bruhdge

Bryan Mui UID 506021334

2025-12-04

```r
# Import necessary library
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(knitr)
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
#library(kableExtra)

# Read the final dataset
data <- read.csv("df_2024_model.csv")

# Rename the columns to be more readable for the regression analysis
names(data) <- c("State", "Demographics", "Tot_citizen", "Tot_reg", "Prop_reg",
                 "Tot_voted", "Prop_voted", "Voter_part_rate")

# Establish the age levels as detailed by the census
age_patterns <- c(
  "18 to 24 years", "25 to 34 years", "35 to 44 years",
  "45 to 64 years", "65 years and over"
```

```r
)

# Establish the gender levels as detailed by the census
gender_patterns <- c("Male", "Female")

# Establish the region levels as detailed by the census (See Note)
## Note: The region separations are defined here:
### https://www2.census.gov/geo/pdfs/maps-data/maps/reference/us_regdiv.pdf
west <- c("WASHINGTON", "OREGON", "MONTANA", "IDAHO", "WYOMING", "COLORADO", "NEW MEXICO",
          "ARIZONA", "UTAH", "NEVADA", "CALIFORNIA", "HAWAII", "ALASKA")

midwest <- c("NORTH DAKOTA", "SOUTH DAKOTA", "NEBRASKA", "KANSAS", "MINNESOTA", "IOWA", "MISSOURI",
             "WISCONSIN", "ILLINOIS", "MICHIGAN", "INDIANA", "OHIO")

northeast <- c("CONNECTICUT", "MAINE", "MASSACHUSETTS", "NEW HAMPSHIRE", "RHODE ISLAND", "VERMONT", "NEW
               "NEW YORK", "PENNSYLVANIA")

# Separate demographic and region factors of interest into separate columns for easier analysis
data_24 <- data %>% filter(Demographics != "Total", State != "UNITED STATES")
data_24["Age"] <- ifelse(data_24[["Demographics"]] %in% age_patterns, data_24[["Demographics"]], NA)
data_24["Gender"] <- ifelse(data_24[["Demographics"]] %in% gender_patterns, data_24[["Demographics"]], N
data_24["Race"] <- ifelse((!(data_24[["Demographics"]] %in% age_patterns) & !(data_24[["Demographics"]]
                          data_24[["Demographics"]], NA)

data_24["Region"] <- "South"
data_24["Region"] <- ifelse(data_24[["State"]] %in% west, "West", data_24[["Region"]])
data_24["Region"] <- ifelse(data_24[["State"]] %in% midwest, "Midwest", data_24[["Region"]])
data_24["Region"] <- ifelse(data_24[["State"]] %in% northeast, "Northeast", data_24[["Region"]])

head(data_24)
```

```
##     State      Demographics Tot_citizen Tot_reg Prop_reg Tot_voted Prop_voted
## 1 ALABAMA    18 to 24 years         400     212     53.0       167       41.6
## 2 ALABAMA    25 to 34 years         681     409     60.0       318       46.7
## 3 ALABAMA    35 to 44 years         555     370     66.7       304       54.8
## 4 ALABAMA    45 to 64 years        1228     904     73.6       793       64.6
## 5 ALABAMA 65 years and over         915     709     77.5       637       69.7
## 6 ALABAMA       Asian alone          45      30     65.9        30       65.9
##   Voter_part_rate               Age Gender        Race Region
## 1       0.7877358    18 to 24 years   <NA>        <NA>  South
## 2       0.7775061    25 to 34 years   <NA>        <NA>  South
## 3       0.8216216    35 to 44 years   <NA>        <NA>  South
## 4       0.8772124    45 to 64 years   <NA>        <NA>  South
## 5       0.8984485 65 years and over   <NA>        <NA>  South
## 6       1.0000000              <NA>   <NA> Asian alone  South
```

Contingency Tables

Region

Table 1: Contingengy Table: Count by Region

Region	Count
Midwest	180
Northeast	135
South	255
West	195
Total	765

```
contingency_region <- data_24 %>%
  group_by(Region) %>%
  summarise(Count=n())

grand_total <- data_24 %>%
  summarise(
    Region = "Total", # Set a label for the total row
    Count = n()    # Sum the 'count' column from the group summary
  )

contingency_region <- bind_rows(contingency_region, grand_total)
contingency_region
```

```
## # A tibble: 5 x 2
##   Region    Count
##   <chr>     <int>
## 1 Midwest     180
## 2 Northeast   135
## 3 South       255
## 4 West        195
## 5 Total       765
```

```
(kable(contingency_region, format = "latex", caption = "Contingengy Table: Count by Region"))
```

Race

```
contingency_race <- data_24 %>%
  filter(!is.na(Race)) %>%
  group_by(Race) %>%
  summarise(Count=n())

grand_total <- data_24 %>%
  filter(!is.na(Race)) %>%
  summarise(
    Race = "Total", # Set a label for the total row
    Count = n()    # Sum the 'count' column from the group summary
  )

contingency_race <- bind_rows(contingency_race, grand_total)
contingency_race
```

```
## # A tibble: 9 x 2
```

Table 2: Contingengy Table: Count by Race

Region	Count
Midwest	180
Northeast	135
South	255
West	195
Total	765

```
##   Race                           Count
##   <chr>                          <int>
## 1 Asian alone                       51
## 2 Asian alone or in combination     51
## 3 Black alone                       51
## 4 Black alone or in combination     51
## 5 Hispanic (any race)               51
## 6 White alone                       51
## 7 White alone or in combination     51
## 8 White non-Hispanic alone          51
## 9 Total                            408
```

```
(kable(contingency_region, format = "latex", caption = "Contingengy Table: Count by Race"))
```

Gender

```
contingency_gender <- data_24 %>%
  filter(!is.na(Gender)) %>%
  group_by(Gender) %>%
  summarise(Count=n())

grand_total_gender <- data_24 %>%
  filter(!is.na(Gender)) %>%
  summarise(
    Gender = "Total",
    Count = n()
  )

contingency_gender <- bind_rows(contingency_gender, grand_total_gender)
contingency_gender
```

```
## # A tibble: 3 x 2
##   Gender Count
##   <chr>  <int>
## 1 Female    51
## 2 Male      51
## 3 Total    102
```

```
(kable(contingency_gender, format = "latex", caption = "Contingengy Table: Count by Gender"))
```

Age

Table 3: Contingengy Table: Count by Gender

Gender	Count
Female	51
Male	51
Total	102

Table 4: Contingengy Table: Count by Age

Gender	Count
Female	51
Male	51
Total	102

```r
contingency_age <- data_24 %>%
  filter(!is.na(Age)) %>%
  group_by(Age) %>%
  summarise(Count=n())

grand_total_age <- data_24 %>%
  filter(!is.na(Age)) %>%
  summarise(
    Age = "Total",
    Count = n()
  )

contingency_age <- bind_rows(contingency_age, grand_total_age)
contingency_age
```

```
## # A tibble: 6 x 2
##   Age                Count
##   <chr>              <int>
## 1 18 to 24 years        51
## 2 25 to 34 years        51
## 3 35 to 44 years        51
## 4 45 to 64 years        51
## 5 65 years and over     51
## 6 Total                255
```

```r
(kable(contingency_gender, format = "latex", caption = "Contingengy Table: Count by Age"))
```

VIF

Race + Region

```r
# --- Model A: Race + Region ---
# Filter for non-NA values in Region and Race
data_region_race <- data_24 %>%
  filter(!is.na(Region), !is.na(Race))

# Linear Model
```

Table 5: VIF Results: Region and Race

	GVIF	Df	GVIF^(1/(2*Df))
Region	1	3	1
Race	1	7	1

```r
model_region_race <- lm(Voter_part_rate ~ Region + Race, data = data_region_race)

# Calculate VIF
vif_region_race <- vif(model_region_race)

print("--- VIF: Region + Race (LaTeX) ---")
```

```
## [1] "--- VIF: Region + Race (LaTeX) ---"
```

```r
print(vif_region_race)
```

```
##        GVIF Df GVIF^(1/(2*Df))
## Region    1  3               1
## Race      1  7               1
```

```r
kable(as.data.frame(vif_region_race), format = "latex", caption = "VIF Results: Region and Race")
```

Region + Age

```r
# Filter for non-NA values in Region and Age
data_region_age <- data_24 %>%
  filter(!is.na(Region), !is.na(Age))

# Linear Model
model_region_age <- lm(Voter_part_rate ~ Region + Age, data = data_region_age)

# Calculate VIF
vif_region_age <- vif(model_region_age)

print("--- VIF: Region + Age (LaTeX) ---")
```

```
## [1] "--- VIF: Region + Age (LaTeX) ---"
```

```r
print(vif_region_age)
```

```
##        GVIF Df GVIF^(1/(2*Df))
## Region    1  3               1
## Age       1  4               1
```

```r
kable(as.data.frame(vif_region_age), format = "latex", caption = "VIF Results: Region and Age")
```

Region + Gender

Table 6: VIF Results: Region and Age

	GVIF	Df	GVIF^(1/(2*Df))
Region	1	3	1
Age	1	4	1

Table 7: VIF Results: Region and Gender

	GVIF	Df	GVIF^(1/(2*Df))
Region	1	3	1
Gender	1	1	1

```
# Filter for non-NA values in Region and Gender
data_region_gender <- data_24 %>%
  filter(!is.na(Region), !is.na(Gender))

# Linear Model
model_region_gender <- lm(Voter_part_rate ~ Region + Gender, data = data_region_gender)

# Calculate VIF
vif_region_gender <- vif(model_region_gender)

print("--- VIF: Region + Gender (LaTeX) ---")
```

```
## [1] "--- VIF: Region + Gender (LaTeX) ---"
```

```
print(vif_region_gender)
```

```
##        GVIF Df GVIF^(1/(2*Df))
## Region    1  3               1
## Gender    1  1               1
```

```
kable(as.data.frame(vif_region_gender), format = "latex", caption = "VIF Results: Region and Gender")
```

VIF Debugging

```
# # 1. Create a dataset where one variable is a perfect copy of another
# data_vif_debug <- data_24 %>%
#   filter(!is.na(Region), !is.na(Gender)) %>%
#   mutate(
#     Region = Region,
#     Region_Copy = Region # Perfect correlation
#   )
#
# # 2. Linear Model with two identical predictors
# # Note: lm() will often drop one of the perfectly correlated predictors automatically,
# # but VIF analysis should still flag the issue or throw an error/warning.
# model_vif_debug <- lm(Voter_part_rate ~ Region + Region_Copy, data = data_vif_debug)
#
# # 3. Calculate VIF
```

```
# # This is expected to produce a high number (or an error/warning)
# vif_vif_debug <- vif(model_vif_debug)
#
# print("--- VIF: Debug Test (Region + Region_Copy) (LaTeX) ---")
# # Only kable the result if the VIF calculation was successful
# if (is.numeric(vif_vif_debug)) {
#   kable(as.data.frame(vif_vif_debug), format = "latex", caption = "VIF Debug Results: Region and Regi
# } else {
#   print("VIF calculation failed or returned non-numeric result due to perfect collinearity. Check mod
# }
#
# summary(model_vif_debug)
```