The Incumbent's Shadow

Emmanuel Kasigazi

2025-04-05

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(openintro)
```

```
## Loading required package: airports
## Loading required package: cherryblossom
## Loading required package: usdata
```

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(dplyr)
library(tinytex)
```

R Markdown

```
library(tidyverse)
campaign_events <- read.csv("candidate_visits.csv")
```

```
# Load required libraries
library(tidyverse)

# Total Trump events
```

```r
trump_events_count <- sum(campaign_events$Candidate.name == "Donald Trump")

# Total non-Trump events
non_trump_events_count <- nrow(campaign_events) - trump_events_count

# Events by candidate
candidate_summary <- campaign_events %>%
  group_by(Candidate.name) %>%
  summarize(
    event_count = n(),
    percentage = round(n() / nrow(campaign_events) * 100, 1)
  ) %>%
  arrange(desc(event_count))

# Distribution across states
state_summary <- campaign_events %>%
  group_by(State) %>%
  summarize(event_count = n()) %>%
  arrange(desc(event_count))

# Time distribution by month
campaign_events$Date <- as.Date(campaign_events$Date, format="%m/%d/%Y")
campaign_events$month <- format(campaign_events$Date, "%Y-%m")
time_summary <- campaign_events %>%
  group_by(month) %>%
  summarize(event_count = n()) %>%
  arrange(month)

# Print summaries
print(paste("Total Trump events:", trump_events_count))
```

```
## [1] "Total Trump events: 116"
```

```r
print(paste("Total non-Trump events:", non_trump_events_count))
```

```
## [1] "Total non-Trump events: 1625"
```

```r
print(candidate_summary)
```

```
## # A tibble: 11 x 3
##    Candidate.name  event_count percentage
##    <chr>                 <int>      <dbl>
##  1 Vivek Ramaswamy         490       28.1
##  2 Ron DeSantis            285       16.4
##  3 Nikki Haley             272       15.6
##  4 Asa Hutchinson          157        9
##  5 Donald Trump            116        6.7
##  6 Doug Burgum             101        5.8
##  7 Mike Pence               99        5.7
##  8 Tim Scott                94        5.4
##  9 Chris Christie           75        4.3
## 10 Will Hurd                30        1.7
## 11 Francis Suarez           22        1.3
```

```r
print(head(state_summary, 15))
```

```
## # A tibble: 15 x 2
##    State                event_count
##    <chr>                      <int>
##  1 Iowa                         900
##  2 New Hampshire                395
##  3 South Carolina               116
##  4 Florida                       44
##  5 Texas                         40
##  6 District of Columbia          38
##  7 California                    33
##  8 Nevada                        17
##  9 New York                      17
## 10 Michigan                      16
## 11 Georgia                       13
## 12 Pennsylvania                  12
## 13 Wisconsin                     11
## 14 Alabama                        8
## 15 Indiana                        8
```

```r
print(time_summary)
```

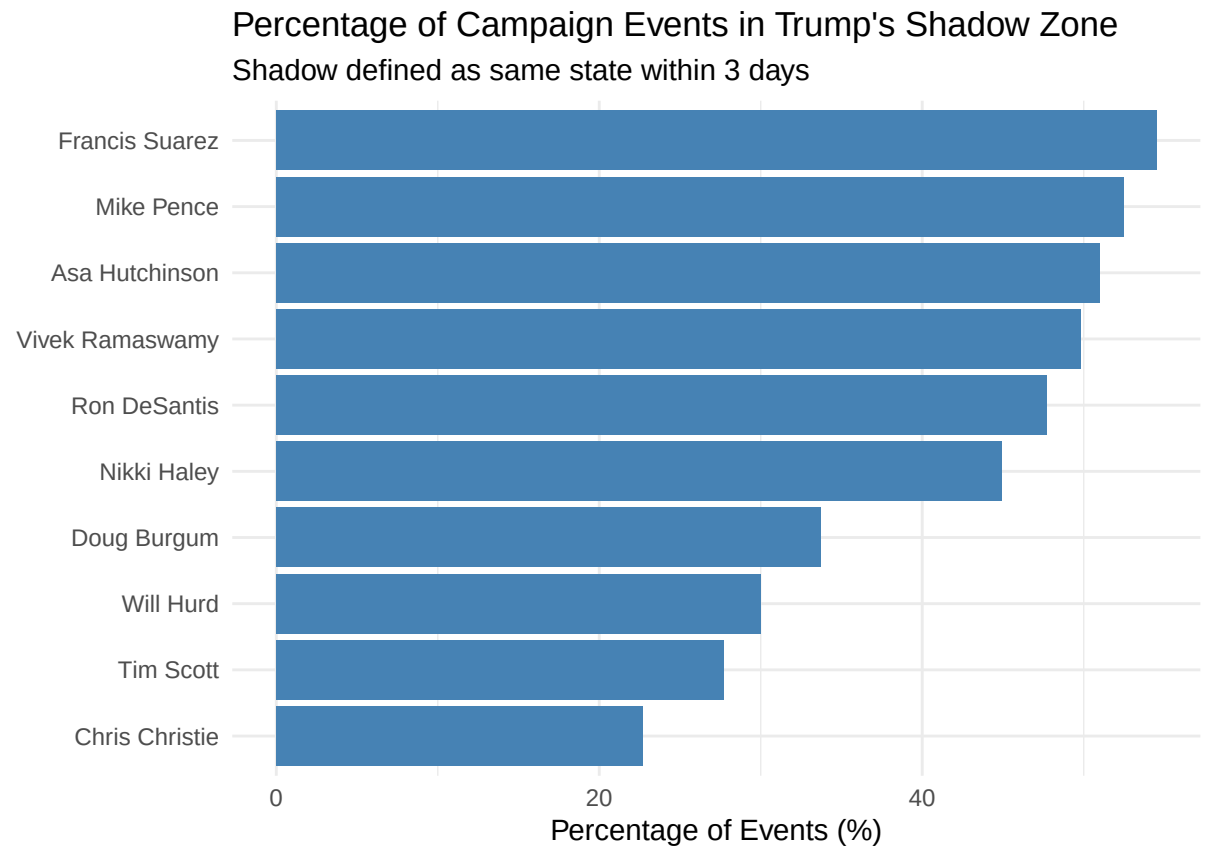
```
## # A tibble: 15 x 2
##    month   event_count
##    <chr>         <int>
##  1 2023-01           3
##  2 2023-02          15
##  3 2023-03          19
##  4 2023-04          52
##  5 2023-05          64
##  6 2023-06         125
##  7 2023-07         155
##  8 2023-08         207
##  9 2023-09         187
## 10 2023-10         194
## 11 2023-11         185
## 12 2023-12         245
## 13 2024-01         214
## 14 2024-02          67
## 15 2024-03           9
```

```r
# Load required packages
library(tidyverse)
library(lubridate)

# Read the dataset
campaign_events <- read.csv("candidate_visits.csv")

# Convert date column to proper Date format
campaign_events$Date <- as.Date(campaign_events$Date, format = "%m/%d/%Y")
```

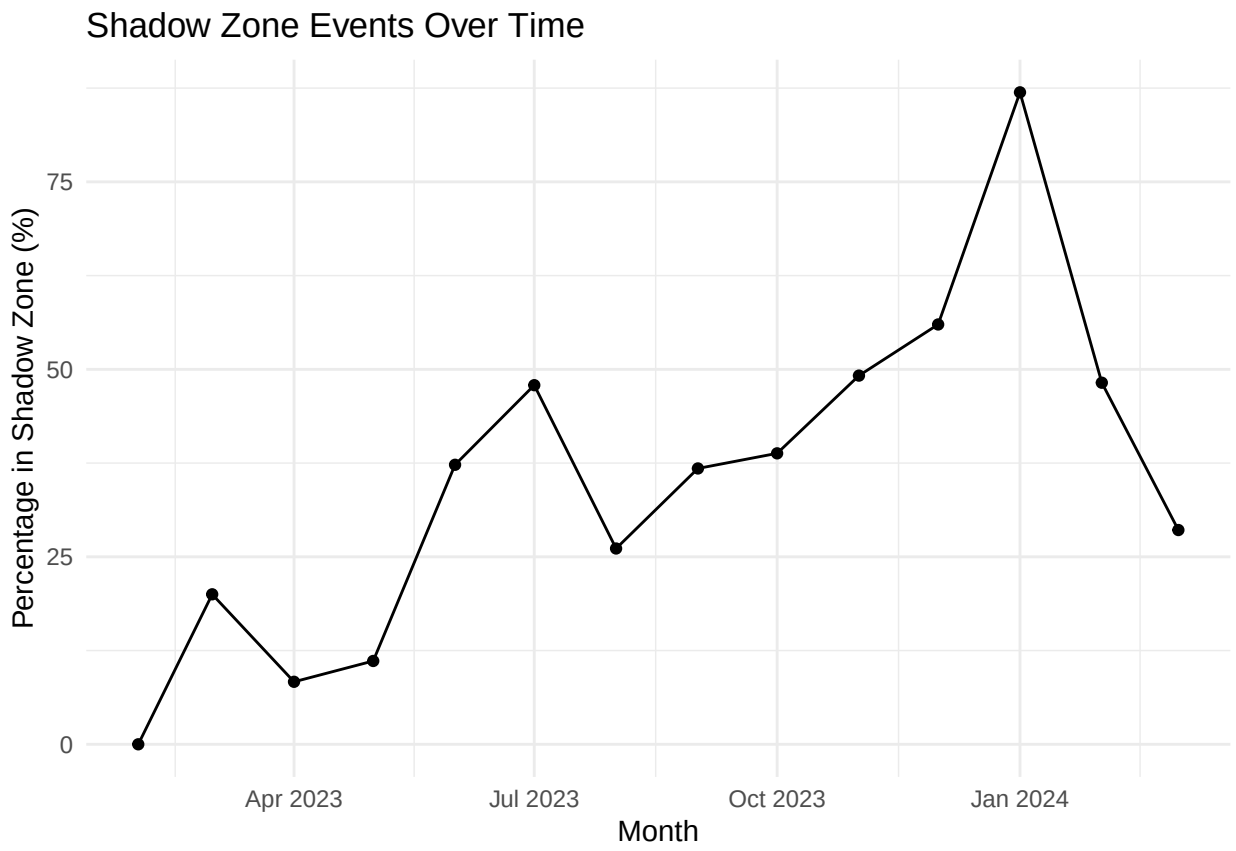
```r
# Extract Trump's events
trump_events <- campaign_events %>%
  filter(Candidate.name == "Donald Trump") %>%
  select(Date, State, City)

# Define shadow zone parameters
days_window <- 3  # Events within 3 days before/after a Trump event
location_level <- "State"  # Define shadow at state level

# Create shadow zone indicator
campaign_events <- campaign_events %>%
  mutate(in_shadow_zone = FALSE)  # Initialize

# For each non-Trump event, check if it falls within a shadow zone
for (i in 1:nrow(campaign_events)) {
  if (campaign_events$Candidate.name[i] != "Donald Trump") {
    # Check if state matches any Trump event
    state_match <- campaign_events$State[i] %in% trump_events$State

    # Check time proximity to Trump events
    event_date <- campaign_events$Date[i]
    time_matches <- trump_events %>%
      filter(abs(as.numeric(difftime(Date, event_date, units = "days"))) <= days_window)

    # Check if any of the time-proximate events are in the same state
    shadow_match <- time_matches %>%
      filter(State == campaign_events$State[i]) %>%
      nrow() > 0

    # If in shadow zone, update indicator
    campaign_events$in_shadow_zone[i] <- shadow_match
  }
}
```

```r
# Summary of events in shadow zones by candidate
shadow_summary <- campaign_events %>%
  filter(Candidate.name != "Donald Trump") %>%
  group_by(Candidate.name) %>%
  summarize(
    total_events = n(),
    in_shadow = sum(in_shadow_zone),
    pct_in_shadow = round(in_shadow / total_events * 100, 1)
  ) %>%
  arrange(desc(pct_in_shadow))

# Visualization of shadow zone percentages by candidate
ggplot(shadow_summary, aes(x = reorder(Candidate.name, pct_in_shadow), y = pct_in_shadow)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  coord_flip() +
  labs(
    title = "Percentage of Campaign Events in Trump's Shadow Zone",
    subtitle = paste0("Shadow defined as same state within ", days_window, " days"),
    x = "",
```

```
    y = "Percentage of Events (%)"
  ) +
  theme_minimal()
```




```
# Time series visualization of shadow zone events
monthly_shadow <- campaign_events %>%
  filter(Candidate.name != "Donald Trump") %>%
  mutate(month = floor_date(Date, "month")) %>%
  group_by(month, in_shadow_zone) %>%
  summarise(count = n(), .groups = "drop") %>%  # Add .groups="drop" to prevent the warning
  spread(in_shadow_zone, count, fill = 0) %>%
  rename(outside_shadow = `FALSE`, inside_shadow = `TRUE`) %>%
  mutate(pct_in_shadow = inside_shadow / (inside_shadow + outside_shadow) * 100)

ggplot(monthly_shadow, aes(x = month, y = pct_in_shadow)) +
  geom_line() +
  geom_point() +
  labs(
    title = "Shadow Zone Events Over Time",
    x = "Month",
    y = "Percentage in Shadow Zone (%)"
  ) +
  theme_minimal()
```




```r
# Create contingency table: Candidate vs. Shadow Zone
shadow_table <- table(
  campaign_events$Candidate.name[campaign_events$Candidate.name != "Donald Trump"],
  campaign_events$in_shadow_zone[campaign_events$Candidate.name != "Donald Trump"]
)

# Run chi-square test
chi_test <- chisq.test(shadow_table)

# Print results
print(chi_test)
```

```
##
##  Pearson's Chi-squared test
##
## data:  shadow_table
## X-squared = 45.229, df = 9, p-value = 8.368e-07
```

```r
# Examine standardized residuals to understand which candidates
# significantly avoided or sought Trump's shadow
chi_residuals <- data.frame(
  Candidate = rownames(chi_test$residuals),
  Avoid_Shadow = chi_test$residuals[,1],
  Seek_Shadow = chi_test$residuals[,2]
)
```

```r
# Sort by most extreme residuals
chi_residuals <- chi_residuals %>%
  arrange(desc(abs(Seek_Shadow)))
```

```r
# Create contingency table: Candidate vs. Shadow Zone
shadow_table <- table(
  campaign_events$Candidate.name[campaign_events$Candidate.name != "Donald Trump"],
  campaign_events$in_shadow_zone[campaign_events$Candidate.name != "Donald Trump"]
)

# Run chi-square test
chi_test <- chisq.test(shadow_table)

# Print results
print(chi_test)
```

```
##
##  Pearson's Chi-squared test
##
## data:  shadow_table
## X-squared = 45.229, df = 9, p-value = 8.368e-07
```

```r
# Examine standardized residuals to understand which candidates
# significantly avoided or sought Trump's shadow
chi_residuals <- data.frame(
  Candidate = rownames(chi_test$residuals),
  Avoid_Shadow = chi_test$residuals[,1],
  Seek_Shadow = chi_test$residuals[,2]
)

# Sort by most extreme residuals
chi_residuals <- chi_residuals %>%
  arrange(desc(abs(Seek_Shadow)))
```

```r
# Create contingency table: Candidate vs. Shadow Zone
shadow_table <- table(
  campaign_events$Candidate.name[campaign_events$Candidate.name != "Donald Trump"],
  campaign_events$in_shadow_zone[campaign_events$Candidate.name != "Donald Trump"]
)

# Run chi-square test
chi_test <- chisq.test(shadow_table)

# Print results and examine residuals
print(chi_test)
```

```
##
##  Pearson's Chi-squared test
##
## data:  shadow_table
## X-squared = 45.229, df = 9, p-value = 8.368e-07
```

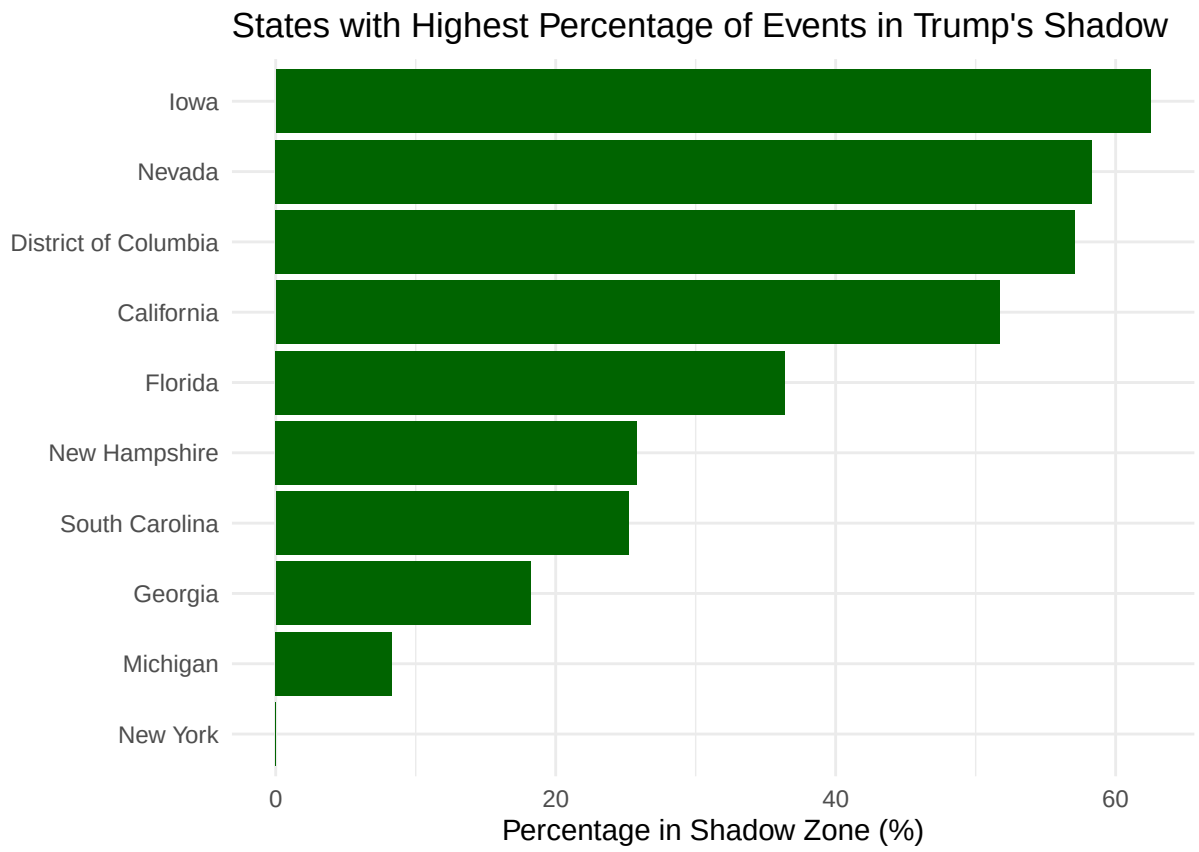
```r
print(chi_test$residuals)
```

```
##
##                          FALSE        TRUE
##   Asa Hutchinson  -0.99881232  1.10319956
##   Chris Christie   2.61445828 -2.88769888
##   Doug Burgum      1.54315755 -1.70443505
##   Francis Suarez  -0.60104077  0.66385636
##   Mike Pence      -1.00384767  1.10876116
##   Nikki Haley      0.04298540 -0.04747787
##   Ron DeSantis    -0.60875957  0.67238187
##   Tim Scott        2.27394014 -2.51159273
##   Vivek Ramaswamy -1.41830970  1.56653918
##   Will Hurd        1.11169865 -1.22788379
```

```r
# 1. Analyze by state
state_shadow <- campaign_events %>%
  filter(Candidate.name != "Donald Trump") %>%
  group_by(State) %>%
  summarize(
    total_events = n(),
    in_shadow = sum(in_shadow_zone),
    pct_in_shadow = round(in_shadow / total_events * 100, 1)
  ) %>%
  filter(total_events >= 10) %>%  # Only include states with sufficient data
  arrange(desc(pct_in_shadow))

# 2. Analyze by event type
event_shadow <- campaign_events %>%
  filter(Candidate.name != "Donald Trump") %>%
  group_by(Primary.Purpose) %>%
  summarize(
    total_events = n(),
    in_shadow = sum(in_shadow_zone),
    pct_in_shadow = round(in_shadow / total_events * 100, 1)
  ) %>%
  filter(total_events >= 10) %>%  # Only include event types with sufficient data
  arrange(desc(pct_in_shadow))
```

```r
# State-level shadow zone analysis
state_shadow <- campaign_events %>%
  filter(Candidate.name != "Donald Trump") %>%
  group_by(State) %>%
  summarise(
    total_events = n(),
    in_shadow = sum(in_shadow_zone),
    pct_in_shadow = round(in_shadow / total_events * 100, 1),
    .groups = "drop"
  ) %>%
  filter(total_events >= 10) %>%  # Only include states with sufficient data
  arrange(desc(pct_in_shadow))

# Visualize top states by shadow percentage
```

```r
ggplot(head(state_shadow, 10), aes(x = reorder(State, pct_in_shadow), y = pct_in_shadow)) +
  geom_bar(stat = "identity", fill = "darkgreen") +
  coord_flip() +
  labs(
    title = "States with Highest Percentage of Events in Trump's Shadow",
    x = "",
    y = "Percentage in Shadow Zone (%)"
  ) +
  theme_minimal()
```

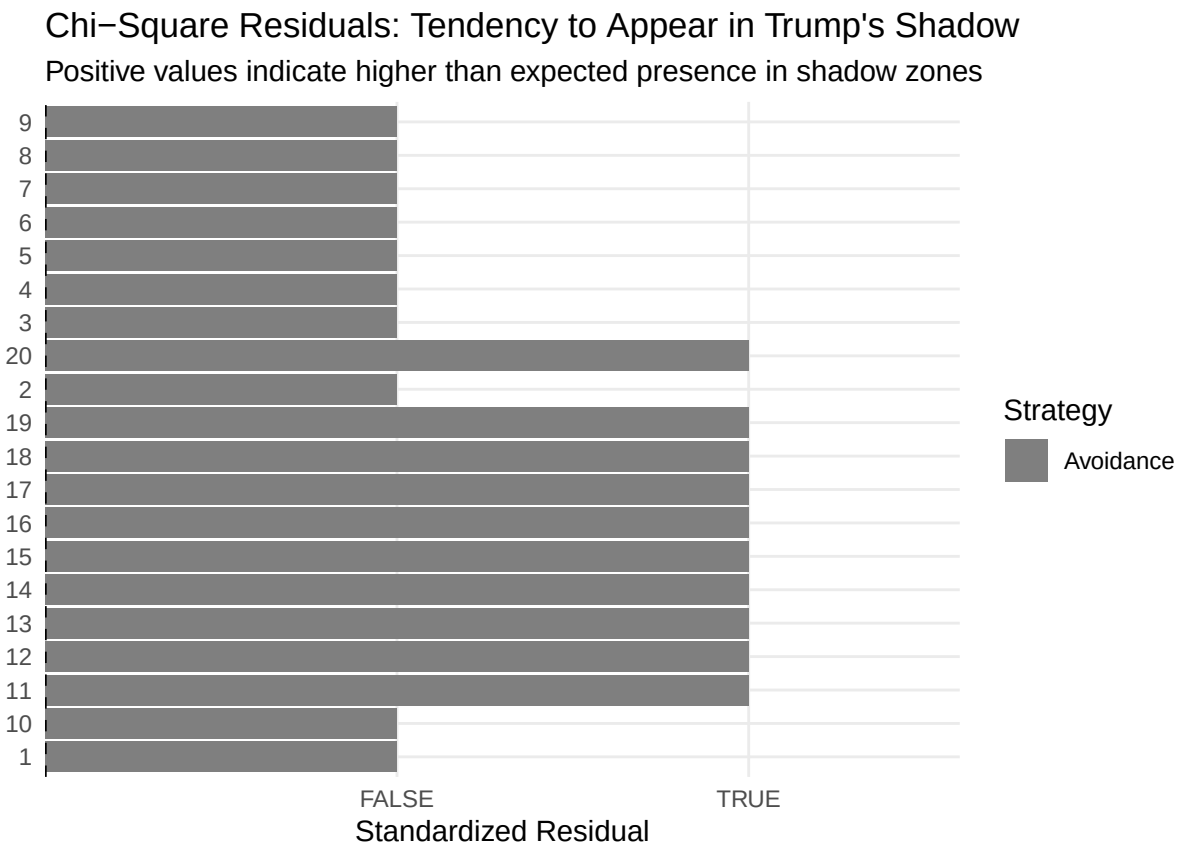



```r
# Calculate and visualize standardized residuals
residuals_df <- as.data.frame(chi_test$residuals)
names(residuals_df) <- c("Outside_Shadow", "In_Shadow")
residuals_df$Candidate <- rownames(residuals_df)

# Plot residuals for "In Shadow" values
ggplot(residuals_df, aes(x = reorder(Candidate, In_Shadow), y = In_Shadow)) +
  geom_bar(stat = "identity", aes(fill = In_Shadow > 0)) +
  coord_flip() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(
    title = "Chi-Square Residuals: Tendency to Appear in Trump's Shadow",
    subtitle = "Positive values indicate higher than expected presence in shadow zones",
    x = "",
    y = "Standardized Residual"
```

```r
  ) +
  scale_fill_manual(values = c("red", "blue"),
                    name = "Strategy",
                    labels = c("Avoidance", "Following")) +
  theme_minimal()
```

```
## Warning in Ops.factor(In_Shadow, 0): '>' not meaningful for factors
```

```
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
```




```r
#visualize the standardized residuals from my chi-square test:
# Create a data frame from the residuals for the "TRUE" column only
# (representing "in shadow zone")
residuals_df <- data.frame(
  Candidate = rownames(chi_test$residuals),
  Shadow_Residual = chi_test$residuals[, "TRUE"]
)

# Sort by residual value
residuals_df <- residuals_df %>%
  arrange(Shadow_Residual)

# Add a column indicating if the candidate follows or avoids
residuals_df$Strategy <- ifelse(residuals_df$Shadow_Residual > 0,
                                "Shadow Seeker", "Shadow Avoider")

# Create a cleaner visualization
ggplot(residuals_df, aes(x = reorder(Candidate, Shadow_Residual),
                         y = Shadow_Residual,
                         fill = Strategy)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(
    title = "Chi-Square Residuals: Tendency to Appear in Trump's Shadow",
    subtitle = "Positive values indicate higher than expected presence in shadow zones",
```

```
  x = "",
  y = "Standardized Residual"
) +
scale_fill_manual(values = c("Shadow Avoider" = "#D55E00",
                             "Shadow Seeker" = "#0072B2")) +
theme_minimal()
```