", againstPrivatization),  
 againstPrivatization = ifelse(againstPrivatization == "Evet", "Yes", againstPrivatization),  
 execution = ifelse(execution == "Hayır", "No", execution),  
 execution = ifelse(execution == "Evet", "Yes", execution),  
 objectiveMedia = ifelse(objectiveMedia == "Hayır", "No", objectiveMedia),  
 objectiveMedia = ifelse(objectiveMedia == "Evet", "Yes", objectiveMedia),  
 alcoholBan = ifelse(alcoholBan == "Hayır", "No", alcoholBan),  
 alcoholBan = ifelse(alcoholBan == "Evet", "Yes", alcoholBan),  
 secularState = ifelse(secularState == "Hayır", "No", secularState),  
 secularState = ifelse(secularState == "Evet", "Yes", secularState),  
 abortionBan = ifelse(abortionBan == "Hayır", "No", abortionBan),  
 abortionBan = ifelse(abortionBan == "Evet", "Yes", abortionBan),  
 stateOfEmergency = ifelse(stateOfEmergency == "Hayır", "No", stateOfEmergency),  
 stateOfEmergency = ifelse(stateOfEmergency == "Evet", "Yes", stateOfEmergency),  
 newParty = ifelse(newParty == "Hayır", "No", newParty),  
 newParty = ifelse(newParty == "Evet", "Yes", newParty))

* Datasets: Political Chronology:

#reading in the data, translating it to English, assigning 1-7 for political positions  
political\_chronology <- read\_csv("political\_chronology.csv") %>%  
 dplyr::select(-c("Source", "Publications", "Notes")) %>%  
 mutate(PoliticalPosition = case\_when(  
 PoliticalPosition == "Aşırı Sol" ~ "Far-left",  
 PoliticalPosition == "Sol" ~ "Left",   
 PoliticalPosition == "Merkez Sol" ~ "Center-left",   
 PoliticalPosition == "Merkez Sağ" ~ "Center-right",   
 PoliticalPosition == "Sağ" ~ "Right",   
 PoliticalPosition == "Merkez" ~ "Center",   
 PoliticalPosition == "Aşırı Sağ" ~ "Far-right"),  
 Ideology = case\_when(  
 Ideology == "Bolşevizm" ~ "Bolshevism",   
 Ideology == "Komünizm" ~ "Communism",   
 Ideology == "Atatürkçülük, Sosyal liberalizm, Sosyal demokrasi" ~ "Ataturkism, Social Liberalism, Social Democracy",   
 Ideology == "Muhafazakarlık, Liberalizm" ~ "Conservatism, Liberalism",   
 Ideology == "Liberalizm" ~ "Liberalism",   
 Ideology == "Ekonomik liberalizm, Muhafazakarlık" ~ "Economic Liberalism, Conservatism",   
 Ideology == "Marksizm, Sosyalizm" ~ "Marxism, Socialism",   
 Ideology == "Muhafazakarlık, Milliyetçilik" ~ "Conservatism, Nationalism",   
 Ideology == "Milliyetçilik" ~ "Nationalism",   
 Ideology == "Marksizm, Leninizm, Sosyalizm" ~ "Marxism, Leninism, Socialism",   
 Ideology == "Milliyetçilik, Turancılık, Türkçülük" ~ "Nationalism, Turanism, pan-Turkism",   
 Ideology == "Bilimsel Sosyalizm" ~ "Scientific Socialism",   
 Ideology == "Liberalizm, Mufazakarlık" ~ "Conservatism, Liberalism",   
 Ideology == "Marksizm-Leninizm, Maoizm" ~ "Marxism, Leninism, Maoism",   
 Ideology == "Sosyalizm" ~ "Socialism",   
 Ideology == "Marksizm-Leninizm" ~ "Marxism, Leninism",   
 Ideology == "Bilimsel Sosyalizm, Maoizm, Milli Demokratik Devrim" ~ "Scientific Socialism, Maoism, National Democratic Revolution",   
 Ideology == "Ülkücülük, Türk Milliyetçiliği, Türkçülük, Turancılık" ~ "Idealism, Turkish Nationalism, pan-Turkism, Turanism",   
 Ideology == "Milli Görüş,İslamcılık,Muhafazakarlk" ~ "National Vision, Islamism, Conservatism",   
 Ideology == "Milliyetçilik,Demokrasi,Anti-liberalizm,Anti-sosyalizm" ~ "Nationalism, Democracy, Anti-liberalism, Anti-socialism",   
 Ideology == "Soyalizm, Maoizm" ~ "Socialism, Maoism",   
 Ideology == "Sosyal Demokrasi, Laiklik" ~ "Social Democracy, Securalism",   
 Ideology == "Muhafazakârlık, Türk milliyetçiliği, Ekonomik liberalizm" ~ "Conservatism, Turkish Nationalism, Economic Liberalism",   
 Ideology == "Sosyal Demokrasi, Atatürkçülük" ~ "Social Democracy, Ataturkism",   
 Ideology == "Bilimsel Sosyalizm, Maoizm" ~ "Scientific Socialism, Maoism",   
 Ideology == "Kürt milliyetçiliği" ~ "Kurdish Nationalism",   
 Ideology == "Milliyetçilik,Halkçılık,Sosyalizm" ~"Nationalism, Populism, Socialism",   
 Ideology == "İslamcılık, Ümmetçilik, Muhafazakârlık, Yeni Osmanlıcılık, Türk milliyetçiliği, Ülkücülük, Turancılık" ~  
 "Islamism, pan-Islamism, Conservatism, neo-Ottomanism, Turkish Nationalism, Idealism, Turanism",   
 Ideology == "Sosyalizm, Komünizm" ~ "Socialism, Communism",   
 Ideology == "Klasik Liberalizm" ~ "Classical Liberalism",   
 Ideology == "Liberter Sosyalizm" ~ "Libertarian Socialism",   
 Ideology == "Troçkizm" ~ "Trotskyism",   
 Ideology == "Kemalizm, Türk Milliyetçiliği" ~ "Ataturkism, Turkish Nationalism",   
 Ideology == "Milliyetçilik,Muhafazakarlık" ~ "Conservatism, Nationalism",   
 Ideology == "Muhafazakarlık,Ekonomik liberalizm,Anti-milliyetçilik" ~ "Conservatism, Economic Liberalism, Anti-nationalism",   
 Ideology == "Milli Görüş, Muhafazakarlık, İslamcılık, Ümmetçilik" ~ "National Vision, Conservatism, Islamism, pan-Islamism",   
 Ideology == "Muhafazakarlık, Türk milliyetçiliği" ~ "Conservatism, Turkish Nationalism",   
 Ideology == "Komünizm, Marksizim-Leninizm" ~ "Communism, Marxism, Leninism",   
 Ideology == "Milliyetçilik,Halkçılık,Sosyalizm" ~ "Nationalism, Populism, Socialism",   
 Ideology == "Atatürkçülük" ~ "Ataturkism",   
 Ideology == "Sosyalist Demokrasi,Marksizm-Leninizm,Enternasyonalizm" ~ "Socialism Democracy, Marxism, Leninism, Internationalism",   
 Ideology == "Marksizm-Leninizm,Bilimsel Sosyalizm" ~ "Marxism, Leninism, Scientific Socialism",   
 Ideology == "Atatürkçülük, Türk Milliyetçiliği, Sosyal Demokrasi" ~ "Ataturkism, Turkish Nationalism, Social Democracy",   
 Ideology == "Marksizm-Leninizm,Sosyalizm" ~ "Marxism, Leninism, Socialism",   
 Ideology == "Sosyal Demokrasi, Kürt Milliyetçiliği" ~ "Social Democracy, Kurdish Nationalism",   
 Ideology == "Demokrat" ~ "Democrat",   
 Ideology == "Ekonomik Liberalizm,Demokrasi,Muhafazakarlık,Milliyetçilik" ~ "Economic Liberalism, Democracy, Conservatism, Nationalism",   
 Ideology == "Türk milliyetçiliği" ~ "Turkish Nationalism",   
 Ideology == "İslamcılık,Ümmetçilik" ~ "Islamism, pan-Islamism",   
 Ideology == "Devrimci Marksizm, Komünizm, Sosyalizm" ~ "Revolutionary Marxism, Communism, Socialism",   
 Ideology == "Türk Milliyetçiliği, Halkçılık, Sosyal Adaletçilik" ~ "Turkish Nationalism, Populism, Social Justice",   
 Ideology == "Sosyal Demokrasi,Sosyal Liberalizm,Liberal Milliyetçilik,Etnik Milliyetçilik" ~ "Social Democracy, Social Liberalism, Liberal Nationalism, Ethnic Nationalism",   
 Ideology == "Türkçülük,Ulusalcılık,Pan-Türkizm" ~ "Turkism, Nationalism, pan-Turkism",   
 Ideology == "Sosyal Demokrasi" ~ "Social Democracy",   
 Ideology == "Muhafazakârlık, Ümmetçilik, Ekonomik liberalizm" ~ "Conservatism, pan-Islamism, Economic Liberalism",   
 Ideology == "Demokratik sosyalizm, LGBT, Feminizm, Anti-millietçilik" ~ "Democratic Socialism, LGBT, Feminism, Anti-nationalism",   
 Ideology == "Türk Milliyetçiliği, Sağcılık, Hakçılık" ~ "Turkish Nationalism, Rightism, Populism",   
 Ideology == "Sol liberteryenizm, Yeşil liberalizm, Sosyal liberalizm" ~ "Left Libertarianism, Green Liberalism, Social Liberalism",   
 Ideology == "İslami Demokrasi, Ümmetçilik, Muhafazakârlık, Sosyal adaletçilik" ~ "Islamic Democracy, pan-Islamism, Conservatism, Social Justice",   
 Ideology == "Türk milliyetçiliği,Muhafazakarlık" ~ "Turkish Nationalism, Conservatism",   
 Ideology == "Marksizm-Leninizm,Enternasyonalizm" ~ "Marxism, Leninism, Internationalism",   
 Ideology == "Çevrecilik" ~ "Environmentalism",   
 Ideology == "Komünizm, Marksizm-Leninizm" ~ "Communism, Marxism, Leninism",   
 Ideology == "Sol liberteryenizm, Sosyal liberalizm" ~ "Left Libertarianism, Social Liberalism",   
 Ideology == "Marksizm-Leninizm, Bilimsel sosyalizm" ~ "Marxism, Leninism, Scientific Socialism",   
 Ideology == "Ekonomik liberalizm, Muhafazakârlık" ~ "Economic Liberalism, Conservatism",   
 Ideology == "Marksizm" ~ "Marxism",   
 Ideology == "Ulusalcılık, Muhafazakarlık" ~ "Nationalism, Conservatism",   
 Ideology == "Türk Milliyetçiliği" ~ "Turkish Nationalism")) %>%  
 rename(Parti = Abbreviation) %>%  
 mutate(Founded = ifelse(Parti == "CHP", 1992, Founded), # these parties were closed and opened again in the history. I am just assigning one of the durations that they were operating in for now, since I will address this later  
 Dissolved = ifelse(Parti == "CHP", "-", Dissolved),  
 Founded = ifelse(Leader == "Hikmet Kıvılcımlı", 1954, Founded),  
 Dissolved = ifelse(Leader == "Hikmet Kıvılcımlı", 1957, Dissolved),  
 Founded = ifelse(Parti == "MHP", 1969, Founded),  
 Dissolved = ifelse(Parti == "MHP", 1980, Dissolved)) %>%  
 rbind(c("Vatan Partisi", "VP", "Hikmet Kıvılcımlı", 1975, 1981, "İstanbul Sulh Ceza Hakimliği Karar", "TKP1920", "+HKP", "Far-left", "Marxism, Leninism, Socialism")) %>% # was in another dataset but was missing here, so I thought I could add it myself  
 mutate(Dissolved = ifelse(Dissolved == "-", 2021, Dissolved)) %>% # if the parties did not dissolve yet, I will assign 2021 -- this is just for the purposes of graphing the intervals later  
 mutate(PoliticalPosition = case\_when(  
 PoliticalPosition == "Far-right" ~ 1, # assigning 1-7 for political positions  
 PoliticalPosition == "Right" ~ 2,  
 PoliticalPosition == "Center-right" ~ 3,   
 PoliticalPosition == "Center" ~ 4,   
 PoliticalPosition == "Center-left" ~ 5,   
 PoliticalPosition == "Left" ~ 6,   
 PoliticalPosition == "Far-left" ~ 7))

## Rows: 106 Columns: 13

## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (12): Name, Abbreviation, Leader, Dissolved, ReasonOfDissolution, Descen...

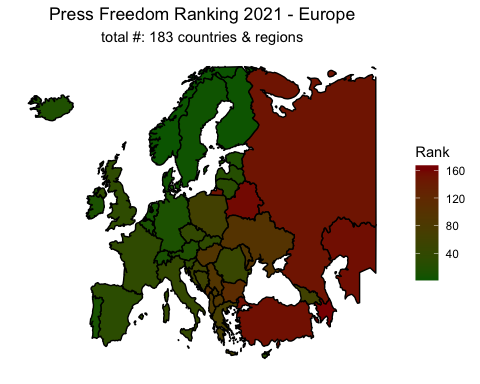
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

* Datasets: Press Freedom:

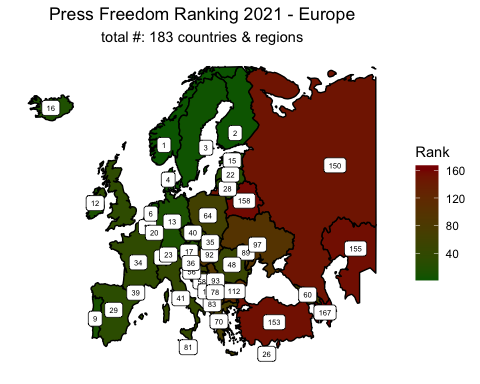
# reading in the data, renaming the countries to resolve conflicts  
press\_freedom <- "https://rsf.org/en/ranking\_table" %>%   
 read\_html() %>%   
 html\_nodes("table") %>%  
 .[[1]] %>%  
 html\_table() %>%  
 dplyr::select(c(2,3)) %>%   
 rename("region" = "Countries & regions") %>%  
 mutate(region = ifelse(region == "Bosnia-Herzegovina", "Bosnia and Herzegovina", region),  
 region = ifelse(region == "United Kingdom", "UK", region))

* Press Freedom in Europe:

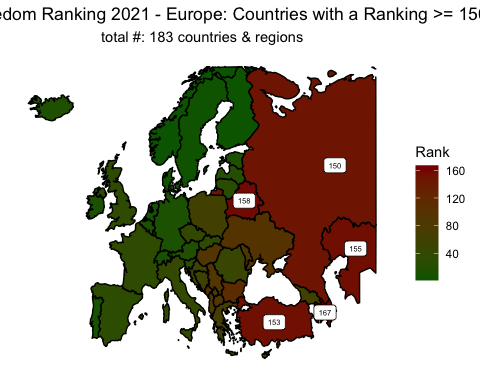
# list of european countries  
european\_countries <- c("Albania", "Andorra", "Austria", "Azerbaijan", "Belarus", "Belgium", "Bosnia and Herzegovina", "Bulgaria", "Croatia", "Cyprus", "Czech Republic", "Denmark", "Estonia", "Finland", "France", "Georgia", "Germany", "Greece", "Hungary", "Iceland", "Ireland", "Italy", "Kazakhstan", "Kosovo", "Latvia", "Liechtenstein", "Lithuania", "Luxembourg", "Malta", "Moldova", "Monaco", "Montenegro", "Netherlands", "North Macedonia", "Norway", "Poland", "Portugal", "Romania", "Russia", "San Marino", "Serbia", "Slovakia", "Slovenia", "Spain", "Sweden", "Switzerland", "Turkey", "Ukraine", "UK", "Vatican City")  
  
# press freedom data for european countries  
europe <- map\_data("world") %>%  
 filter(region %in% european\_countries) %>%  
 left\_join(press\_freedom, by = "region")   
  
# centroids for all countries  
centroids <- "https://developers.google.com/public-data/docs/canonical/countries\_csv" %>%   
 read\_html() %>%   
 html\_nodes("table") %>%  
 .[[1]] %>%   
 html\_table() %>%  
 rename("region" = "name") %>%  
 left\_join(press\_freedom, by = "region")  
  
# centroids for european countries  
european\_centroids <- centroids %>%  
 filter(region %in% european\_countries)  
  
# arranging positions for labels using the centroids, fixing Russia and Kazakhstan values for graphing purposes  
european\_positions <- european\_centroids %>%  
 mutate(latitude = ifelse(region == "Russia", 58, latitude),  
 longitude = ifelse(region == "Russia", 50, longitude),  
 latitude = ifelse(region == "Kazakhstan", 47.90227, latitude),  
 longitude = ifelse(region == "Kazakhstan", 55, longitude))  
  
# graphing Press Freedom Ranking 2021 - Europe  
ggplot() +  
 geom\_polygon(data = europe, aes(x = long, y = lat, group = group, fill = Ranking), color = "black") +   
 xlim(NA, 60) + ylim(NA, 70) +  
 scale\_fill\_gradient(high = "darkred", low = "darkgreen") +  
 labs(x = "", y = "", fill = "Rank", title = "Press Freedom Ranking 2021 - Europe", subtitle = "total #: 183 countries & regions") +   
 theme(panel.background = element\_rect(fill = "transparent",colour = NA),  
 plot.background = element\_rect(fill = "transparent",colour = NA),  
 axis.title = element\_blank(),  
 axis.text = element\_blank(),  
 axis.ticks = element\_blank(),  
 plot.title = element\_text(hjust = 0.5),  
 plot.subtitle = element\_text(hjust = 0.5))



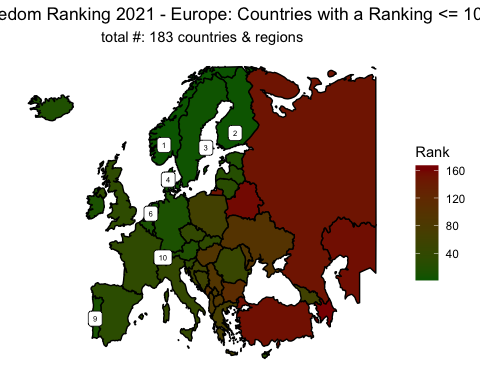
# graphing Press Freedom Ranking 2021 - Europe with labels showing the rankings for each country  
ggplot() +  
 geom\_polygon(data = europe, aes(x = long, y = lat, group = group, fill = Ranking), color = "black") +   
 xlim(NA, 60) + ylim(NA, 70) +  
 scale\_fill\_gradient(high = "darkred", low = "darkgreen") +  
 labs(x = "", y = "", fill = "Rank", title = "Press Freedom Ranking 2021 - Europe", subtitle = "total #: 183 countries & regions") +   
 theme(panel.background = element\_rect(fill = "transparent",colour = NA),  
 plot.background = element\_rect(fill = "transparent",colour = NA),  
 axis.title = element\_blank(),  
 axis.text = element\_blank(),  
 axis.ticks = element\_blank(),  
 plot.title = element\_text(hjust = 0.5),  
 plot.subtitle = element\_text(hjust = 0.5)) +   
 geom\_label(data = european\_positions, size = 2, aes(x = longitude, y = latitude, label = Ranking))



# the countries with a rank <= 10 and the countries with a rank >= 150  
ranking\_ge\_150\_countries = c("Azerbaijan", "Belarus", "Kazakhstan", "Turkey", "Russia")  
ranking\_ge\_150 <- european\_positions %>%  
 filter(region %in% ranking\_ge\_150\_countries)  
ranking\_le\_10\_countries = c("Norway", "Finland", "Sweden", "Denmark", "Netherlands", "Portugal", "Switzerland")  
ranking\_le\_10 <- european\_positions %>%  
 filter(region %in% ranking\_le\_10\_countries)  
  
# graphing the countries with a rank >= 150  
ggplot() +  
 geom\_polygon(data = europe, aes(x = long, y = lat, group = group, fill = Ranking), color = "black") +   
 xlim(NA, 60) + ylim(NA, 70) +  
 scale\_fill\_gradient(high = "darkred", low = "darkgreen") +  
 labs(x = "", y = "", fill = "Rank", title = "Press Freedom Ranking 2021 - Europe: Countries with a Ranking >= 150", subtitle = "total #: 183 countries & regions") +   
 theme(panel.background = element\_rect(fill = "transparent",colour = NA),  
 plot.background = element\_rect(fill = "transparent",colour = NA),  
 axis.title = element\_blank(),  
 axis.text = element\_blank(),  
 axis.ticks = element\_blank(),  
 plot.title = element\_text(hjust = 0.5),  
 plot.subtitle = element\_text(hjust = 0.5)) +   
 geom\_label(data = ranking\_ge\_150, size = 2, aes(x = longitude, y = latitude, label = Ranking))

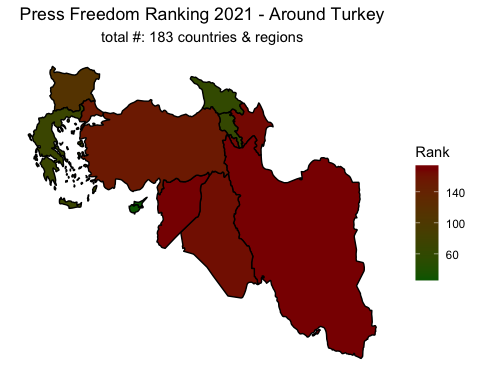


# graphing the countries with a rank <= 10  
ggplot() +  
 geom\_polygon(data = europe, aes(x = long, y = lat, group = group, fill = Ranking), color = "black") +   
 xlim(NA, 60) + ylim(NA, 70) +  
 scale\_fill\_gradient(high = "darkred", low = "darkgreen") +  
 labs(x = "", y = "", fill = "Rank", title = "Press Freedom Ranking 2021 - Europe: Countries with a Ranking <= 10", subtitle = "total #: 183 countries & regions") +   
 theme(panel.background = element\_rect(fill = "transparent",colour = NA),  
 plot.background = element\_rect(fill = "transparent",colour = NA),  
 axis.title = element\_blank(),  
 axis.text = element\_blank(),  
 axis.ticks = element\_blank(),  
 plot.title = element\_text(hjust = 0.5),  
 plot.subtitle = element\_text(hjust = 0.5)) +   
 geom\_label(data = ranking\_le\_10, size = 2, aes(x = longitude, y = latitude, label = Ranking))

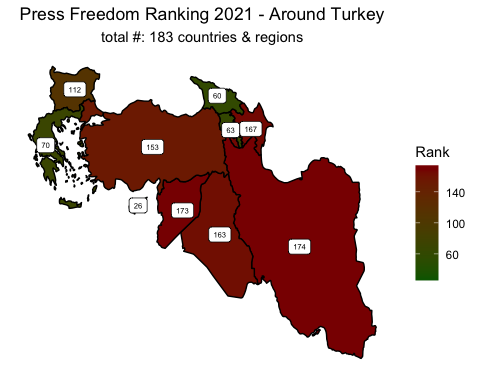


* Press Freedom in Neighbors:

# list of turkey's neighbors and Cyprus. Cyprus is included due to proximity and also having a Northern Turkish Republic.   
turkey\_and\_neighbors\_and\_cyprus <- c("Turkey", "Greece", "Bulgaria", "Azerbaijan", "Georgia", "Armenia", "Syria", "Iran", "Iraq", "Cyprus")  
  
# data for the countries around turkey  
around\_turkey <- map\_data("world") %>%  
 filter(region %in% turkey\_and\_neighbors\_and\_cyprus) %>%  
 left\_join(press\_freedom, by = "region")   
  
# centroids  
around\_turkey\_centroids <- centroids %>%  
 filter(region %in% turkey\_and\_neighbors\_and\_cyprus)  
  
# graphing the Press Freedom Ranking 2021 - Around Turkey  
ggplot() +  
 geom\_polygon(data = around\_turkey, aes(x = long, y = lat, group = group, fill = Ranking), color = "black") +   
 scale\_fill\_gradient(high = "darkred", low = "darkgreen") +  
 labs(x = "", y = "", fill = "Rank", title = "Press Freedom Ranking 2021 - Around Turkey", subtitle = "total #: 183 countries & regions") +   
 theme(panel.background = element\_rect(fill = "transparent",colour = NA),  
 plot.background = element\_rect(fill = "transparent",colour = NA),  
 axis.title = element\_blank(),  
 axis.text = element\_blank(),  
 axis.ticks = element\_blank(),  
 plot.title = element\_text(hjust = 0.5),  
 plot.subtitle = element\_text(hjust = 0.5))

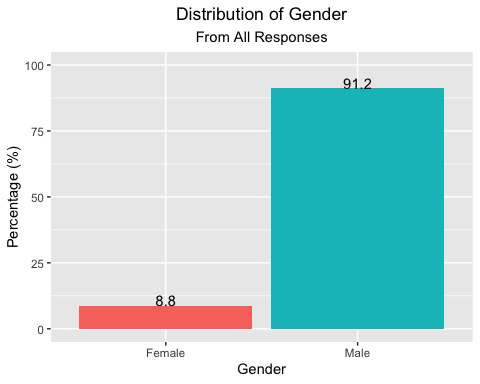


# adding labels  
ggplot() +  
 geom\_polygon(data = around\_turkey, aes(x = long, y = lat, group = group, fill = Ranking), color = "black") +   
 scale\_fill\_gradient(high = "darkred", low = "darkgreen") +  
 labs(x = "", y = "", fill = "Rank", title = "Press Freedom Ranking 2021 - Around Turkey", subtitle = "total #: 183 countries & regions") +   
 theme(panel.background = element\_rect(fill = "transparent",colour = NA),  
 plot.background = element\_rect(fill = "transparent",colour = NA),  
 axis.title = element\_blank(),  
 axis.text = element\_blank(),  
 axis.ticks = element\_blank(),  
 plot.title = element\_text(hjust = 0.5),  
 plot.subtitle = element\_text(hjust = 0.5)) +   
 geom\_label(data = around\_turkey\_centroids, size = 2, aes(x = longitude, y = latitude, label = Ranking))

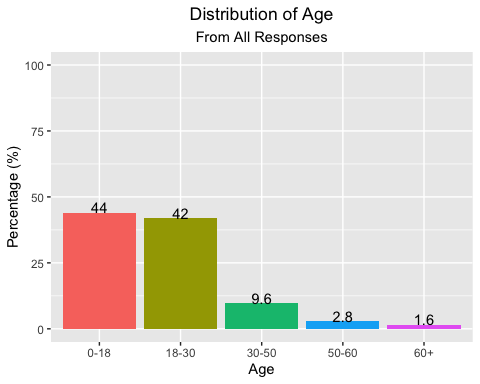


* Univariate Summaries:

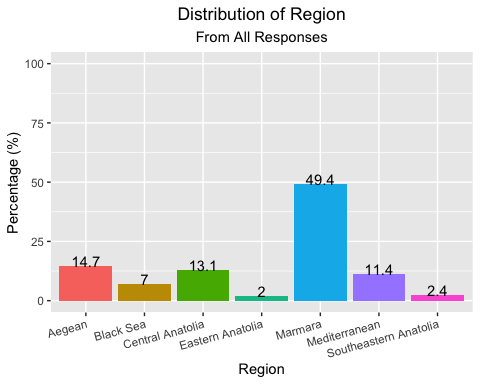
# to make the code more readable, I am creating center and center\_sub  
center <- theme(plot.title = element\_text(hjust = 0.5))  
center\_sub <- theme(plot.subtitle = element\_text(hjust = 0.5))  
  
# the function graphProducer takes an input given and produces the univariate summary for it  
graphProducer <- function(input)  
{  
 ggplot(data = survey\_data) +   
 geom\_bar(aes(x = input, y = round(..count../sum(..count..),3)\*100, fill = input)) + #round(..count../sum(..count..),3)\*100 is used to calculate the percentage with one decimal  
 geom\_text(aes(x = input, y = round(..count../sum(..count..),3)\*100 + 2, label = round(..count../sum(..count..),3)\*100), stat = "count", colour = "black") + # y value is increased by two to make the labels a little higher   
 guides(fill = "none") +   
 labs(y = "Percentage (%)", subtitle = "From All Responses") +   
 ylim(0, 100) +   
 center +   
 center\_sub  
}  
  
# the univariate summaries for all of the variables are generated using the function graphProducer declared above   
graphProducer(survey\_data$gender) + labs(title = "Distribution of Gender", x = "Gender")



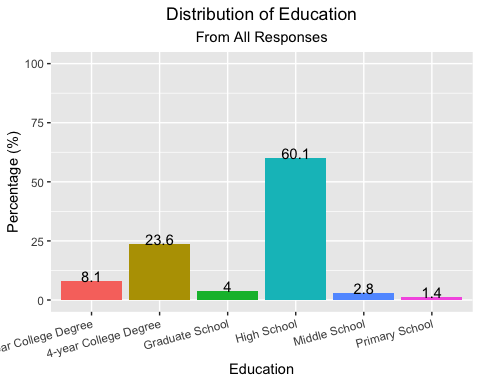
graphProducer(survey\_data$age) + labs(title = "Distribution of Age", x = "Age")



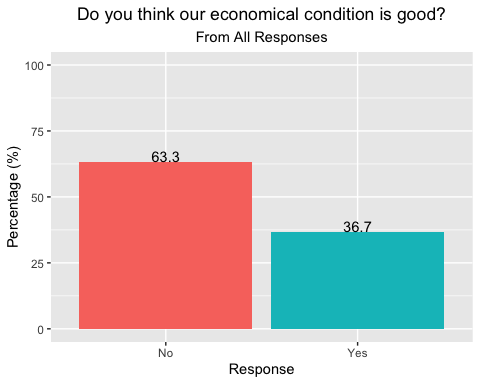
graphProducer(survey\_data$region) + labs(title = "Distribution of Region", x = "Region") + theme(axis.text.x = element\_text(angle = 15, hjust = 1));



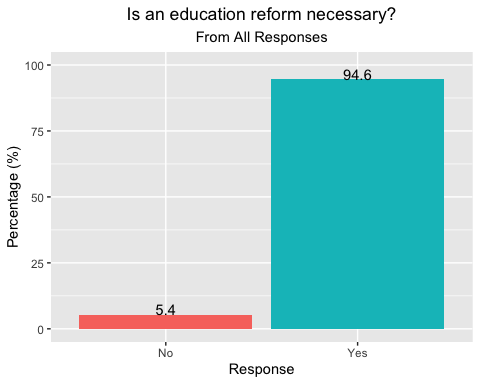
graphProducer(survey\_data$education) + labs(title = "Distribution of Education", x = "Education") + theme(axis.text.x = element\_text(angle = 15, hjust = 1));



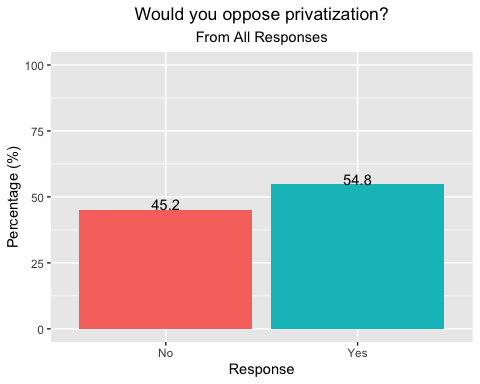
graphProducer(survey\_data$economicalSituation) + labs(title = "Do you think our economical condition is good?", x = "Response")



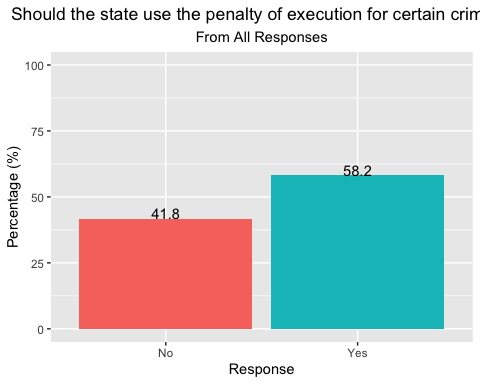
graphProducer(survey\_data$educationalReform) + labs(title = "Is an education reform necessary?", x = "Response")



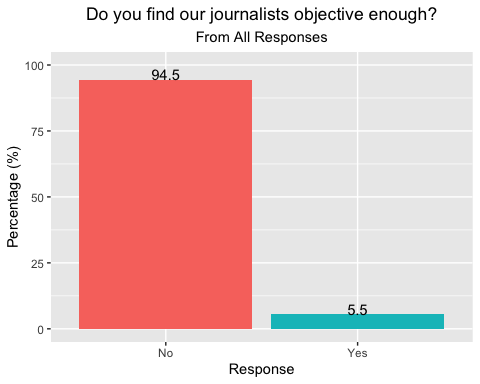
graphProducer(survey\_data$againstPrivatization) + labs(title = "Would you oppose privatization?", x = "Response")



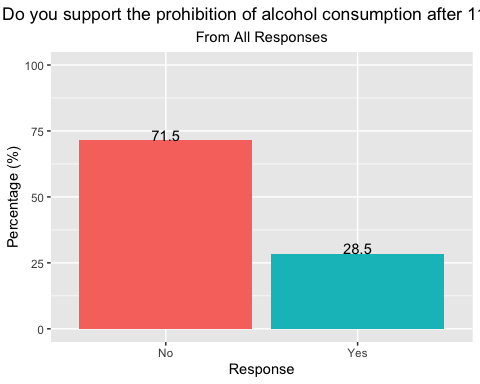
graphProducer(survey\_data$execution) + labs(title = "Should the state use the penalty of execution for certain crimes?", x = "Response")



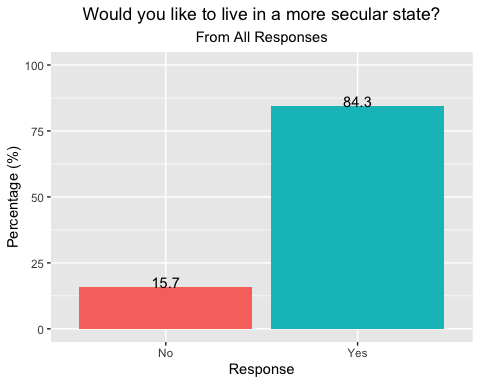
graphProducer(survey\_data$objectiveMedia) + labs(title = "Do you find our journalists objective enough?", x = "Response")



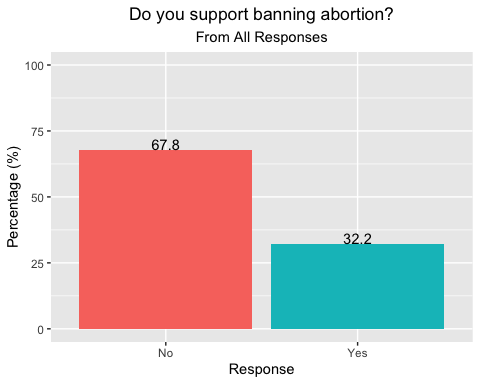
graphProducer(survey\_data$alcoholBan) + labs(title = " Do you support the prohibition of alcohol consumption after 11 pm?", x = "Response")



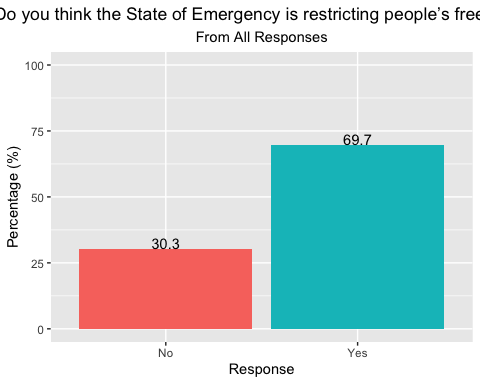
graphProducer(survey\_data$secularState) + labs(title = "Would you like to live in a more secular state?", x = "Response")



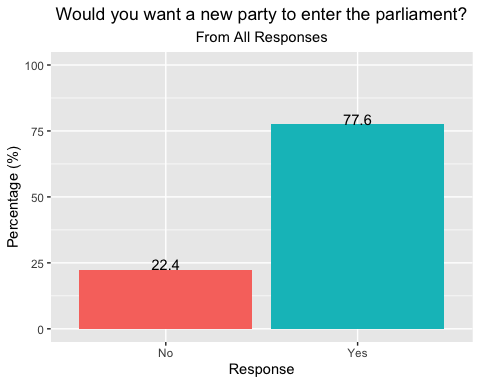
graphProducer(survey\_data$abortionBan) + labs(title = "Do you support banning abortion?", x = "Response")



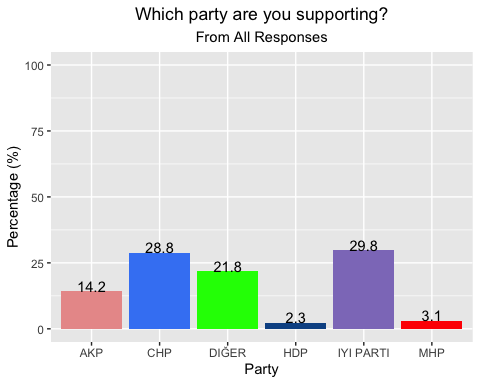
graphProducer(survey\_data$stateOfEmergency) + labs(title = "Do you think the State of Emergency is restricting people’s freedom?", x = "Response")



graphProducer(survey\_data$newParty) + labs(title = "Would you want a new party to enter the parliament?", x = "Response")

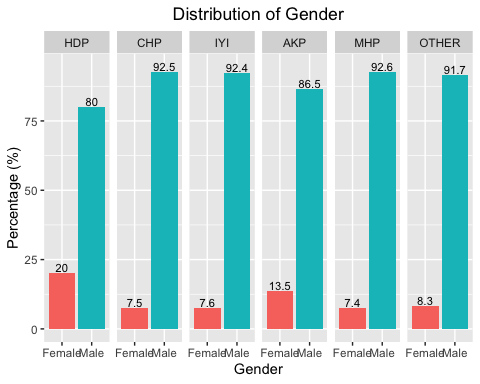


graphProducer(survey\_data$party) + labs(title = "Which party are you supporting?", x = "Party") +  
 scale\_fill\_manual(values = c("MHP" = "#ff0100", "AKP" = "#e99a99", "IYI PARTI" = "#8e7cc3", "CHP" = "#4285f4", "HDP" = "#0c5394", "DIĞER" = "green")) # used to assign colors consistently based on the political positions of the countries

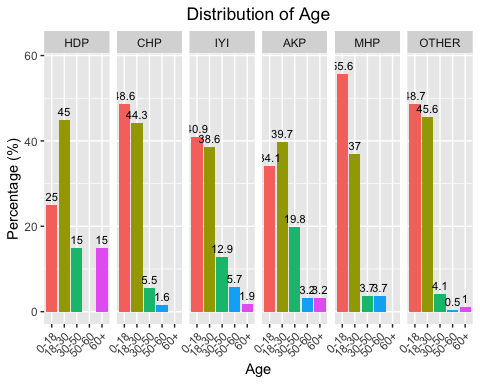


* Bivariate Summaries:

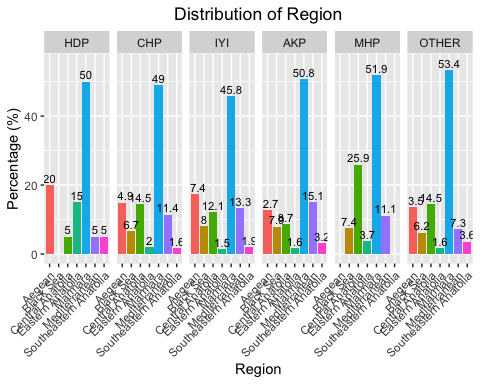
# this chunk is used to create bivariate summaries for each variable.  
  
# this factor is defined for the purpose of reordering the facets.  
survey\_data$party\_factor = factor(survey\_data$party, levels=c("HDP", "CHP", "IYI PARTI", "AKP", "MHP", "DIĞER"))  
  
# this list is used for renaming the facet labels, since I forgot to translate "diğer" to "other" in one of the chunks above  
facet\_names <- list("HDP" = "HDP",   
 "CHP" = "CHP",   
 "IYI PARTI" = "IYI",  
 "AKP" = "AKP",  
 "MHP" = "MHP",  
 "DIĞER" = "OTHER")  
  
# function facet\_labeller returns the appropriate label for each facet  
facet\_labeller <- function(variable,value){  
 return(facet\_names[value])  
}  
  
# function compareParties facets the results by the party voted for, and shows the percentage of people responding to the questions vs. their responses  
compareParties <- function(input)  
{  
 ggplot(data = survey\_data) +   
 geom\_bar(aes(x = input, y = round((..count..)/(tapply(..count..,..PANEL..,sum)[..PANEL..]),3)\*100, fill = input)) + # the y argument created thanks to https://stackoverflow.com/questions/4725339/percentage-on-y-lab-in-a-faceted-ggplot-barchart, used for the purpose of generating the percentages facet-wise  
 facet\_grid(. ~ survey\_data$party\_factor, labeller=facet\_labeller, scales = "free\_y") +   
 geom\_text(aes(x = input,   
 y = round((..count..)/(tapply(..count..,..PANEL..,sum)[..PANEL..]),3)\*100 + 2,   
 label = round((..count..)/(tapply(..count..,..PANEL..,sum)[..PANEL..]),3)\*100,   
 group=party\_factor),   
 size = 3, stat = "count", colour = "black") +  
 guides(fill = "none") +   
 labs(y = "Percentage (%)") +   
 center   
}  
  
# creating the bivariate summaries for each variable  
compareParties(survey\_data$gender) + labs(title = "Distribution of Gender", x = "Gender")



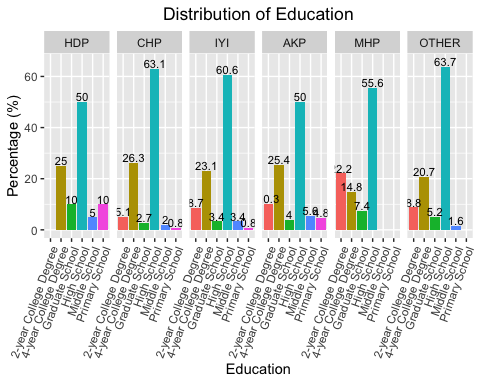
compareParties(survey\_data$age) + labs(title = "Distribution of Age", x = "Age") + theme(axis.text.x = element\_text(angle = 45, hjust = 1));



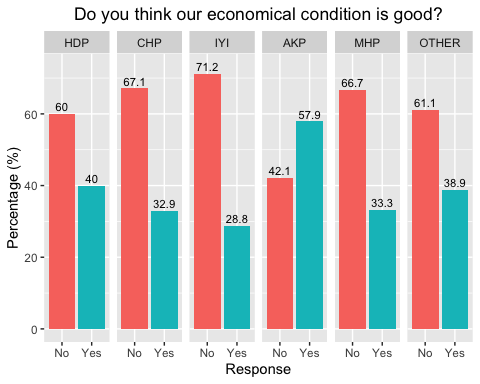
compareParties(survey\_data$region) + labs(title = "Distribution of Region", x = "Region") + theme(axis.text.x = element\_text(angle = 45, hjust = 1));



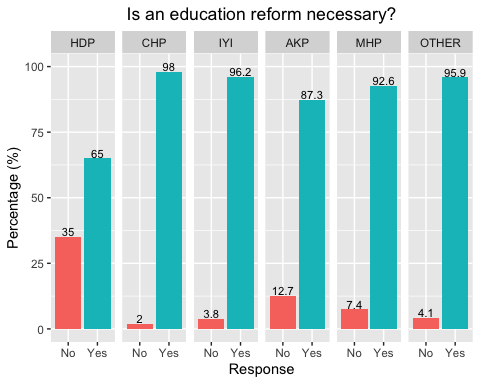
compareParties(survey\_data$education) + labs(title = "Distribution of Education", x = "Education") + theme(axis.text.x = element\_text(angle = 70, hjust = 1));



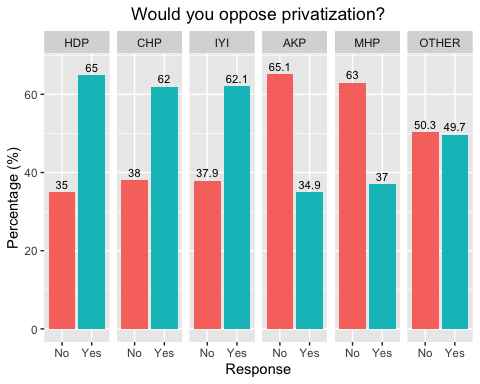
compareParties(survey\_data$economicalSituation) + labs(title = "Do you think our economical condition is good?", x = "Response")



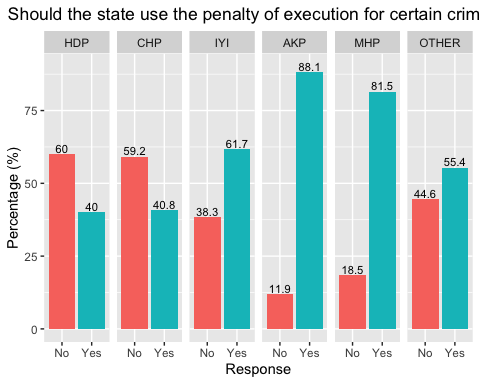
compareParties(survey\_data$educationalReform) + labs(title = "Is an education reform necessary?", x = "Response")



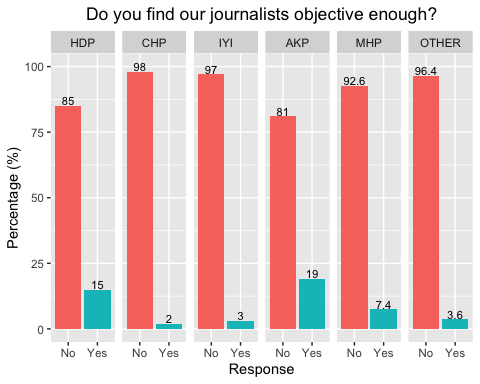
compareParties(survey\_data$againstPrivatization) + labs(title = "Would you oppose privatization?", x = "Response")



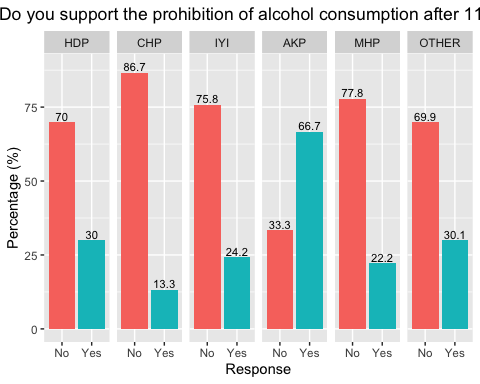
compareParties(survey\_data$execution) + labs(title = "Should the state use the penalty of execution for certain crimes?", x = "Response")



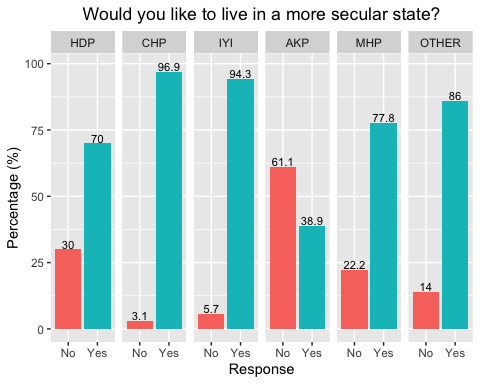
compareParties(survey\_data$objectiveMedia) + labs(title = "Do you find our journalists objective enough?", x = "Response")



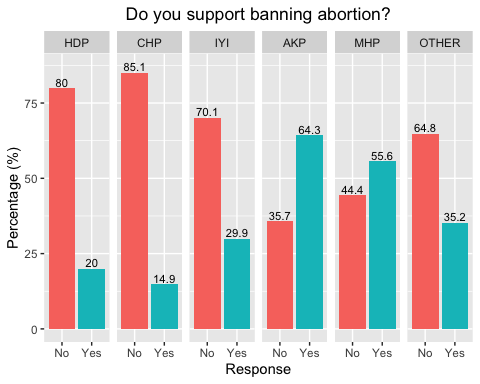
compareParties(survey\_data$alcoholBan) + labs(title = " Do you support the prohibition of alcohol consumption after 11 pm?", x = "Response")



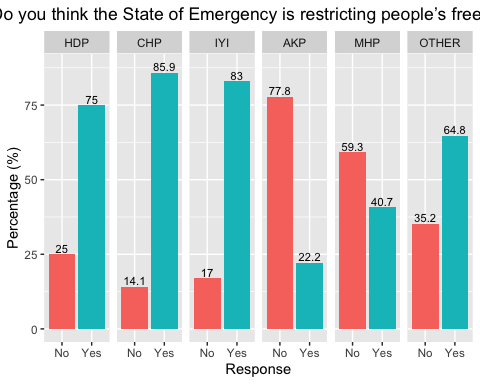
compareParties(survey\_data$secularState) + labs(title = "Would you like to live in a more secular state?", x = "Response")



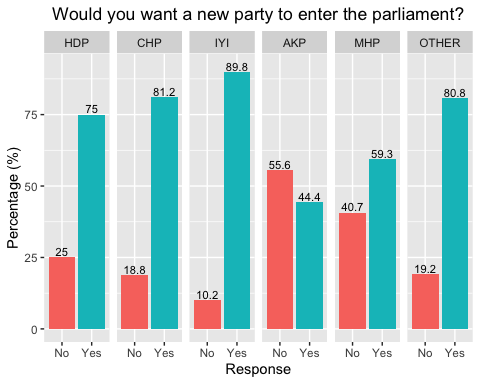
compareParties(survey\_data$abortionBan) + labs(title = "Do you support banning abortion?", x = "Response")



compareParties(survey\_data$stateOfEmergency) + labs(title = "Do you think the State of Emergency is restricting people’s freedom?", x = "Response")



compareParties(survey\_data$newParty) + labs(title = "Would you want a new party to enter the parliament?", x = "Response")



* Logistic Regression & Prediction Model:

# log\_data prepares the survey\_data for logistic regression by assigning 1 for yes and 0 for no for all the question responses  
log\_data <- survey\_data %>%  
 mutate(economicalSituation = ifelse(economicalSituation == "Yes", 1, 0),  
 educationalReform = ifelse(educationalReform == "Yes", 1, 0),  
 againstPrivatization = ifelse(againstPrivatization == "Yes", 1, 0),  
 execution = ifelse(execution == "Yes", 1, 0),  
 objectiveMedia = ifelse(objectiveMedia == "Yes", 1, 0),  
 alcoholBan = ifelse(alcoholBan == "Yes", 1, 0),  
 secularState = ifelse(secularState == "Yes", 1, 0),  
 abortionBan = ifelse(abortionBan == "Yes", 1, 0),  
 stateOfEmergency = ifelse(stateOfEmergency == "Yes", 1, 0),  
 newParty = ifelse(newParty == "Yes", 1, 0))  
  
# selecting the data for AKP voters only  
log\_data\_akp <- log\_data %>%  
 mutate(party = ifelse(party == "AKP", 1, 0)) %>%  
 rename(akpVoter = party)  
  
# creates a prediction model for voting for AKP, using logistic regression   
modelAKP <- glm(akpVoter ~ economicalSituation + educationalReform + againstPrivatization + execution + objectiveMedia +   
 alcoholBan + secularState + abortionBan + stateOfEmergency + newParty, family = binomial(link = "logit"), data = log\_data\_akp)  
summary(modelAKP)

##   
## Call:  
## glm(formula = akpVoter ~ economicalSituation + educationalReform +   
## againstPrivatization + execution + objectiveMedia + alcoholBan +   
## secularState + abortionBan + stateOfEmergency + newParty,   
## family = binomial(link = "logit"), data = log\_data\_akp)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.3034 -0.3763 -0.2255 -0.1498 2.9802   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.3268 0.6304 -2.105 0.03532 \*   
## economicalSituation 0.3933 0.2620 1.501 0.13324   
## educationalReform 0.4726 0.4979 0.949 0.34259   
## againstPrivatization -0.3377 0.2642 -1.278 0.20122   
## execution 0.8834 0.3368 2.623 0.00871 \*\*   
## objectiveMedia 1.3034 0.4307 3.026 0.00247 \*\*   
## alcoholBan 0.8656 0.2687 3.222 0.00127 \*\*   
## secularState -1.4627 0.2773 -5.275 1.32e-07 \*\*\*  
## abortionBan 0.4610 0.2708 1.702 0.08867 .   
## stateOfEmergency -1.4658 0.2709 -5.410 6.30e-08 \*\*\*  
## newParty -0.7017 0.2700 -2.599 0.00934 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 724.37 on 884 degrees of freedom  
## Residual deviance: 433.28 on 874 degrees of freedom  
## AIC: 455.28  
##   
## Number of Fisher Scoring iterations: 6

#coefficients for the model  
modelAKP[1]

## $coefficients  
## (Intercept) economicalSituation educationalReform   
## -1.3268184 0.3933365 0.4725536   
## againstPrivatization execution objectiveMedia   
## -0.3376633 0.8834005 1.3033749   
## alcoholBan secularState abortionBan   
## 0.8655805 -1.4627025 0.4609730   
## stateOfEmergency newParty   
## -1.4658354 -0.7016738

So, our model for voting for AKP is, -1.3268 + 0.3933 \* economicalSituation + 0.4726 \* educationalReform + -0.3377 \* againstPrivatization + 0.8834 \* execution + 1.3034 \* objectiveMedia + 0.8656 \* alcoholBan + -1.4627 \* secularState + 0.4610 \* abortionBan + -1.4658 \* stateOfEmergency + -0.7017 \* newParty.

Let’s calculate the change that someone votes for AKP, given that they think, - our economical situation is not good - we need to have an educational reform - is against privatization - is against execution - does not think media is objective enough - does not support alcohol ban - wants to live in a more secular state - does not support abortion ban - does not support state of emergency - does want a new party to enter the parliement. These are actually my own views, so let’s see what the model will come up with.

# estimates the chances of voting for AKP  
plogis(-1.3268 + 0.3933 \* 0 + 0.4726 \* 1 + -0.3377 \* 1 + 0.8834 \* 0 + 1.3034 \* 0 + 0.8656 \* 0 + -1.4627 \* 1 + 0.4610 \* 0 + -1.4658 \* 1 + -0.7017 \* 1)

## [1] 0.007985582

My chances of voting for AKP, according to our model, is 0.7985582 %.

# see chunk akp\_voter above for more details. the same process is done by using data of CHP voters, and the modelCHP predicts the chances of voting for CHP  
log\_data\_chp <- log\_data %>%  
 mutate(party = ifelse(party == "CHP", 1, 0)) %>%  
 rename(chpVoter = party)  
  
modelCHP <- glm(chpVoter ~ economicalSituation + educationalReform + againstPrivatization + execution + objectiveMedia +   
 alcoholBan + secularState + abortionBan + stateOfEmergency + newParty, family = binomial(link = "logit"), data = log\_data\_chp)  
summary(modelCHP)

##   
## Call:  
## glm(formula = chpVoter ~ economicalSituation + educationalReform +   
## againstPrivatization + execution + objectiveMedia + alcoholBan +   
## secularState + abortionBan + stateOfEmergency + newParty,   
## family = binomial(link = "logit"), data = log\_data\_chp)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.4618 -0.8732 -0.5008 1.1348 2.4344   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.23981 0.65422 -3.424 0.000618 \*\*\*  
## economicalSituation 0.05376 0.17400 0.309 0.757369   
## educationalReform 0.81346 0.51342 1.584 0.113101   
## againstPrivatization 0.28724 0.16729 1.717 0.085985 .   
## execution -0.60489 0.16469 -3.673 0.000240 \*\*\*  
## objectiveMedia -0.77799 0.52200 -1.490 0.136120   
## alcoholBan -0.64039 0.22187 -2.886 0.003898 \*\*   
## secularState 1.01551 0.40474 2.509 0.012106 \*   
## abortionBan -0.86860 0.20923 -4.151 3.3e-05 \*\*\*  
## stateOfEmergency 0.77121 0.22001 3.505 0.000456 \*\*\*  
## newParty -0.54653 0.21904 -2.495 0.012592 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 1062.84 on 884 degrees of freedom  
## Residual deviance: 921.86 on 874 degrees of freedom  
## AIC: 943.86  
##   
## Number of Fisher Scoring iterations: 5

Our model for voting for CHP is, -2.23981 + 0.05376 \* economicalSituation + 0.81346 \* educationalReform + 0.28724 \* againstPrivatization + -0.60489 \* execution + -0.77799 \* objectiveMedia + -0.64039 \* alcoholBan + 1.01551 \* secularState + -0.86860 \* abortionBan + 0.77121 \* stateOfEmergency + -0.54653 \* newParty.

Let’s calculate the chances of me voting for CHP:

plogis(-2.23981 + 0.05376 \* 0 + 0.81346 \* 1 + 0.28724 \* 1 + -0.60489 \* 0 + -0.77799 \* 0 + -0.64039 \* 0 + 1.01551 \* 1 + -0.86860 \* 0 + 0.77121 \* 1 + -0.54653 \* 1)

## [1] 0.5252485

The chances of me voting for CHP is 52.5%.

# see akp\_voter above for comments  
log\_data\_iyi <- log\_data %>%  
 mutate(party = ifelse(party == "IYI PARTI", 1, 0)) %>%  
 rename(iyiVoter = party)  
  
modelIYI <- glm(iyiVoter ~ economicalSituation + educationalReform + againstPrivatization + execution + objectiveMedia +   
 alcoholBan + secularState + abortionBan + stateOfEmergency + newParty, family = binomial(link = "logit"), data = log\_data\_iyi)  
summary(modelIYI)

##   
## Call:  
## glm(formula = iyiVoter ~ economicalSituation + educationalReform +   
## againstPrivatization + execution + objectiveMedia + alcoholBan +   
## secularState + abortionBan + stateOfEmergency + newParty,   
## family = binomial(link = "logit"), data = log\_data\_iyi)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.2560 -0.8885 -0.6421 1.1955 2.4717   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.3655 0.5586 -6.025 1.69e-09 \*\*\*  
## economicalSituation -0.3038 0.1694 -1.793 0.073004 .   
## educationalReform -0.1375 0.4072 -0.338 0.735529   
## againstPrivatization 0.1721 0.1607 1.071 0.284145   
## execution 0.5753 0.1640 3.508 0.000451 \*\*\*  
## objectiveMedia -0.2933 0.4311 -0.680 0.496307   
## alcoholBan 0.0296 0.1929 0.153 0.878045   
## secularState 1.0964 0.3168 3.461 0.000538 \*\*\*  
## abortionBan 0.1956 0.1799 1.087 0.276894   
## stateOfEmergency 0.6845 0.2018 3.393 0.000692 \*\*\*  
## newParty 0.9324 0.2362 3.948 7.89e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 1078.68 on 884 degrees of freedom  
## Residual deviance: 992.28 on 874 degrees of freedom  
## AIC: 1014.3  
##   
## Number of Fisher Scoring iterations: 5

modelIYI[1]

## $coefficients  
## (Intercept) economicalSituation educationalReform   
## -3.36552669 -0.30377034 -0.13754404   
## againstPrivatization execution objectiveMedia   
## 0.17213921 0.57524759 -0.29328171   
## alcoholBan secularState abortionBan   
## 0.02960413 1.09639751 0.19561939   
## stateOfEmergency newParty   
## 0.68448596 0.93237757

Our model for voting for IYI is, -3.3655 + -0.3038 \* economicalSituation + -0.1375 \* educationalReform + 0.1721 \* againstPrivatization + 0.5753 \* execution + -0.2933 \* objectiveMedia + 0.0296 \* alcoholBan + 1.0964 \* secularState + 0.1956 \* abortionBan + 0.6845 \* stateOfEmergency + 0.9324 \* newParty.

plogis(-3.3655 + -0.3038 \* 0 + -0.1375 \* 1 + 0.1721 \* 1 + 0.5753 \* 0 + -0.2933 \* 0 + 0.0296 \* 0 + 1.0964 \* 1 + 0.1956 \* 0 + 0.6845 \* 1 + 0.9324 \* 1)

## [1] 0.3503275

The chances of me voting for IYI is 35.0% according to our model.

# see akp\_voter above for comments  
log\_data\_mhp <- log\_data %>%  
 mutate(party = ifelse(party == "MHP", 1, 0)) %>%  
 rename(mhpVoter = party)  
  
modelMHP <- glm(mhpVoter ~ economicalSituation + educationalReform + againstPrivatization + execution + objectiveMedia +   
 alcoholBan + secularState + abortionBan + stateOfEmergency + newParty, family = binomial(link = "logit"), data = log\_data\_mhp)  
summary(modelMHP)

##   
## Call:  
## glm(formula = mhpVoter ~ economicalSituation + educationalReform +   
## againstPrivatization + execution + objectiveMedia + alcoholBan +   
## secularState + abortionBan + stateOfEmergency + newParty,   
## family = binomial(link = "logit"), data = log\_data\_mhp)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.7803 -0.2622 -0.1908 -0.1220 3.1355   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.96928 1.06917 -2.777 0.00548 \*\*  
## economicalSituation -0.42112 0.44444 -0.948 0.34337   
## educationalReform -0.07761 0.79286 -0.098 0.92202   
## againstPrivatization -0.60740 0.42792 -1.419 0.15577   
## execution 0.91140 0.52776 1.727 0.08418 .   
## objectiveMedia -0.12832 0.79906 -0.161 0.87242   
## alcoholBan -1.08545 0.54335 -1.998 0.04575 \*   
## secularState 0.28218 0.57328 0.492 0.62256   
## abortionBan 0.82007 0.43291 1.894 0.05818 .   
## stateOfEmergency -1.07898 0.44824 -2.407 0.01608 \*   
## newParty -0.45740 0.44989 -1.017 0.30929   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 241.61 on 884 degrees of freedom  
## Residual deviance: 215.45 on 874 degrees of freedom  
## AIC: 237.45  
##   
## Number of Fisher Scoring iterations: 7

Our model for voting for MHP is, -2.96927959 + -0.42112092\* economicalSituation + -0.07761032 \* educationalReform + -0.60740164 \* againstPrivatization + 0.91139984 \* execution + -0.12832052 \* objectiveMedia + -1.08544652 \* alcoholBan + 0.28217820 \* secularState + 0.82007304 \* abortionBan + -1.07897657 \* stateOfEmergency + -0.45740433 \* newParty.

plogis(-2.96927959 + -0.42112092\* 0 + -0.07761032 \* 1 + -0.60740164 \* 1 + 0.91139984 \* 0 + -0.12832052 \* 0 + -1.08544652 \* 0 + 0.28217820 \* 1 + 0.82007304 \* 0 + -1.07897657 \* 1 + -0.45740433 \* 1)

## [1] 0.00732948

The chances of me voting for MHP is 0.732948% according to our model.

# see akp\_voter above for comments  
log\_data\_hdp <- log\_data %>%  
 mutate(party = ifelse(party == "HDP", 1, 0)) %>%  
 rename(hdpVoter = party)  
  
modelHDP <- glm(hdpVoter ~ economicalSituation + educationalReform + againstPrivatization + execution + objectiveMedia +   
 alcoholBan + secularState + abortionBan + stateOfEmergency + newParty, family = binomial(link = "logit"), data = log\_data\_hdp)  
summary(modelHDP)

##   
## Call:  
## glm(formula = hdpVoter ~ economicalSituation + educationalReform +   
## againstPrivatization + execution + objectiveMedia + alcoholBan +   
## secularState + abortionBan + stateOfEmergency + newParty,   
## family = binomial(link = "logit"), data = log\_data\_hdp)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.2760 -0.2061 -0.1613 -0.1016 2.9736   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.96130 0.93620 -1.027 0.3045   
## economicalSituation -0.13197 0.52090 -0.253 0.8000   
## educationalReform -2.64712 0.61423 -4.310 1.63e-05 \*\*\*  
## againstPrivatization 0.98631 0.56456 1.747 0.0806 .   
## execution -1.05927 0.53795 -1.969 0.0489 \*   
## objectiveMedia 0.41089 0.78168 0.526 0.5991   
## alcoholBan 0.03229 0.63105 0.051 0.9592   
## secularState -1.37492 0.71490 -1.923 0.0545 .   
## abortionBan -0.62313 0.64206 -0.971 0.3318   
## stateOfEmergency 0.66240 0.68206 0.971 0.3315   
## newParty 0.11683 0.68789 0.170 0.8651   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 191.14 on 884 degrees of freedom  
## Residual deviance: 160.65 on 874 degrees of freedom  
## AIC: 182.65  
##   
## Number of Fisher Scoring iterations: 7

modelHDP[1]

## $coefficients  
## (Intercept) economicalSituation educationalReform   
## -0.96129803 -0.13196787 -2.64712350   
## againstPrivatization execution objectiveMedia   
## 0.98630771 -1.05927191 0.41088629   
## alcoholBan secularState abortionBan   
## 0.03229371 -1.37492229 -0.62312652   
## stateOfEmergency newParty   
## 0.66240439 0.11682536

Our model for voting for HDP is, -0.96129803 + -0.13196787 \* economicalSituation + -2.64712350 \* educationalReform + 0.98630771 \* againstPrivatization + -1.05927191 \* execution + 0.41088629 \* objectiveMedia + 0.03229371 \* alcoholBan + -1.37492229 \* secularState + -0.62312652 \* abortionBan + 0.66240439 \* stateOfEmergency + 0.11682536 \* newParty.

plogis(-0.96129803 + -0.13196787 \* 0 + -2.64712350 \* 1 + 0.98630771 \* 1 + -1.05927191 \* 0 + 0.41088629 \* 0 + 0.03229371 \* 0 + -1.37492229 \* 1 + -0.62312652 \* 0 + 0.66240439 \* 1 + 0.11682536 \* 1)

## [1] 0.03850111

The chances of me voting for HDP is 3.850111% according to our model.

# see akp\_voter above for comments  
log\_data\_other <- log\_data %>%  
 mutate(party = ifelse(party == "DIĞER", 1, 0)) %>%  
 rename(otherVoter = party)  
  
modelOther <- glm(otherVoter ~ economicalSituation + educationalReform + againstPrivatization + execution + objectiveMedia +   
 alcoholBan + secularState + abortionBan + stateOfEmergency + newParty, family = binomial(link = "logit"), data = log\_data\_other)  
summary(modelOther)

##   
## Call:  
## glm(formula = otherVoter ~ economicalSituation + educationalReform +   
## againstPrivatization + execution + objectiveMedia + alcoholBan +   
## secularState + abortionBan + stateOfEmergency + newParty,   
## family = binomial(link = "logit"), data = log\_data\_other)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.0285 -0.7367 -0.6535 -0.5209 2.2766   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.7109 0.5117 -3.343 0.000828 \*\*\*  
## economicalSituation 0.1025 0.1758 0.583 0.559841   
## educationalReform 0.2958 0.4163 0.711 0.477326   
## againstPrivatization -0.2994 0.1708 -1.752 0.079738 .   
## execution -0.1983 0.1779 -1.115 0.264878   
## objectiveMedia -0.4914 0.4337 -1.133 0.257201   
## alcoholBan 0.1500 0.2014 0.745 0.456441   
## secularState 0.3837 0.2778 1.381 0.167203   
## abortionBan 0.2227 0.1896 1.175 0.240090   
## stateOfEmergency -0.4499 0.1961 -2.294 0.021816 \*   
## newParty 0.3455 0.2239 1.543 0.122871   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 928.30 on 884 degrees of freedom  
## Residual deviance: 912.12 on 874 degrees of freedom  
## AIC: 934.12  
##   
## Number of Fisher Scoring iterations: 4

modelOther[1]

## $coefficients  
## (Intercept) economicalSituation educationalReform   
## -1.7109071 0.1024852 0.2957923   
## againstPrivatization execution objectiveMedia   
## -0.2993574 -0.1983104 -0.4913799   
## alcoholBan secularState abortionBan   
## 0.1499936 0.3836609 0.2227080   
## stateOfEmergency newParty   
## -0.4498599 0.3454827

Our model for voting for other parties is, -1.7109071 + 0.1024852 \* economicalSituation + 0.2957923 \* educationalReform + -0.2993574 \* againstPrivatization + -0.1983104 \* execution + -0.4913799 \* objectiveMedia + 0.1499936 \* alcoholBan + 0.3836609 \* secularState + 0.2227080 \* abortionBan + -0.4498599 \* stateOfEmergency + 0.3454827 \* newParty.

plogis(-1.7109071 + 0.1024852 \* 0 + 0.2957923 \* 1 + -0.2993574 \* 1 + -0.1983104 \* 0 + -0.4913799 \* 0 + 0.1499936 \* 0 + 0.3836609 \* 1 + 0.2227080 \* 0 + -0.4498599 \* 1 + 0.3454827 \* 1)

## [1] 0.1922915

The chances of me voting for another party is 19.22915%.

So, according to the models that I have built, they estimate my chances of voting for parties as: CHP - 52.5%. IYI - 35.0%. OTHER - 19.22915%. HDP - 3.850111. AKP - 0.7985582%. MHP - 0.732948%.

* Trying using multinom():

# this chunk is an effort to use multinom() to come up with the prediction model. It is not used since I was not sure about how the interpret the coefficients, given that the coefficients were calculated with reference to the first group.  
corr\_data <- log\_data %>%  
 mutate(party = ifelse(party == "AKP", 1, party)) %>%  
 mutate(party = ifelse(party == "CHP", 2, party)) %>%  
 mutate(party = ifelse(party == "MHP", 3, party)) %>%  
 mutate(party = ifelse(party == "IYI PARTI", 4, party)) %>%  
 mutate(party = ifelse(party == "HDP", 5, party)) %>%  
 mutate(party = ifelse(party == "DIĞER", 6, party)) %>%  
 dplyr::select(-c("Timestamp", "gender", "age", "region", "education", "party\_factor")) %>%  
 mutate(party = as.numeric(party))  
  
model <- multinom(party ~ economicalSituation + educationalReform + againstPrivatization + execution + objectiveMedia +   
 alcoholBan + secularState + abortionBan + stateOfEmergency + newParty, data = corr\_data)

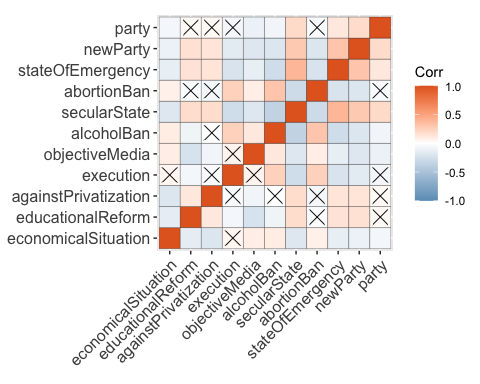
## # weights: 72 (55 variable)  
## initial value 1585.707130   
## iter 10 value 1154.202335  
## iter 20 value 1114.174302  
## iter 30 value 1106.911079  
## iter 40 value 1106.476033  
## final value 1106.474573   
## converged

summary(model)

## Call:  
## multinom(formula = party ~ economicalSituation + educationalReform +   
## againstPrivatization + execution + objectiveMedia + alcoholBan +   
## secularState + abortionBan + stateOfEmergency + newParty,   
## data = corr\_data)  
##   
## Coefficients:  
## (Intercept) economicalSituation educationalReform againstPrivatization  
## 2 -0.7930319 -0.3476930 0.2605410 0.6303945  
## 3 -1.1045577 -0.6681295 -0.3430432 -0.2944262  
## 4 -1.2982361 -0.5710842 -0.4132863 0.5300460  
## 5 0.6715293 -0.4482343 -2.7876221 1.3412754  
## 6 0.1518052 -0.2432243 -0.1226832 0.1327438  
## execution objectiveMedia alcoholBan secularState abortionBan  
## 2 -1.23530395 -1.8346308 -1.2826214 1.99317346 -1.1737775  
## 3 0.07148116 -0.8439904 -1.5332542 1.09958184 0.3526162  
## 4 -0.46662365 -1.3776233 -0.7805669 1.92541267 -0.4024201  
## 5 -1.83883077 -0.5320462 -0.6226423 -0.06031889 -1.0105919  
## 6 -0.96158046 -1.3638955 -0.6086379 1.20934636 -0.2967191  
## stateOfEmergency newParty  
## 2 1.9982942 0.29357761  
## 3 0.2950313 0.07768353  
## 4 1.8554962 1.28120393  
## 5 1.7996675 0.56151499  
## 6 1.0157009 0.76326185  
##   
## Std. Errors:  
## (Intercept) economicalSituation educationalReform againstPrivatization  
## 2 0.8802075 0.2993381 0.6875087 0.2992292  
## 3 1.1444895 0.4770744 0.8632685 0.4739535  
## 4 0.8012084 0.2920804 0.6004557 0.2908718  
## 5 1.0177980 0.5616133 0.7414443 0.6015334  
## 6 0.7203874 0.2849670 0.5682131 0.2868441  
## execution objectiveMedia alcoholBan secularState abortionBan stateOfEmergency  
## 2 0.3608273 0.6383460 0.3290022 0.4531286 0.3244878 0.3267054  
## 3 0.6047616 0.8145813 0.5434671 0.5541063 0.4754053 0.4869215  
## 4 0.3600193 0.5470384 0.3040721 0.3736435 0.3008278 0.3076453  
## 5 0.6034393 0.8194930 0.6294858 0.6823585 0.6592465 0.6563616  
## 6 0.3562837 0.5233958 0.2965413 0.3208834 0.2958571 0.2982296  
## newParty  
## 2 0.3238987  
## 3 0.4777314  
## 4 0.3302209  
## 5 0.6734378  
## 6 0.3035277  
##   
## Residual Deviance: 2212.949   
## AIC: 2322.949

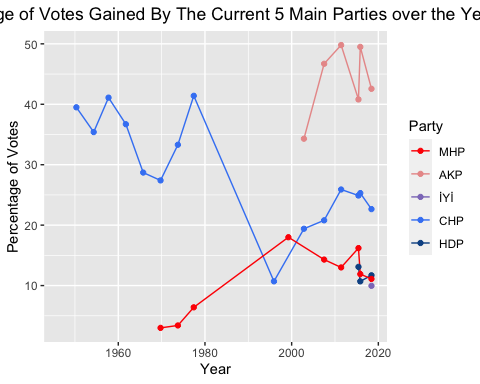
* Correlation Matrix:

# this chunk produces a correlation matrix, where the correlations between variables that are not statistically significant have an X mark  
corr <- cor(corr\_data)  
pmat <- cor\_pmat(corr\_data)  
ggcorrplot(corr, outline.color = "black", ggtheme = ggplot2::theme\_gray,  
 colors = c("#6D9EC1", "white", "#E46726"), p.mat = pmat)

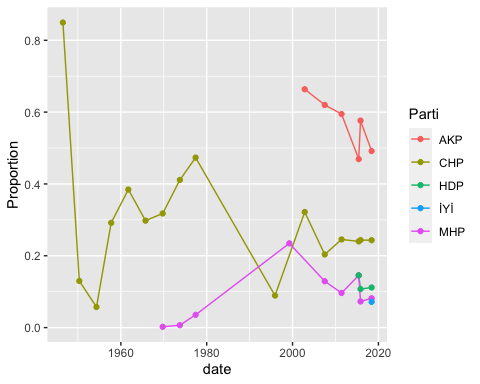


* Votes for Current 5 Main Parties:

names(general\_election\_results)[2] <- 'percentage' # I experienced that having the column name as % was causing problems, so here I am changing it to percentage  
  
general\_election\_results <- general\_election\_results %>%  
 mutate(date = make\_date(Year, Month), # creating a new variable called date based on the year and month columns  
 percentage = gsub( "%", "", percentage), # getting rid of the percentage sign in the beginning of some values in the percentage column  
 percentage = as.numeric(gsub( ",", ".", percentage))) # making percentage numeric and changing , to .   
  
#selecting the top 5 current main parties, graphing Percentage of Votes Gained By The Current 5 Main Parties over the Years  
general\_election\_results %>%  
 filter(Parti %in% c("AKP", "CHP", "MHP", "HDP", "İYİ")) %>%  
 ggplot() +   
 geom\_line(aes(x = date, y = percentage, color = Parti)) +   
 geom\_point(aes(x = date, y = percentage, color = Parti)) +  
 scale\_color\_manual(values = c("MHP" = "#ff0100", "AKP" = "#e99a99", "İYİ" = "#8e7cc3", "CHP" = "#4285f4", "HDP" = "#0c5394")) +   
 labs(color = "Party", x = "Year", y = "Percentage of Votes", title = "Percentage of Votes Gained By The Current 5 Main Parties over the Years") + center

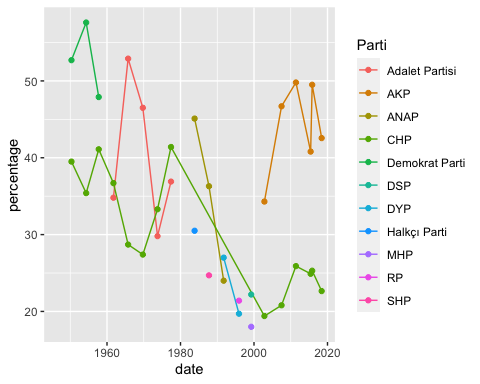


# graphs the proportion of the seats in the parliement over the years, not included in the presentation  
general\_election\_results %>%  
 filter(Parti %in% c("AKP", "CHP", "MHP", "HDP", "İYİ")) %>%  
 ggplot() +   
 geom\_line(aes(x = date, y = Proportion, color = Parti)) +   
 geom\_point(aes(x = date, y = Proportion, color = Parti))

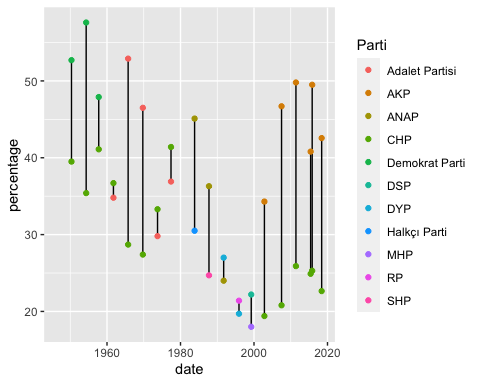


* Top Two Parties Voted over Time:

# top two parties voted over time, not included in the presentation  
general\_election\_results %>%  
 group\_by(date) %>%  
 arrange(desc(percentage)) %>%  
 slice(1:2) %>%  
 ggplot() +   
 geom\_line(aes(x = date, y = percentage, color = Parti)) +   
 geom\_point(aes(x = date, y = percentage, color = Parti))

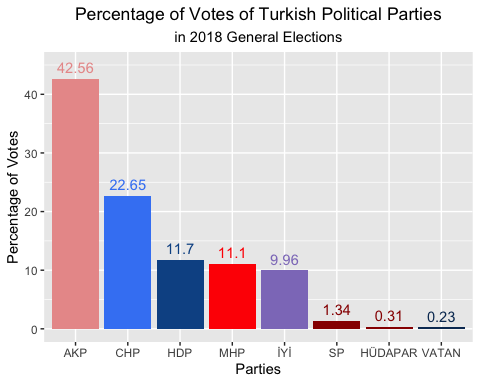


# top two parties voted over time, not included in the presentation, I just wanted to take a look at the gaps between them and how they are changing  
general\_election\_results %>%  
 group\_by(date) %>%  
 arrange(desc(percentage)) %>%  
 slice(1:2) %>%  
 ggplot() +   
 geom\_line(aes(x = date, y = percentage, group = date)) +   
 geom\_point(aes(x = date, y = percentage, color = Parti))

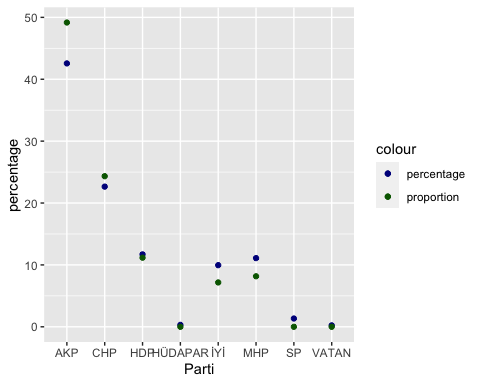


* Most Recent Election Results:

# setting the colors that I will be using in the graph  
colors\_x <- c("#e99a99", "#4285f4", "#0c5394", "#ff0100", "#8e7cc3", "#980100", "#980100", "#093762")  
  
# graphing the percentages of votes of Turkish political parties in the 2018 election  
general\_election\_results %>%  
 filter(Year == 2018) %>%  
 dplyr::select(Parti, percentage, Proportion) %>%  
 ggplot() +   
 geom\_col(aes(x = reorder(Parti, -percentage), y = percentage), fill = colors\_x) +   
 geom\_text(aes(x = Parti, y = percentage, label = percentage), vjust = -0.6, color = colors\_x) +   
 ylim(0, 45) +   
 guides(fill = "none") +   
 labs(x = "Parties", y = "Percentage of Votes", title = "Percentage of Votes of Turkish Political Parties", subtitle = "in 2018 General Elections") + center + center\_sub

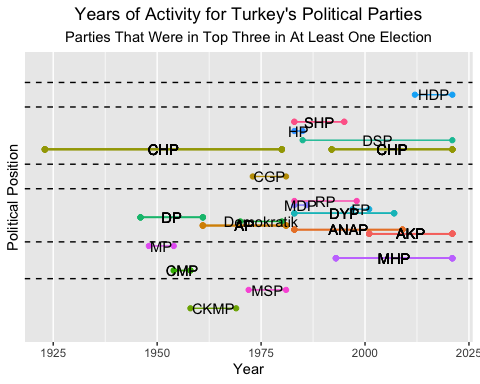


# looking at percentage of votes vs. proportion of seats in the parliament for the 2018 election, not included in the presentation   
general\_election\_results %>%  
 filter(Year == 2018) %>%  
 dplyr::select(Parti, percentage, Proportion) %>%  
 ggplot() +   
 geom\_point(aes(x = Parti, y = percentage, color = "percentage")) +   
 geom\_point(aes(x = Parti, y = Proportion \* 100, color = "proportion")) +   
 scale\_color\_manual(breaks = c("percentage", "proportion"),  
 values = c("darkblue", "darkgreen"))

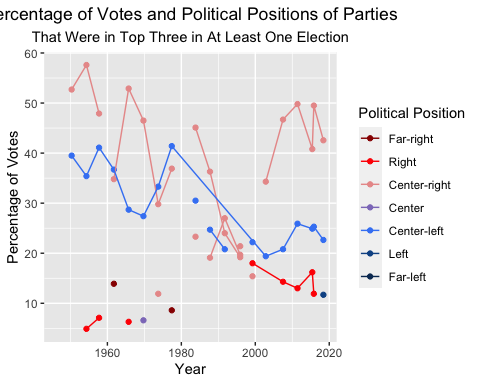


* Top Three Parties:

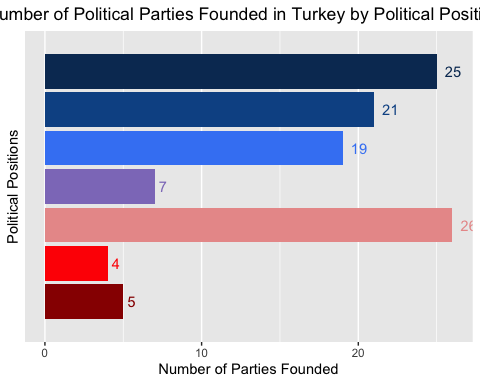
# here I am trying to make a dataset of the parties that were in the top three in at least one election  
top\_three <- general\_election\_results %>%  
 mutate(MP = as.integer(MP), # I had some problems with the variable types before, so here I am trying to make sure they all have the type that I want  
 TotalMP = as.integer(TotalMP),  
 Month = as.integer(Month),  
 Year = as.integer(Year)) %>%  
 left\_join(political\_chronology, by = "Parti") %>%  
 mutate(Founded = as.integer(Founded), # I had some problems with the variable types before, so here I am trying to make sure they all have the type that I want  
 Dissolved = as.integer(Dissolved),  
 percentage = as.double(percentage)) %>%  
 mutate(Founded = ifelse((Name == "Milliyetçi Hareket Partisi" & date <= 1980), 1969, Founded), # some parties were founded twice. so I am adjusting the years founded-dissolved based on this  
 Dissolved = ifelse((Name == "Milliyetçi Hareket Partisi" & date <= 1980), 1980, Dissolved),  
 Founded = ifelse((Name == "Milliyetçi Hareket Partisi" & date > 1980), 1993, Founded),  
 Dissolved = ifelse((Name == "Milliyetçi Hareket Partisi" & date > 1980), 2021, Dissolved),  
 Founded = ifelse((Name == "Cumhuriyet Halk Partisi" & date <= 1980), 1923, Founded),  
 Dissolved = ifelse((Name == "Cumhuriyet Halk Partisi" & date <= 1980), 1980, Dissolved),  
 Founded = ifelse((Name == "Cumhuriyet Halk Partisi" & date >= 1992), 1992, Founded),  
 Dissolved = ifelse((Name == "Cumhuriyet Halk Partisi" & date >= 1992), 2021, Dissolved),  
 Founded = ifelse((Name == "Demokrat Parti" & date <= 1961), 1946, Founded),  
 Dissolved = ifelse((Name == "Demokrat Parti" & date <= 1961), 1961, Dissolved),  
 Founded = ifelse((Name == "Demokrat Parti" & date > 1961), 2007, Founded),  
 Dissolved = ifelse((Name == "Demokrat Parti" & date > 1961), 2021, Dissolved),  
 Founded = ifelse((Name == "Adalet Partisi" & date <= 1961), 1946, Founded),  
 Dissolved = ifelse((Name == "Adalet Partisi" & date <= 1981), 1981, Dissolved),  
 Founded = ifelse((Name == "Adalet Partisi" & date > 1981), 2015, Founded),  
 Dissolved = ifelse((Name == "Adalet Partisi" & date > 1981), 2021, Dissolved),  
 Founded = ifelse((Parti == "HDP"), 2012, Founded), # for some reason didn't have this info for HDP  
 Dissolved = ifelse((Parti == "HDP"), 2021, Dissolved),  
 Founded = as.integer(Founded),  
 Dissolved = as.integer(Dissolved)) %>%  
 group\_by(date) %>%  
 mutate(percentage = as.double(percentage)) %>%  
 arrange(desc(percentage)) %>%  
 slice(1:3) %>% # getting only the top three  
 ungroup() %>%  
 filter(! Parti == "Bağımsız") %>% #if there was an independent candidate in the top three, I am not listing them since I don't know how to decide on their political positions or ideologies.   
   
 # for some reason the political chronologies dataset did not have information for these parties below. so I searched for them on Wikipedia and manually updated the variables.  
 mutate( Name = ifelse(Parti == "Cumhuriyetçi Millet Partisi", "Cumhuriyetçi Millet Partisi", Name),  
 Leader = ifelse(Parti == "Cumhuriyetçi Millet Partisi", "Osman Bölükbaşı", Leader),  
 Founded = ifelse(Parti == "Cumhuriyetçi Millet Partisi", 1954, Founded),  
 Dissolved = ifelse(Parti == "Cumhuriyetçi Millet Partisi", 1958, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Cumhuriyetçi Millet Partisi", 2, PoliticalPosition),  
 Ideology = ifelse(Parti == "Cumhuriyetçi Millet Partisi", "Turkish nationalism", Ideology),  
 Parti = ifelse(Parti == "Cumhuriyetçi Millet Partisi", "CMP", Parti),  
   
 Name = ifelse(Parti == "Cumhuriyetçi Köylü Millet Partisi", "Cumhuriyetçi Köylü Millet Partisi", Name),  
 Leader = ifelse(Parti == "Cumhuriyetçi Köylü Millet Partisi", "Osman Bölükbaşı", Leader),  
 Founded = ifelse(Parti == "Cumhuriyetçi Köylü Millet Partisi", 1958, Founded),  
 Dissolved = ifelse(Parti == "Cumhuriyetçi Köylü Millet Partisi", 1969, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Cumhuriyetçi Köylü Millet Partisi", 1, PoliticalPosition),  
 Ideology = ifelse(Parti == "Cumhuriyetçi Köylü Millet Partisi", "Turkish nationalism", Ideology),  
 Parti = ifelse(Parti == "Cumhuriyetçi Köylü Millet Partisi", "CKMP", Parti),  
   
 Name = ifelse(Parti == "Yeni Türkiye Partisi", "Yeni Türkiye Partisi", Name),  
 Leader = ifelse(Parti == "Yeni Türkiye Partisi", "Ekrem Alican", Leader),  
 Founded = ifelse(Parti == "Yeni Türkiye Partisi", 1961, Founded),  
 Dissolved = ifelse(Parti == "Yeni Türkiye Partisi", 1973, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Yeni Türkiye Partisi", 3, PoliticalPosition),  
 Ideology = ifelse(Parti == "Yeni Türkiye Partisi", "Liberal conservatism", Ideology),  
 Parti = ifelse(Parti == "Yeni Türkiye Partisi", "YTP", Parti),  
   
 Name = ifelse(Parti == "Millet Partisi", "Millet Partisi", Name),  
 Leader = ifelse(Parti == "Millet Partisi", "Fevzi Çakmak", Leader),  
 Founded = ifelse(Parti == "Millet Partisi", 1948, Founded),  
 Dissolved = ifelse(Parti == "Millet Partisi", 1954, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Millet Partisi", 2, PoliticalPosition),  
 Ideology = ifelse(Parti == "Millet Partisi", "Turkish nationalism, conservatism", Ideology),  
 Parti = ifelse(Parti == "Millet Partisi", "MP", Parti),  
   
 Name = ifelse(Parti == "Demokratik Parti", "Demokratik Parti", Name),  
 Leader = ifelse(Parti == "Demokratik Parti", "Ferruh Bozbeyli", Leader),  
 Founded = ifelse(Parti == "Demokratik Parti", 1970, Founded),  
 Dissolved = ifelse(Parti == "Demokratik Parti", 1980, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Demokratik Parti", 3, PoliticalPosition),  
 Ideology = ifelse(Parti == "Demokratik Parti", "Conservatism, Liberalism, Conservative liberalism, Liberal conservatism", Ideology),  
 Parti = ifelse(Parti == "Demokratik Parti", "Demokratik", Parti),  
   
 Name = ifelse(Parti == "Halkçı Parti", "Halkçı Parti", Name),  
 Leader = ifelse(Parti == "Halkçı Parti", "Necdet Calp", Leader),  
 Founded = ifelse(Parti == "Halkçı Parti", 1983, Founded),  
 Dissolved = ifelse(Parti == "Halkçı Parti", 1985, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Halkçı Parti", 5, PoliticalPosition),  
 Ideology = ifelse(Parti == "Halkçı Parti", "Social democracy, Secularism", Ideology),  
 Parti = ifelse(Parti == "Halkçı Parti", "HP", Parti),  
   
 Name = ifelse(Parti == "MDP", "Milliyetçi Demokrasi Partisi", Name),  
 Leader = ifelse(Parti == "MDP", "Turgut Sunalp", Leader),  
 Founded = ifelse(Parti == "MDP", 1983, Founded),  
 Dissolved = ifelse(Parti == "MDP", 1986, Dissolved),  
 PoliticalPosition = ifelse(Parti == "MDP", 3, PoliticalPosition),  
 Ideology = ifelse(Parti == "MDP", "Nationalism", Ideology),  
  
 Name = ifelse(Parti == "Demokrat Parti", "Demokrat Parti", Name),  
 Leader = ifelse(Parti == "Demokrat Parti", "Celal Bayar", Leader),  
 Founded = ifelse(Parti == "Demokrat Parti", 1946, Founded),  
 Dissolved = ifelse(Parti == "Demokrat Parti", 1961, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Demokrat Parti", 3, PoliticalPosition),  
 Ideology = ifelse(Parti == "Demokrat Parti", "Liberal conservatism, Economic liberalism, Right-wing populism, Ataturkism", Ideology),  
 Parti = ifelse(Parti == "Demokrat Parti", "DP", Parti),  
  
 Name = ifelse(Parti == "Adalet Partisi", "Adalet Partisi", Name),  
 Leader = ifelse(Parti == "Adalet Partisi", "Süleyman Demirel", Leader),  
 Founded = ifelse(Parti == "Adalet Partisi", 1961, Founded),  
 Dissolved = ifelse(Parti == "Adalet Partisi", 1981, Dissolved),  
 PoliticalPosition = ifelse(Parti == "Adalet Partisi", 3, PoliticalPosition),  
 Ideology = ifelse(Parti == "Adalet Partisi", "Liberal conservatism, Economic liberalism, Right-wing populism", Ideology),  
 Parti = ifelse(Parti == "Adalet Partisi", "AP", Parti))  
  
#making a vector for the colors that I will be using  
colors <- c("#980100", "#ff0100", "#e99a99", "#8e7cc3", "#4285f4", "#0c5394", "#093762")  
  
# graphing the years of activity for the parties that were in the top three in at least one election  
top\_three %>%  
 ggplot() +   
 geom\_point(aes(x = Founded, y = PoliticalPosition, color = Parti), position = position\_dodge(0.9)) + # I am adding these dodges so that they are not overlapping  
 geom\_point(aes(x = Dissolved, y = PoliticalPosition, color = Parti), position = position\_dodge(0.9)) +   
 geom\_linerangeh(aes(xmin = Founded, xmax = Dissolved, y = PoliticalPosition, color = Parti), position = position\_dodge(0.9)) +   
 geom\_text(aes(x = (Founded+Dissolved)/2 , y = PoliticalPosition, label = Parti, group = Parti), position = position\_dodge(0.9)) +   
 geom\_hline(yintercept = 1.5, linetype = "dashed") +  
 geom\_hline(yintercept = 2.4, linetype = "dashed") +  
 geom\_hline(yintercept = 3.7, linetype = "dashed") +  
 geom\_hline(yintercept = 4.3, linetype = "dashed") +  
 geom\_hline(yintercept = 5.7, linetype = "dashed") +  
 geom\_hline(yintercept = 6.3, linetype = "dashed") +  
 guides(color = "none") +   
 scale\_y\_discrete(labels = c("Far-right", "Right", "Center-right", "Center", "Center-left", "Left")) +   
 labs(x = "Year", y = "Political Position", title = "Years of Activity for Turkey's Political Parties", subtitle = "Parties That Were in Top Three in At Least One Election") +   
 center + center\_sub +   
 theme(axis.text.y = element\_text(color = colors))



# For this visualization I am not listing the names of the parties (could optionally be done with a simple geom\_text, but it's not what I am looking for here)  
# What I am trying to do is to see the percentage of votes and political positions of the parties that were in the top three in at least one election  
top\_three %>%  
 mutate(PoliticalPosition = case\_when(  
 PoliticalPosition == "1" ~ "Far-right",   
 PoliticalPosition == "2" ~ "Right",  
 PoliticalPosition == "3" ~ "Center-right",  
 PoliticalPosition == "4" ~ "Center",  
 PoliticalPosition == "5" ~ "Center-left",  
 PoliticalPosition == "6" ~ "Left",  
 PoliticalPosition == "7" ~"Far-left")) %>%  
 ggplot() +   
 geom\_point(aes(y = percentage, x = date, color = PoliticalPosition)) +  
 geom\_line(aes(y = percentage, x = date, group = Parti, color = PoliticalPosition)) +   
 scale\_color\_manual(values = c("Far-right" = "#980100", "Right" = "#ff0100", "Center-right" = "#e99a99", "Center" = "#8e7cc3", "Center-left" = "#4285f4", "Left" = "#0c5394", "Far-left" = "#093762")) +   
 labs(x = "Year", y = "Percentage of Votes", title = "Percentage of Votes and Political Positions of Parties", subtitle = "That Were in Top Three in At Least One Election", color = "Political Position") +   
 center + center\_sub



# making a graph of number of parties founded by their political position  
political\_chronology %>%  
 group\_by(PoliticalPosition) %>%  
 summarize(count = n()) %>%  
 ggplot() + geom\_col(aes(x = count, y = PoliticalPosition), fill = colors) +   
 scale\_y\_discrete(labels = c("Far-right", "Right", "Center-right", "Center", "Center-left", "Left", "Far-left")) +  
 geom\_text(aes(x = count, y = PoliticalPosition, label = count), color = colors, hjust = -0.5) +   
 labs (x = "Number of Parties Founded", y = "Political Positions", title = "Number of Political Parties Founded in Turkey by Political Positions") +   
 center



* Word Clouds:

# here I have all the party names  
party\_names <- political\_chronology %>%  
 full\_join(general\_election\_results, by = "Parti") %>%  
 dplyr::select(Name, Parti) %>%  
 pivot\_longer(Name:Parti) %>%  
 dplyr::select(value) %>%  
 distinct() %>%  
 mutate(value = ifelse(value == "İYİ", "İyi Parti", value),  
 value = ifelse(value == "VATAN", "Vatan Partisi", value),  
 value = ifelse(value == "MDP", "Milliyetçi Demokrasi Partisi", value),  
 value = ifelse(value == "TBP", "Türkiye Birlik Partisi", value)) %>%  
 filter(nchar(value) > 8) %>%  
 filter(! value %in% c("YÜCEDİRİ-P", "HALKINKURTULUŞU", "MİLLET1948", "Demokrat Parti 1964"))  
  
# finding the most commonly used words in party names  
data\_frame(text = party\_names) %>%   
 mutate\_all(as.character) %>%  
 unnest\_tokens(word, text) %>% # split words  
 count(word, sort = TRUE)

## # A tibble: 107 × 2  
## word n  
## <chr> <int>  
## 1 partisi 9020  
## 2 türkiye 2860  
## 3 parti 1760  
## 4 sosyalist 1430  
## 5 ve 1320  
## 6 devrimci 990  
## 7 i̇şçi 990  
## 8 komünist 990  
## 9 demokrasi 770  
## 10 halkın 660  
## # … with 97 more rows

# making a wordcloud of the most commonly used words in party names (not shown in the presentation)   
wordcloud(party\_names$value, min.freq = 3, max.words = 100, random.order = FALSE, colors=brewer.pal(8, "Dark2"))



# here I have all the ideologies  
ideologies <- political\_chronology %>%  
 mutate(Ideology = strsplit(as.character(Ideology), ",")) %>%  
 unnest(Ideology) %>%  
 dplyr::select(Ideology)  
  
# finding the most commonly used words in ideologies  
data\_frame(text = ideologies) %>%   
 mutate\_all(as.character) %>%  
 unnest\_tokens(word, text) %>% # split words  
 count(word, sort = TRUE)

## # A tibble: 49 × 2  
## word n  
## <chr> <int>  
## 1 nationalism 7590  
## 2 marxism 6210  
## 3 leninism 5520  
## 4 socialism 5520  
## 5 conservatism 4830  
## 6 liberalism 4600  
## 7 social 3220  
## 8 turkish 2990  
## 9 democracy 2760  
## 10 islamism 2760  
## # … with 39 more rows

# finding the most commonly used words in the ideologies of the parties that were in the top three in at least one election  
top\_three\_ideologies <- top\_three %>%  
 mutate(Ideology = strsplit(as.character(Ideology), ",")) %>%  
 unnest(Ideology) %>%  
 dplyr::select(Ideology)  
  
# creating a wordcloud of the most commonly used words in ideologies  
wordcloud(ideologies$Ideology, min.freq = 2, max.words = 100, random.order = FALSE, colors=brewer.pal(8, "Dark2"))



# creating a wordcloud of the most commonly used words in ideologies of the parties that were in the top three in at least one election  
wordcloud(top\_three\_ideologies$Ideology, min.freq = 2, max.words = 100, random.order = FALSE, colors=brewer.pal(8, "Dark2"))



* Linear Discriminant Analysis:

lda\_data <- log\_data %>%  
 dplyr::select(-c("Timestamp", "gender", "age", "region", "education", "party\_factor")) %>%  
 mutate(party = case\_when(  
 party == "AKP" ~ 1,  
 party == "CHP" ~ 2,  
 party == "MHP" ~ 3,  
 party == "IYI PARTI" ~ 4,  
 party == "HDP" ~ 5,  
 party == "DIĞER" ~ 6))  
  
set.seed(123)  
ind <- sample(2, nrow(lda\_data),  
 replace = TRUE,  
 prob = c(0.6, 0.4))  
  
training <- lda\_data[ind==1,]  
testing <- lda\_data[ind==2,]  
  
linear <- lda(party~., training)  
linear

## Call:  
## lda(party ~ ., data = training)  
##   
## Prior probabilities of groups:  
## 1 2 3 4 5 6   
## 0.12149533 0.26915888 0.03738318 0.33084112 0.01495327 0.22616822   
##   
## Group means:  
## economicalSituation educationalReform againstPrivatization execution  
## 1 0.6000000 0.8461538 0.3076923 0.8923077  
## 2 0.3680556 0.9861111 0.5763889 0.3958333  
## 3 0.3500000 0.9000000 0.3000000 0.8000000  
## 4 0.2768362 0.9548023 0.6553672 0.5649718  
## 5 0.3750000 0.5000000 0.6250000 0.5000000  
## 6 0.3305785 0.9752066 0.4958678 0.5041322  
## objectiveMedia alcoholBan secularState abortionBan stateOfEmergency newParty  
## 1 0.21538462 0.7076923 0.2923077 0.5846154 0.1846154 0.3538462  
## 2 0.02777778 0.1319444 0.9583333 0.1527778 0.8472222 0.7986111  
## 3 0.05000000 0.2000000 0.8500000 0.5000000 0.4000000 0.6500000  
## 4 0.02824859 0.2542373 0.9378531 0.2768362 0.8361582 0.9152542  
## 5 0.12500000 0.2500000 0.7500000 0.2500000 0.6250000 0.7500000  
## 6 0.01652893 0.3223140 0.8760331 0.3388430 0.6694215 0.8512397  
##   
## Coefficients of linear discriminants:  
## LD1 LD2 LD3 LD4 LD5  
## economicalSituation 0.19114629 -0.34243418 0.5897658 0.031682459 0.0260376  
## educationalReform 0.09263693 -4.33985152 -0.4284470 -1.591375278 -0.8858521  
## againstPrivatization -0.25540351 0.69879089 0.4862189 -0.634053548 -0.4850675  
## execution 0.31470612 0.18710403 -0.7646819 -0.009261627 -1.6979623  
## objectiveMedia 1.22015849 0.07287601 0.8484164 -0.482336247 -1.3282186  
## alcoholBan 0.30936196 0.15122825 -0.3787856 -1.475835206 0.9798781  
## secularState -1.86203739 -0.35736582 -1.1802340 1.224122413 -0.2638138  
## abortionBan 0.25083562 0.07689895 -1.1727200 0.270279639 -0.1175508  
## stateOfEmergency -0.86382924 0.38805150 1.0487318 -1.010857302 -0.6617734  
## newParty -0.88778001 0.82970275 -1.2505126 -0.824192270 0.7147293  
##   
## Proportion of trace:  
## LD1 LD2 LD3 LD4 LD5   
## 0.8056 0.0739 0.0632 0.0404 0.0168

Based on the training dataset, 12.1% belongs to group 1 (AKP), 26.9% belongs to group 2 (CHP), 3.7% belongs to group 3 (MHP), 33.1% belongs to group 4 (IYI), 1.5% belongs to group 5 (HDP), 22.6% belongs to group 6 (OTHER).

p1 <- predict(linear, training)$class  
tab <- table(Predicted = p1, Actual = training$party)  
tab

## Actual  
## Predicted 1 2 3 4 5 6  
## 1 46 6 3 4 0 13  
## 2 3 57 6 38 0 34  
## 3 3 2 0 3 0 4  
## 4 7 63 4 108 4 48  
## 5 1 2 2 8 4 3  
## 6 5 14 5 16 0 19

sum(diag(tab))/sum(tab)

## [1] 0.4373832

According to <https://www.statstest.com/linear-discriminant-analysis/>, if your dependent variable is binary, then you should use Simple Logistic Regression, Linear Discriminant Analysis is used when each of the predictor variables is continuous, and if your independent variables are all continuous, then you can use Multinomial Logistic Regression. That clarifies a lot for me, because I kind of experimented with all, but didn’t know which one was proper to use. I think since my predictor variables are binary and my dependent variable is categorical (indeed made binary through voting/not voting approach and having different models for each party), using simple logistic regression looks like a better idea to me. I think the 43.7% accuracy rate of LDA is not good, and it could be because my predictor values are not continuous.