5. Visualisations for Thesis Document

Mafalda González González

# libraries

library(flextable)  
library(dplyr)  
library(purrr)  
library(ggplot2)  
library(tidyr)  
library(tibble)  
library(lubridate)  
library(stringr)  
library(readr)  
  
library(DataExplorer) # EDA   
library(skimr)  
library(ggalluvial)  
library(data.table)   
library(lubridate)  
library(zoo)  
library(forcats)  
  
  
library(showtext)  
systemfonts::font\_info("Times New Roman")

# A tibble: 1 × 24  
 path index family style italic bold monospace weight width kerning color  
 <chr> <int> <chr> <chr> <lgl> <lgl> <lgl> <ord> <ord> <lgl> <lgl>  
1 "C:\\WIN… 0 Times… Regu… FALSE FALSE FALSE normal norm… TRUE FALSE  
# ℹ 13 more variables: scalable <lgl>, vertical <lgl>, n\_glyphs <int>,  
# n\_sizes <int>, n\_charmaps <int>, bbox <list>, max\_ascend <dbl>,  
# max\_descend <dbl>, max\_advance\_width <dbl>, max\_advance\_height <dbl>,  
# lineheight <dbl>, underline\_pos <dbl>, underline\_size <dbl>

font\_add("Times New Roman", regular = "C:\\WINDOWS\\Fonts\\times.ttf")  
showtext\_auto()   
showtext\_opts(dpi = 320)  
  
  
  
# party palette   
party\_colors <- c("PP" = "#1db4e8",  
 "PSOE" = "#c30505",  
 "SUMAR" = "#e71853",  
 "PODEMOS" = "#a444b4",  
 "VOX" = "#83b431",  
 "ERC" = "#ffbf41",  
 "ERC-CATSI" = "#ffbf41",  
 "CIU" = "#1b348a",  
 "CDC" = "#1b348a",  
 "DIL" = "#1b348a",  
 "MP" = "#004938",  
 "CS" = "#eb6109",  
 "PNV" = "darkgreen",  
 "BNG" = "lightblue",  
 "EH-BILDU" = "lightgreen",  
 "JXCAT-JUNTS" = "#03cfb4",  
 "CC" = "#2f6da6",  
 "UPN" = "#e72c2e",  
 "NC-BC" = "#81c03b",  
 "UPL" = "#b71966",  
 "EXISTE" = "#227e57",  
 "CUP" = "#fff201",  
 "ECP" = "#a444b4",   
 "ENMAREA" = "#a444b4",  
 "COMPROMIS" = "#d95827",  
 "IU" = "#a9272f",   
 "UPYD" = "#e5007d",  
 "AMAIUR" = "#0198b3",  
 "ERPV" = "#ffbf41",  
 "PSA-PA" = "#19a24a",  
 "CDS" = "#b2c544",  
 "AP-PDP-PL" = "#ffa518",  
 "UCD" = "#1a7e36",  
 "PCE" = "#961425",  
 "HB" = "#613000"  
 )

# Data

survey\_elections <- readRDS("./data/survey\_elections.rds")  
  
vis\_2023 <- survey\_elections %>%   
 filter(id\_elec == "02-2023-07-24")  
vis\_hist <- survey\_elections %>%   
 filter(id\_elec != "02-2023-07-24")  
  
  
data\_2023\_EDA <- read\_rds("./data/data\_2023\_EDA.rds")  
data\_2023\_eval\_full <- read\_rds("./data/data\_2023\_eval\_full.rds")  
  
election\_day <- as.Date(sub("^02-", "", unique(vis\_2023$id\_elec)))  
  
data\_2023\_consensus <- read\_rds("./data/data\_2023\_consensus.rds")  
  
pred\_store\_2\_pct <- read\_rds("./data/results\_models\_2\_pct.rds")  
pred\_store\_1\_pct <- readRDS("./data/results\_models\_1\_pct.rds")  
  
complete\_2023\_2\_pct <- read\_rds("./data/complete\_2023\_2\_pct.rds")  
complete\_2023\_1\_pct <- read\_rds("./data/complete\_2023\_1\_pct.rds")

Saving figures:

dir.create("visual exports", showWarnings = FALSE)

# Table 3.1

Manual:

table\_3\_1 <- tribble(  
 ~Dataset, ~`Original obs`, ~`NA election results`, ~`after drop - NA election results`,  
 ~`obs below 2% vote share cutoff`, ~`after drop - 2% vote share cutoff`, ~`Final parties`,  
 "2023", 11210, 3516, 7694, 4712, 2982, "PP, PSOE, Vox, SUMAR",  
 "historical", 18077, 1777, 16300, 5955, 10345, NA\_character\_  
)  
  
flextable(table\_3\_1) %>%   
 autofit()

| Dataset | Original obs | NA election results | after drop - NA election results | obs below 2% vote share cutoff | after drop - 2% vote share cutoff | Final parties |
| --- | --- | --- | --- | --- | --- | --- |
| 2023 | 11,210 | 3,516 | 7,694 | 4,712 | 2,982 | PP, PSOE, Vox, SUMAR |
| historical | 18,077 | 1,777 | 16,300 | 5,955 | 10,345 |  |

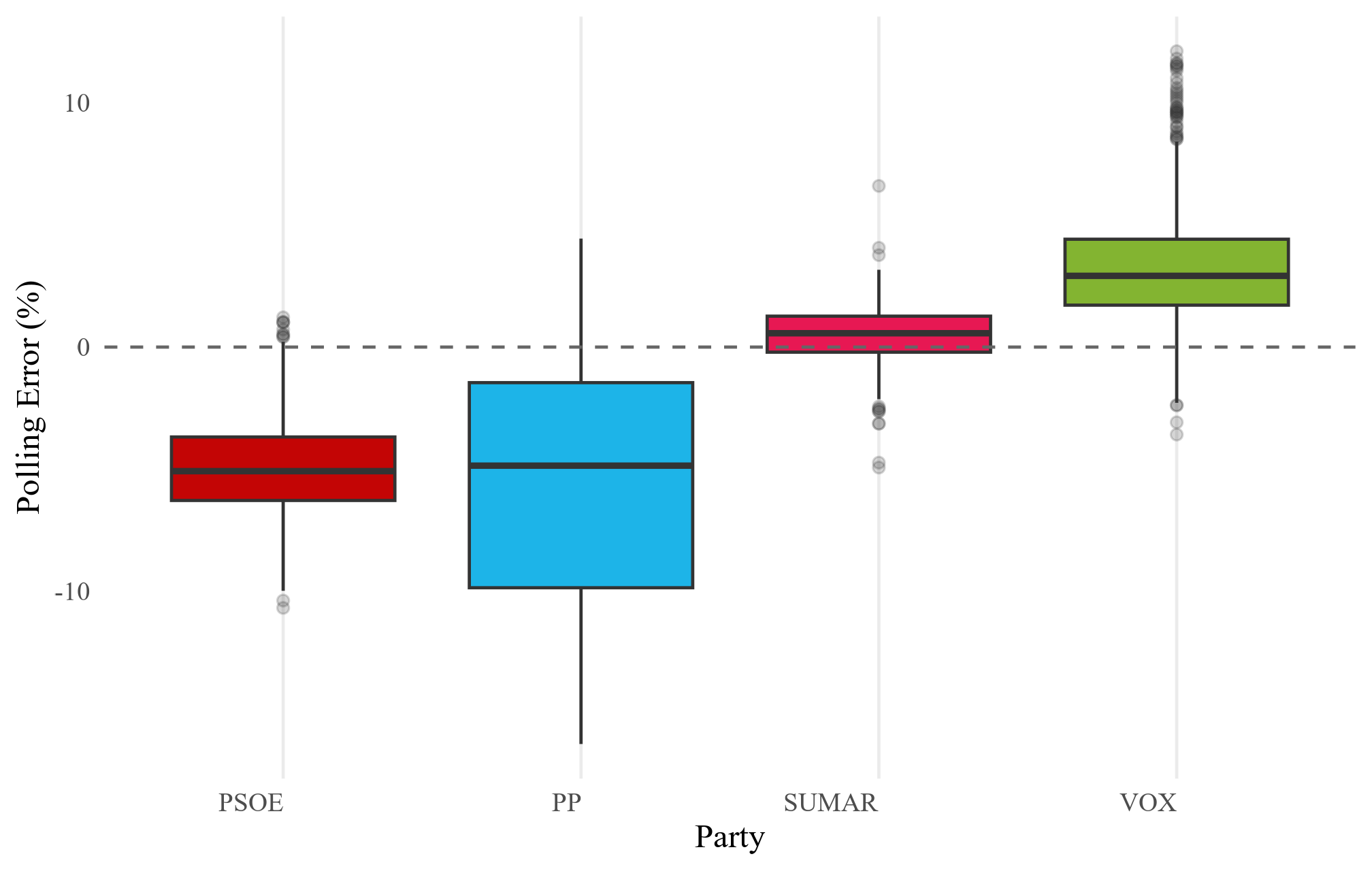
# Table 4.1

data\_2023\_EDA <- data\_2023\_EDA %>%   
 mutate(error = estimated\_voting - voting\_results\_pct)  
  
table\_4\_1 <- data\_2023\_EDA %>%   
 group\_by(abbrev\_candidacies) %>%   
 summarise(  
 "Vote share" = round(first(voting\_results\_pct), 2),   
 "Avg pred" = round(mean(estimated\_voting), 2),   
 "Avg error" = round(mean(error), 2),   
 "Polls" = n(),   
 "First year" = first(first\_year),   
 "Party age" = first(party\_age),   
 "Elections" = first(party\_elec\_count),   
 "Fieldwork start" = min(fieldwork\_start),   
 "Fieldwork end" = max(fieldwork\_end)  
 ) %>%   
 arrange(desc("Vote share"))  
  
flextable(table\_4\_1) %>%   
 autofit()

| abbrev\_candidacies | Vote share | Avg pred | Avg error | Polls | First year | Party age | Elections | Fieldwork start | Fieldwork end |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PP | 33.06 | 27.60 | -5.46 | 927 | 1,989 | 34 | 10 | 2019-11-12 | 2023-07-22 |
| PSOE | 31.68 | 26.70 | -4.98 | 927 | 1,982 | 41 | 12 | 2019-11-12 | 2023-07-22 |
| SUMAR | 12.33 | 12.84 | 0.50 | 202 | 2,023 | 0 | 0 | 2023-03-31 | 2023-07-22 |
| VOX | 12.38 | 15.65 | 3.27 | 926 | 2,015 | 8 | 3 | 2019-11-12 | 2023-07-22 |

# Figure 4.1

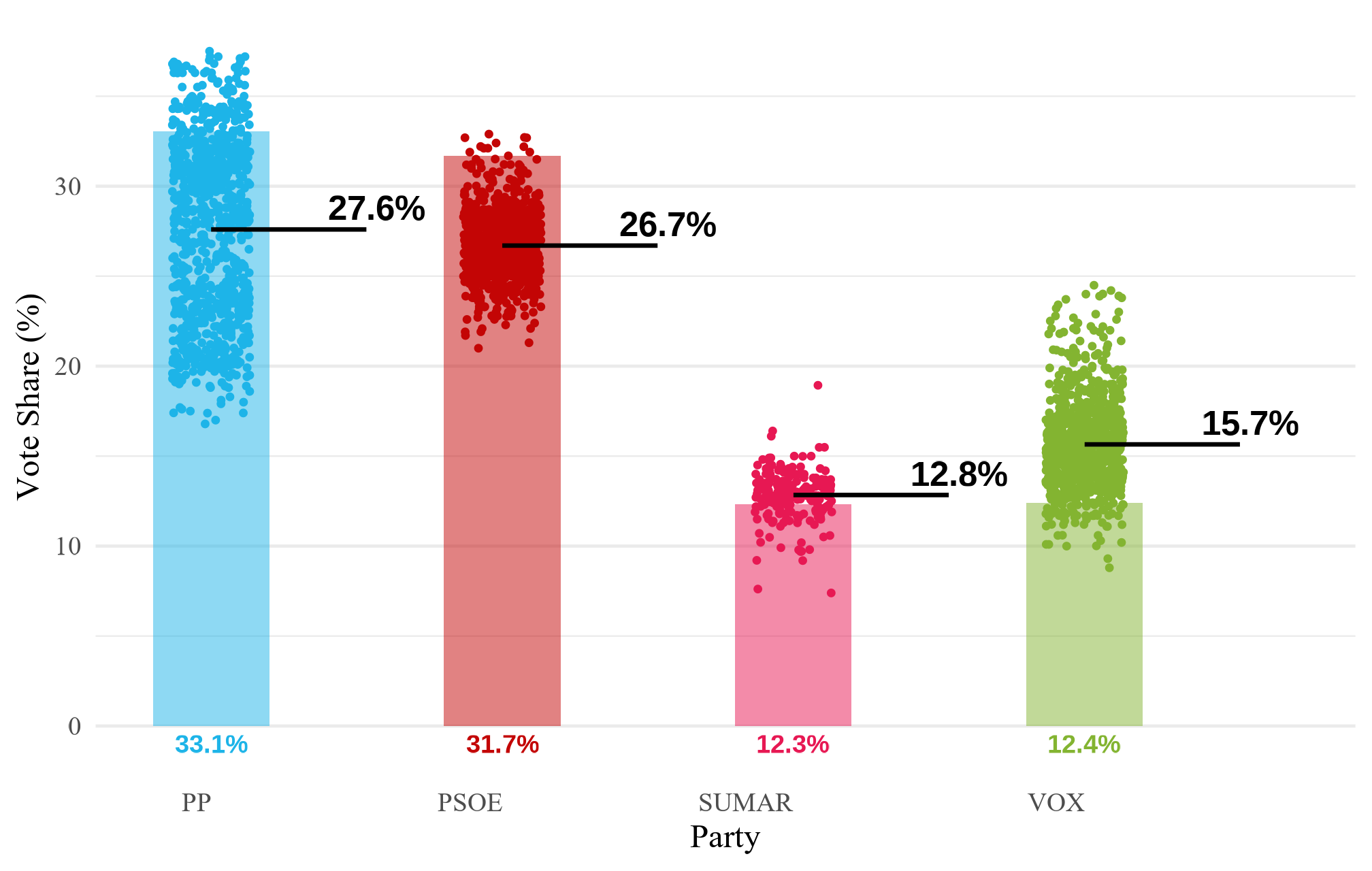
year\_lab <- first(data\_2023\_EDA$year)  
  
figure\_4\_1 <- data\_2023\_EDA %>%  
 ggplot(aes(x = fct\_reorder(abbrev\_candidacies, error, .fun = median),  
 y = error, fill = abbrev\_candidacies)) +  
 geom\_boxplot(outlier.alpha = 0.2) +  
 geom\_hline(yintercept = 0, linetype = "dashed", colour = "grey40") +  
 labs(  
 x = "Party",   
 y = "Polling Error (%)"  
 ) +  
 theme\_minimal(base\_family = "Times New Roman") +  
 theme(  
 legend.position = "none",  
 axis.text.x = element\_text(angle = 0, hjust = 1),  
 panel.grid.major.y = element\_blank(),  
 panel.grid.minor.y = element\_blank()  
 ) +  
 scale\_fill\_manual(values = party\_colors)  
  
figure\_4\_1



ggsave("visual exports/figure\_4\_1.png", figure\_4\_1,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Figure 4.2

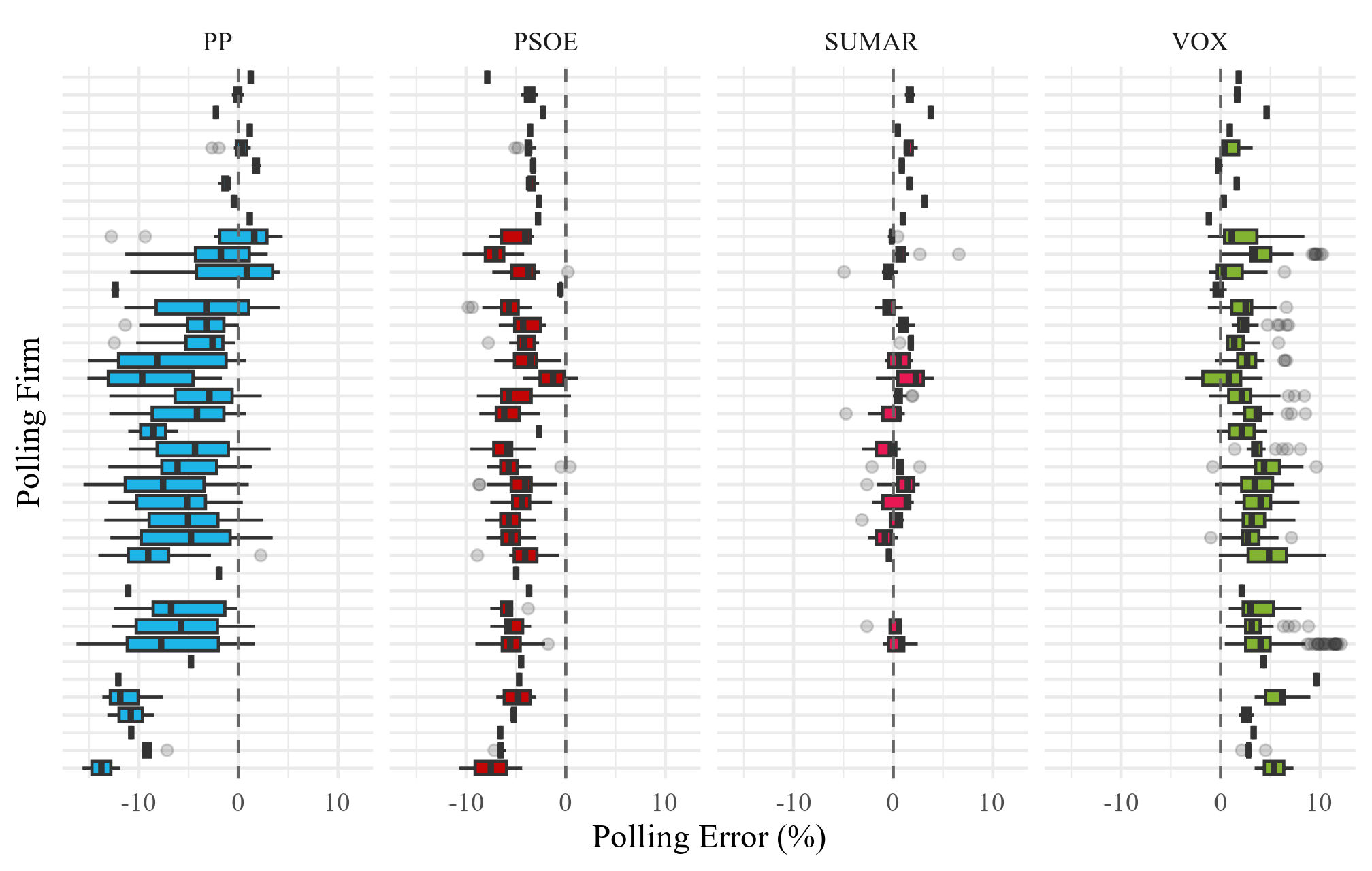
# avg estimate and actual result  
party\_summary <- data\_2023\_EDA %>%   
 group\_by(abbrev\_candidacies) %>%   
 summarise(  
 avg\_poll = mean(estimated\_voting),  
 avg\_poll\_label = paste0(round(avg\_poll, 1), "%"),  
 actual\_result = mean(voting\_results\_pct),  
 actual\_label = paste0(round(actual\_result, 1), "%"),  
 .groups = "drop"  
 )  
  
# more space between bars  
party\_summary <- party\_summary %>%  
 mutate(x\_position = as.numeric(as.factor(abbrev\_candidacies)) \* 1.5)   
  
data\_2023\_EDA\_x <- data\_2023\_EDA %>%  
 mutate(x\_position = as.numeric(as.factor(abbrev\_candidacies)) \* 1.5)  
  
figure\_4\_2 <- ggplot() +  
   
 # bar: actual results  
 geom\_col(data = party\_summary,  
 aes(x = x\_position, y = actual\_result, fill = abbrev\_candidacies),  
 width = 0.6,   
 alpha = 0.5) +   
   
 geom\_text(data = party\_summary,  
 aes(x = x\_position, y = -1, label = actual\_label, colour = abbrev\_candidacies), fontface = "bold", size = 3) +  
  
 # points: individual poll estimates  
 geom\_jitter(data = data\_2023\_EDA\_x,  
 aes(x = x\_position, y = estimated\_voting, colour = abbrev\_candidacies),  
 width = 0.2, size = 0.7) +  
  
 # segments line: avg estimated vote share total  
 geom\_segment(data = party\_summary,  
 aes(x = x\_position,   
 xend = x\_position + 0.8,  
 y = avg\_poll, yend = avg\_poll),  
 colour = "black", linewidth = 0.7) +  
  
 geom\_text(data = party\_summary,  
 aes(x = x\_position + 1.1,   
 y = avg\_poll,   
 label = avg\_poll\_label),  
 colour = "black",   
 fontface = "bold",   
 size = 4,  
 hjust = 1, # aligns to the right  
 vjust = -0.4) + # sits on top of the line  
   
 # custom spacing   
 scale\_x\_continuous(  
 breaks = party\_summary$x\_position,  
 labels = party\_summary$abbrev\_candidacies  
 ) +  
   
 scale\_fill\_manual(values = party\_colors) +  
 scale\_colour\_manual(values = party\_colors) +  
  
 labs(  
 x = "Party",  
 y = "Vote Share (%)"  
 ) +  
   
 theme\_minimal(base\_family = "Times New Roman") +  
 theme(  
 axis.text.x = element\_text(angle = 0, hjust = 1),  
 panel.grid.major.x = element\_blank(),   
 panel.grid.minor.x = element\_blank(),   
 legend.position = "none"  
 )  
  
figure\_4\_2



ggsave("visual exports/figure\_4\_2.png", figure\_4\_2,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Figure 4.3

figure\_4\_3 <- data\_2023\_EDA %>%   
 ggplot(aes(x = fct\_reorder(polling\_firm, error, .fun = median),   
 y = error,   
 fill = abbrev\_candidacies)) +   
 geom\_boxplot(outlier.alpha = 0.2) +  
 geom\_hline(yintercept = 0, linetype = "dashed", colour = "grey40") +  
 facet\_wrap(~abbrev\_candidacies, nrow = 1) +  
 scale\_fill\_manual(values = party\_colors) +   
 coord\_flip() +  
 labs(   
 x = "Polling Firm",  
 y = "Polling Error (%)"  
 ) +  
 theme\_minimal(base\_family = "Times New Roman") +   
 theme(  
 axis.text.y = element\_blank(),   
 legend.position = "none"   
 )  
  
figure\_4\_3



ggsave("visual exports/figure\_4\_3.png", figure\_4\_3,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

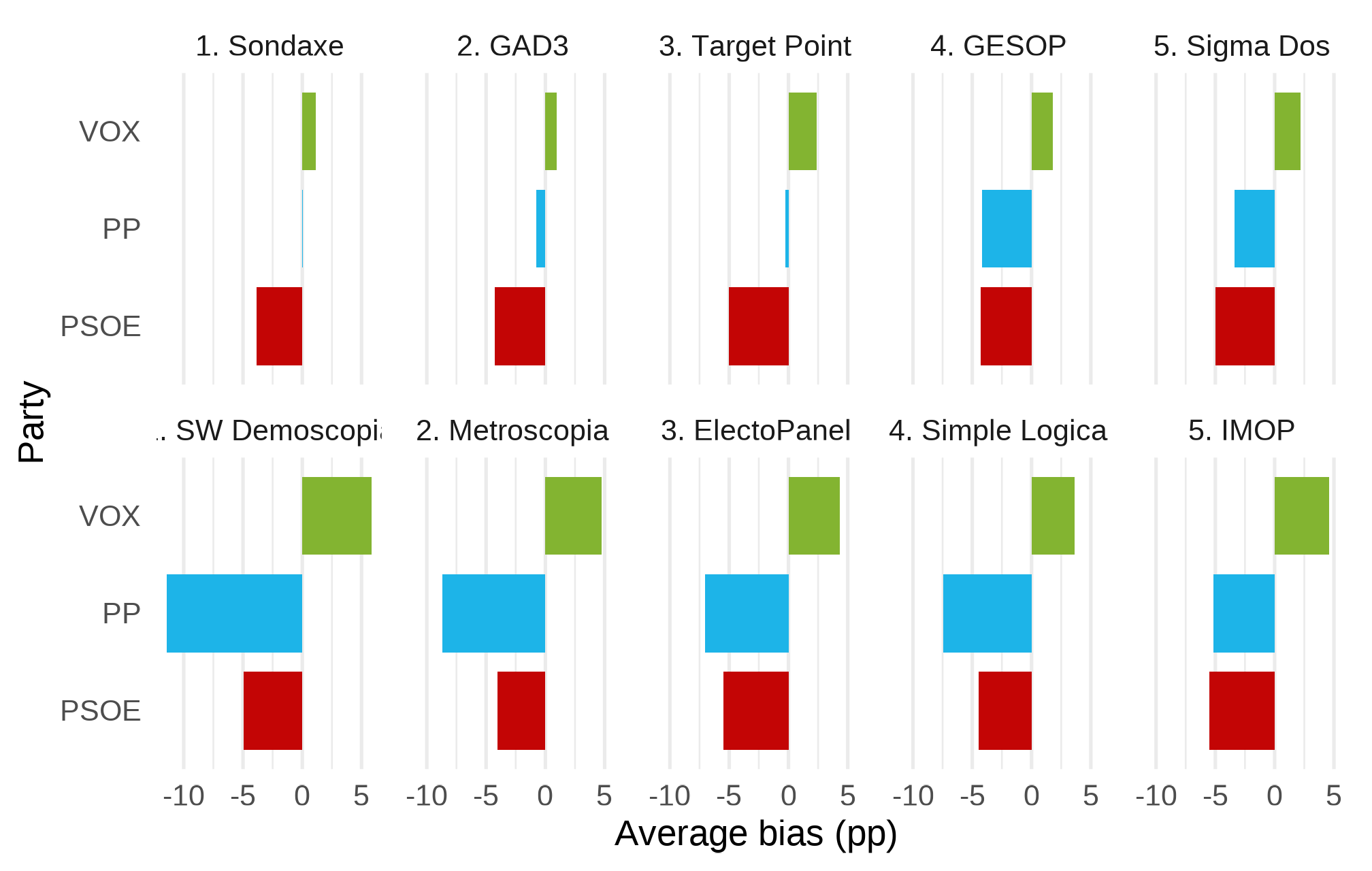
# Table 4.2

keep\_historical <- vis\_hist %>%   
 distinct(polling\_firm) %>%   
 pull(polling\_firm)   
  
# data for plots   
plots\_23 <- data\_2023\_eval\_full %>%  
 filter(  
 abbrev\_candidacies %in% c("VOX", "PP", "PSOE"),  
 polling\_firm %in% keep\_historical  
 ) %>%  
 mutate(error = estimated\_voting - voting\_results\_pct)  
  
he\_23 <- plots\_23 %>%  
 group\_by(polling\_firm, abbrev\_candidacies) %>%  
 summarise(error\_avg = mean(error, na.rm = TRUE), .groups = "drop")  
  
# mae (party-balanced, it is based on error\_avg)  
mae\_23 <- he\_23 %>%  
 group\_by(polling\_firm) %>%  
 summarise(mae = mean(abs(error\_avg), na.rm = TRUE), .groups = "drop") %>%  
 arrange(desc(mae))  
  
table\_4\_2 <- # top 5 and worst 5 (with rank within each group)  
top\_5 <- mae\_23 %>% arrange(mae) %>% slice\_head(n = 5) %>%  
 mutate(group = "Top 5", rank = row\_number())  
  
worst\_5 <- mae\_23 %>% arrange(desc(mae)) %>% slice\_head(n = 5) %>%  
 mutate(group = "Worst 5", rank = row\_number())  
  
# ranks  
rank\_table <- bind\_rows(top\_5, worst\_5) %>%  
 mutate(ranked\_label = paste0(rank, ". ", polling\_firm)) %>%  
 arrange(match(group, c("Top 5","Worst 5")), rank)  
  
table\_4\_2 <- rank\_table %>%  
 select(Rank = ranked\_label, MAE = mae, Group = group)  
  
flextable(table\_4\_2) %>%  
 autofit()

| Rank | MAE | Group |
| --- | --- | --- |
| 1. Sondaxe | 1.665515 | Top 5 |
| 2. GAD3 | 1.993844 | Top 5 |
| 3. Target Point | 2.564500 | Top 5 |
| 4. GESOP | 3.420750 | Top 5 |
| 5. Sigma Dos | 3.514255 | Top 5 |
| 1. SW Demoscopia | 7.390462 | Worst 5 |
| 2. Metroscopia | 5.812417 | Worst 5 |
| 3. ElectoPanel | 5.619337 | Worst 5 |
| 4. Simple Logica | 5.189415 | Worst 5 |
| 5. IMOP | 5.091815 | Worst 5 |

# Figure 4.4

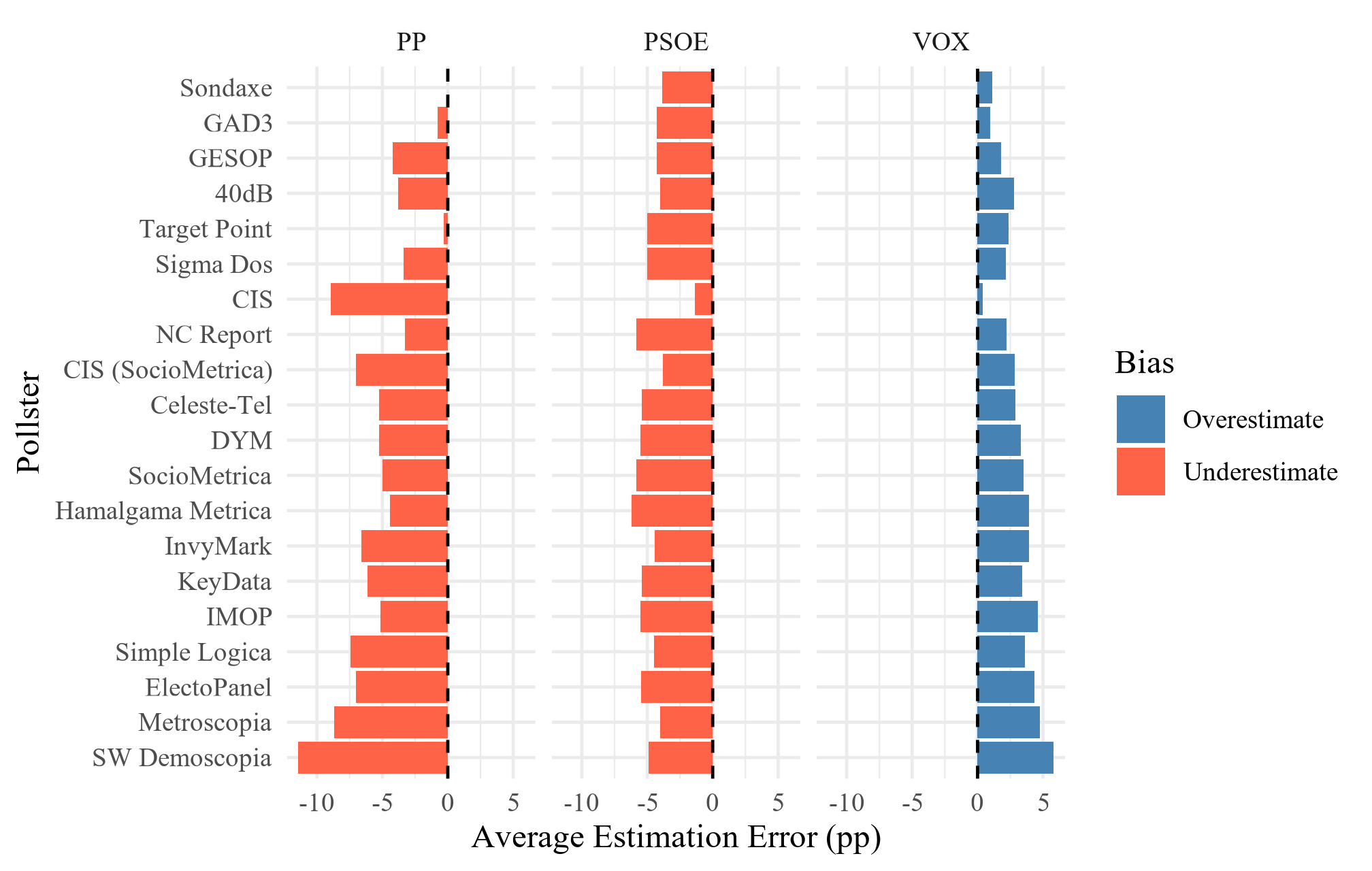
# Plot   
he\_ranked <- he\_23 %>%  
 semi\_join(rank\_table, by = "polling\_firm") %>%  
 left\_join(rank\_table %>% select(polling\_firm, group, ranked\_label),  
 by = "polling\_firm")  
  
# facet order: first the top row (Top 5), then the bottom (Worst 5)  
panel\_levels <- c(  
 rank\_table %>% filter(group == "Top 5") %>% arrange(rank) %>% pull(ranked\_label),  
 rank\_table %>% filter(group == "Worst 5") %>% arrange(rank) %>% pull(ranked\_label)  
)  
  
he\_ranked <- he\_ranked %>%  
 mutate(panel = ranked\_label,  
 panel = factor(panel, levels = panel\_levels))  
  
figure\_4\_4 <- ggplot(he\_ranked,  
 aes(x = fct\_reorder(abbrev\_candidacies, error\_avg), y = error\_avg,  
 fill = abbrev\_candidacies)) +  
 geom\_col(width = 0.8) +  
 scale\_fill\_manual(values = party\_colors, guide = "none") +  
 coord\_flip() +  
 facet\_wrap(~ panel, nrow = 2) +  
 labs(  
 x = "Party", y = "Average bias (pp)"  
 ) +  
 theme\_minimal(base\_size = 12) +  
 theme(  
 panel.grid.major.y = element\_blank(),  
 panel.grid.minor.y = element\_blank()  
 )  
   
figure\_4\_4



ggsave("visual exports/figure\_4\_4.png", figure\_4\_4,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Figure 4.5

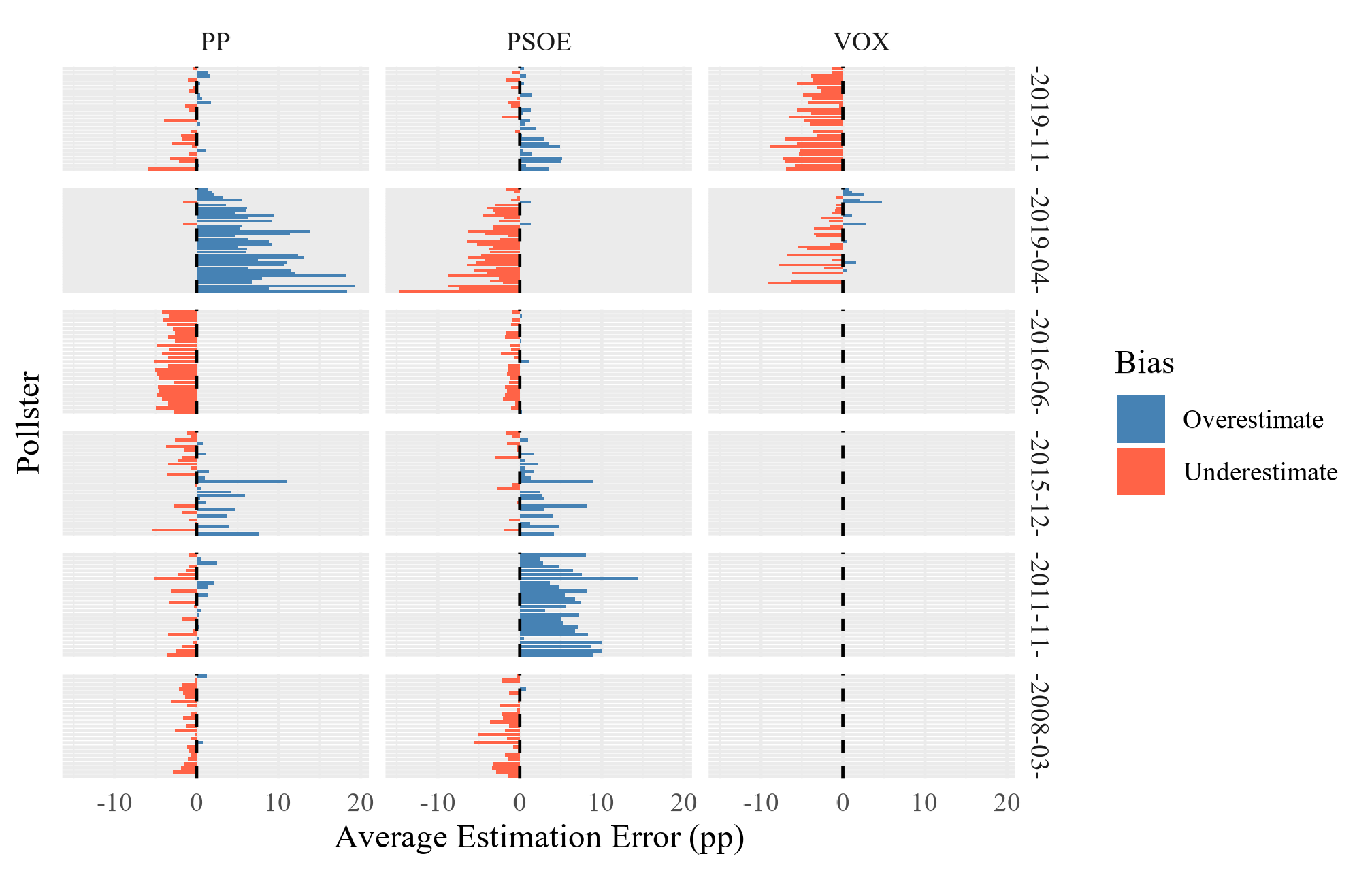
# filtering   
plot23 <- data\_2023\_eval\_full %>%  
 filter(  
 abbrev\_candidacies %in% c("VOX", "PP", "PSOE"),  
 polling\_firm %in% keep\_historical  
 ) %>%  
 mutate(error = estimated\_voting - voting\_results\_pct)  
  
# mae   
mae\_overall\_2023 <- plot23 %>%  
 group\_by(polling\_firm) %>%  
 summarise(mae\_overall = mean(abs(error), na.rm = TRUE), .groups = "drop")  
  
# house effects  
he\_summary\_2023 <- plot23 %>%  
 group\_by(polling\_firm, abbrev\_candidacies) %>%  
 summarise(error\_avg = mean(error, na.rm = TRUE), .groups = "drop") %>%  
 left\_join(mae\_overall\_2023, by = "polling\_firm") %>%  
 mutate(  
 bias\_direction = if\_else(error\_avg > 0, "Overestimate", "Underestimate"),  
 # ranked by overall MAE; set .desc = TRUE to put least biased at the top  
 polling\_firm = fct\_reorder(polling\_firm, mae\_overall, .desc = TRUE)  
 )  
  
  
# plot   
figure\_4\_5 <- ggplot(he\_summary\_2023, aes(x = error\_avg, y = polling\_firm, fill = bias\_direction)) +  
 geom\_col() +  
 geom\_vline(xintercept = 0, linetype = "dashed") +  
 facet\_wrap(~ abbrev\_candidacies) + # same order across facets  
 scale\_fill\_manual(values = c("Overestimate" = "steelblue", "Underestimate" = "tomato")) +  
 labs(  
 x = "Average Estimation Error (pp)",  
 y = "Pollster",  
 fill = "Bias"  
 ) +  
 theme\_minimal(base\_family = "Times New Roman")  
  
figure\_4\_5



ggsave("visual exports/figure\_4\_5.png", figure\_4\_5,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Figure 4.6

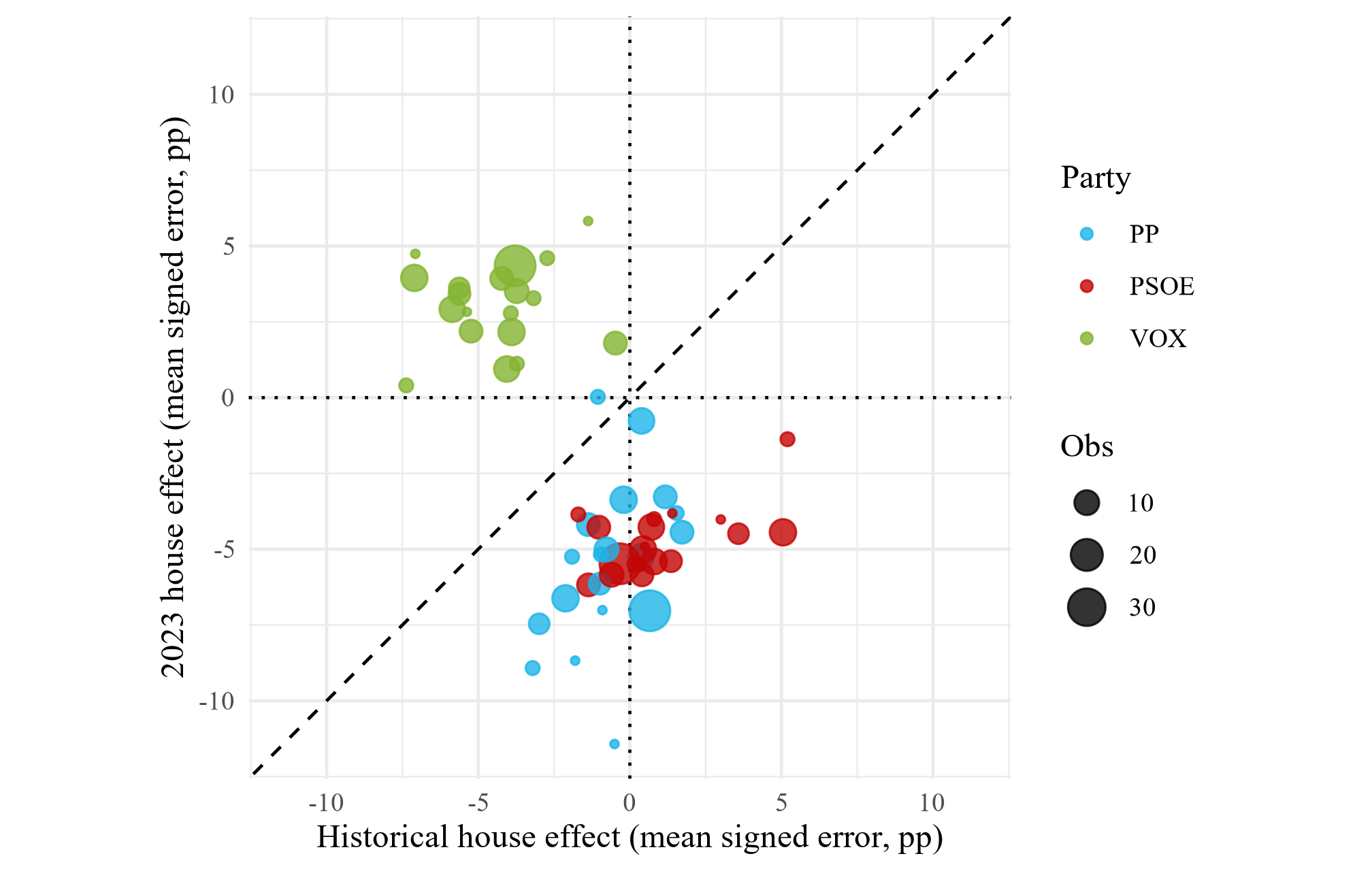
# filter  
plot\_hist <- vis\_hist %>%  
 filter(  
 abbrev\_candidacies %in% c("VOX", "PP", "PSOE"),  
 polling\_firm %in% keep\_historical  
 ) %>%  
 mutate(error = estimated\_voting - voting\_results\_pct)  
  
# last 6 elections   
recent6\_ids <- plot\_hist %>%  
 group\_by(id\_elec) %>%  
 summarise(  
 election\_date = as.Date(sub("^02-", "", unique(id\_elec))),   
 elec\_date = max(election\_date, na.rm = TRUE), .groups = "drop") %>%  
 arrange(desc(elec\_date)) %>%  
 slice\_head(n = 6) %>%  
 pull(id\_elec)  
  
  
plot\_hist <- plot\_hist %>%  
 filter(id\_elec %in% recent6\_ids) %>%  
 mutate(id\_elec = factor(id\_elec, levels = recent6\_ids))  
  
# mae  
mae\_overall\_hist <- plot\_hist %>%  
 group\_by(polling\_firm) %>%  
 summarise(mae\_overall = mean(abs(error), na.rm = TRUE), .groups = "drop")  
  
# house effects   
he\_summary\_hist <- plot\_hist %>%  
 group\_by(id\_elec, polling\_firm, abbrev\_candidacies) %>%  
 summarise(error\_avg = mean(error, na.rm = TRUE), .groups = "drop") %>%  
 left\_join(mae\_overall\_hist, by = "polling\_firm") %>%  
 mutate(  
 bias\_direction = if\_else(error\_avg > 0, "Overestimate", "Underestimate"),  
 polling\_firm = fct\_reorder(polling\_firm, mae\_overall, .desc = TRUE) # least biased at top  
 )  
  
# Plot   
figure\_4\_6 <- ggplot(he\_summary\_hist,  
 aes(x = error\_avg, y = polling\_firm, fill = bias\_direction)) +  
 geom\_col() +  
 geom\_vline(xintercept = 0, linetype = "dashed") +  
 facet\_grid(id\_elec ~ abbrev\_candidacies, scales = "free\_y") +  
 scale\_fill\_manual(values = c(Overestimate = "steelblue", Underestimate = "tomato")) +  
 labs(  
 x = "Average Estimation Error (pp)",  
 y = "Pollster",  
 fill = "Bias"  
 ) +  
 theme\_minimal(base\_family = "Times New Roman") +   
 theme(  
 axis.text.y = element\_blank()  
 )  
  
figure\_4\_6



ggsave("visual exports/figure\_4\_6.png", figure\_4\_6,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Figure 4.7

n\_polls23 <- Inf  
  
# preceeding election to 2023  
election\_before\_2023 <- vis\_hist %>%   
 distinct(id\_elec) %>%   
 slice\_head() %>%   
 pull()  
  
# historical house effects  
hist\_he <- vis\_hist %>%  
 filter(abbrev\_candidacies %in% c("PP","PSOE","VOX"),  
 polling\_firm %in% keep\_historical,   
 id\_elec %in% election\_before\_2023) %>%  
 mutate(error = estimated\_voting - voting\_results\_pct) %>%  
 group\_by(polling\_firm, abbrev\_candidacies) %>%  
 summarise(error\_hist = mean(error, na.rm = TRUE),  
 n\_hist = n(), .groups = "drop")  
  
# 2023 house effects   
plot23 <- data\_2023\_eval\_full %>%  
 filter(abbrev\_candidacies %in% c("PP","PSOE","VOX"),  
 polling\_firm %in% keep\_historical) %>%  
 mutate(  
 error = estimated\_voting - voting\_results\_pct,  
 poll\_date = coalesce(fieldwork\_end, fieldwork\_start)  
 ) %>%   
 filter(!is.na(poll\_date)) %>%  
 arrange(desc(poll\_date)) %>%  
 slice\_head(n = n\_polls23)  
  
he\_2023 <- plot23 %>%  
 group\_by(polling\_firm, abbrev\_candidacies) %>%  
 summarise(error\_2023 = mean(error, na.rm = TRUE),  
 n\_2023 = n(), .groups = "drop")  
  
# join, Plot  
stab <- hist\_he %>%  
 inner\_join(he\_2023, by = c("polling\_firm","abbrev\_candidacies"))  
  
# symmetric limits & 1:1 aspect  
rng <- max(abs(c(stab$error\_hist, stab$error\_2023)), na.rm = TRUE)  
  
figure\_4\_7 <- ggplot(stab, aes(x = error\_hist, y = error\_2023, colour = abbrev\_candidacies)) +  
 geom\_abline(slope = 1, intercept = 0, linetype = "dashed") +  
 geom\_hline(yintercept = 0, linetype = "dotted") +  
 geom\_vline(xintercept = 0, linetype = "dotted") +  
 geom\_point(aes(size = pmin(n\_hist, n\_2023)), alpha = 0.8) +  
   
 scale\_color\_manual(values = party\_colors) +  
 coord\_equal(xlim = c(-rng, rng), ylim = c(-rng, rng)) +  
 labs(  
 x = "Historical house effect (mean signed error, pp)",  
 y = "2023 house effect (mean signed error, pp)",  
 colour = "Party",  
 size = "Obs"  
 ) +  
 theme\_minimal(base\_family = "Times New Roman")  
  
figure\_4\_7



ggsave("visual exports/figure\_4\_7.png", figure\_4\_7,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Table 4.3

comparison\_new\_pollsters <- data\_2023\_eval\_full %>%   
 filter(is\_new\_pollster, first\_time == 0) %>% # new pollster, historical parties   
 summarise(  
 .by = abbrev\_candidacies,  
 MAE\_raw = mean(abs(estimated\_voting - voting\_results\_pct), na.rm = TRUE),  
 MAE\_avg = mean(abs(debiased\_estimate\_avg - voting\_results\_pct), na.rm = TRUE),   
 MAE\_bMAE = mean(abs(debiased\_estimate\_bMAE - voting\_results\_pct), na.rm = TRUE),  
 MAE\_bRMSE = mean(abs(debiased\_estimate\_bRMSE - voting\_results\_pct), na.rm = TRUE)  
 )  
  
comparison\_new\_pollsters\_long <- comparison\_new\_pollsters %>%   
 pivot\_longer(  
 cols = -abbrev\_candidacies,  
 names\_to = c("metric", "method"),  
 names\_sep = "\_",  
 values\_to = "value"  
 )  
  
comparison\_new\_pollsters\_long$method <- factor(  
 comparison\_new\_pollsters\_long$method,   
 levels = c("raw", "avg", "bMAE", "bRMSE"))  
  
  
table\_4\_3 <- comparison\_new\_pollsters %>%  
 select(Parties = abbrev\_candidacies, MAE\_raw = MAE\_raw, MAE\_hist = MAE\_avg, MAE\_bMAE = MAE\_bMAE, MAE\_bRMSE = MAE\_bRMSE)  
  
flextable(table\_4\_3) %>%  
 autofit()

| Parties | MAE\_raw | MAE\_hist | MAE\_bMAE | MAE\_bRMSE |
| --- | --- | --- | --- | --- |
| PSOE | 5.240140 | 5.466435 | 4.130951 | 4.135696 |
| PP | 5.100411 | 6.494748 | 4.734071 | 4.722943 |
| VOX | 3.593341 | 6.903239 | 2.071769 | 2.081066 |

# Table 4.4

comparison\_hist\_pollsters <- data\_2023\_eval\_full %>%   
 filter(!is\_new\_pollster, first\_time == 0) %>% # established pollsters, historical parties  
 summarise(  
 .by = abbrev\_candidacies,  
 MAE\_raw = mean(abs(estimated\_voting - voting\_results\_pct), na.rm = TRUE),# baseline MAE of the raw (uncorrected) estimated\_voting  
 MAE\_avg = mean(abs(debiased\_estimate\_avg - voting\_results\_pct), na.rm = TRUE),   
 MAE\_bMAE = mean(abs(debiased\_estimate\_bMAE - voting\_results\_pct), na.rm = TRUE),  
 MAE\_bRMSE = mean(abs(debiased\_estimate\_bRMSE - voting\_results\_pct), na.rm = TRUE)  
 )  
  
comparison\_hist\_pollsters\_long <- comparison\_hist\_pollsters %>%   
 pivot\_longer(  
 cols = -abbrev\_candidacies,  
 names\_to = c("metric", "method"),  
 names\_sep = "\_",  
 values\_to = "value"  
 )  
  
comparison\_hist\_pollsters\_long$method <- factor(  
 comparison\_hist\_pollsters\_long$method,   
 levels = c("raw", "avg", "bMAE", "bRMSE"))  
  
  
  
table\_4\_4 <- comparison\_hist\_pollsters %>%  
 select(Parties = abbrev\_candidacies, MAE\_raw = MAE\_raw, MAE\_hist = MAE\_avg, MAE\_bMAE = MAE\_bMAE, MAE\_bRMSE = MAE\_bRMSE)  
   
flextable(table\_4\_4) %>%  
 autofit()

| Parties | MAE\_raw | MAE\_hist | MAE\_bMAE | MAE\_bRMSE |
| --- | --- | --- | --- | --- |
| PSOE | 4.972909 | 4.445226 | 3.851331 | 3.856242 |
| PP | 6.092535 | 7.821524 | 3.773900 | 3.773585 |
| VOX | 3.378430 | 6.291359 | 1.747708 | 1.758413 |

# Table 4.5

comparison\_all\_pollsters <- data\_2023\_eval\_full %>%   
 summarise(  
 .by = abbrev\_candidacies,  
 MAE\_raw = mean(abs(estimated\_voting - voting\_results\_pct), na.rm = TRUE), # baseline MAE of the raw (uncorrected) estimated\_voting  
 MAE\_avg = mean(abs(debiased\_estimate\_avg - voting\_results\_pct), na.rm = TRUE),   
 MAE\_bMAE = mean(abs(debiased\_estimate\_bMAE - voting\_results\_pct), na.rm = TRUE),  
 MAE\_bRMSE = mean(abs(debiased\_estimate\_bRMSE - voting\_results\_pct), na.rm = TRUE)  
 )  
  
comparison\_all\_pollsters\_long <- comparison\_all\_pollsters %>%   
 pivot\_longer(  
 cols = -abbrev\_candidacies,  
 names\_to = c("metric", "method"),  
 names\_sep = "\_",  
 values\_to = "value"  
 )  
  
  
comparison\_all\_pollsters\_long$method <- factor(  
 comparison\_all\_pollsters\_long$method,   
 levels = c("raw", "avg", "bMAE", "bRMSE"))  
  
lvl\_methods <- c("raw","avg","bMAE","bRMSE")  
  
# just in case:   
comparison\_new\_pollsters\_long <- comparison\_new\_pollsters\_long %>%   
 mutate(method = factor(method, levels = lvl\_methods))  
  
comparison\_hist\_pollsters\_long <- comparison\_hist\_pollsters\_long %>%   
 mutate(method = factor(method, levels = lvl\_methods))  
  
comparison\_all\_pollsters\_long <- comparison\_all\_pollsters\_long %>%   
 mutate(method = factor(method, levels = lvl\_methods))  
  
  
  
comparison\_3\_long <- bind\_rows(  
 comparison\_new\_pollsters\_long %>% mutate(pollster\_type = "New"),  
 comparison\_hist\_pollsters\_long %>% mutate(pollster\_type = "Historical"),   
 comparison\_all\_pollsters\_long %>% mutate(pollster\_type = "All"),   
)  
  
# summary across parties by pollster type × method  
summary\_table <- comparison\_3\_long %>%  
 group\_by(pollster\_type, metric, method) %>%  
 summarise(  
 mean\_error = mean(value, na.rm = TRUE),  
 median\_error = median(value, na.rm = TRUE),  
 .groups = "drop"  
 ) %>%  
 arrange(metric, pollster\_type, method)  
  
table\_4\_5 <- summary\_table %>%  
 group\_by(pollster\_type, metric) %>%  
 mutate(  
 min\_mean = min(mean\_error, na.rm = TRUE),  
 min\_median = min(median\_error, na.rm = TRUE),  
 is\_mean\_winner = is.finite(mean\_error) & mean\_error <= min\_mean + 1e-12,  
 is\_median\_winner = is.finite(median\_error) & median\_error <= min\_median + 1e-12,  
 mean\_error\_disp = ifelse(is\_mean\_winner, sprintf("%.3f ★", mean\_error),  
 sprintf("%.3f", mean\_error)),  
 median\_error\_disp = ifelse(is\_median\_winner, sprintf("%.3f ★", median\_error),  
 sprintf("%.3f", median\_error))  
 ) %>%  
 ungroup() %>%  
 arrange(metric, pollster\_type, method) %>%  
 select(pollster\_type, metric, method, mean\_error\_disp, median\_error\_disp)  
  
table\_4\_5 <- table\_4\_5 %>%  
 filter(pollster\_type == "All") %>%   
 select(Method = method, "Mean Error" = mean\_error\_disp, "Median Error" = median\_error\_disp)  
  
table\_4\_5

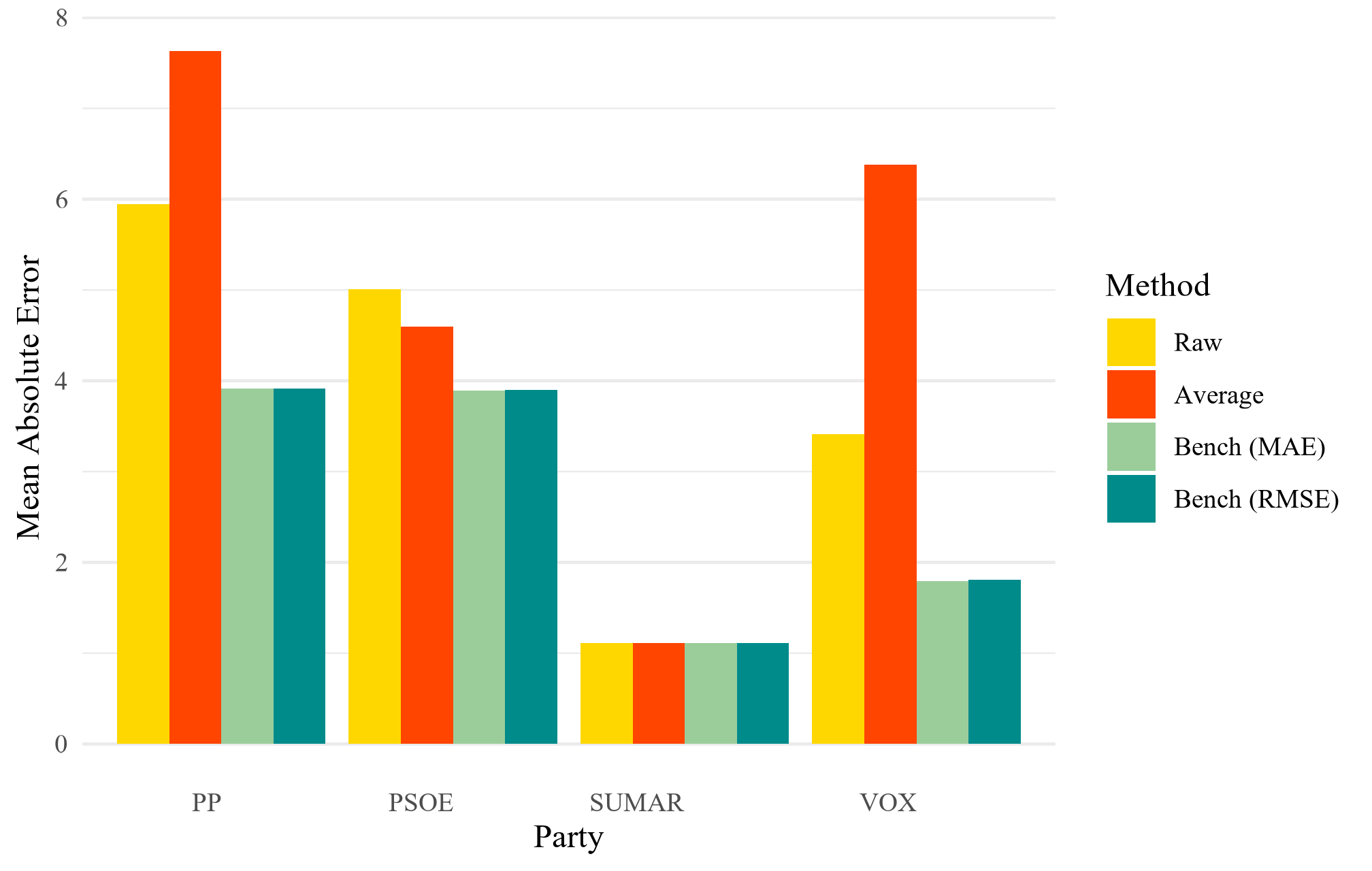
# A tibble: 4 × 3  
 Method `Mean Error` `Median Error`  
 <fct> <chr> <chr>   
1 raw 3.870 4.211   
2 avg 4.928 5.487   
3 bMAE 2.677 ★ 2.843 ★   
4 bRMSE 2.681 2.851

flextable(table\_4\_5) %>%  
 autofit()

| Method | Mean Error | Median Error |
| --- | --- | --- |
| raw | 3.870 | 4.211 |
| avg | 4.928 | 5.487 |
| bMAE | 2.677 ★ | 2.843 ★ |
| bRMSE | 2.681 | 2.851 |

# Figure 4.8

figure\_4\_8 <-   
ggplot(  
 comparison\_all\_pollsters\_long %>% filter(metric == "MAE"),  
 aes(x = abbrev\_candidacies, y = value, fill = method)  
) +  
 geom\_col(position = "dodge") +  
 scale\_fill\_manual(  
 values = c("raw" = "gold1", "avg" = "orangered",  
 "bMAE" = "darkseagreen3", "bRMSE" = "darkcyan"),  
 labels = c("Raw", "Average", "Bench (MAE)", "Bench (RMSE)")  
 ) +  
 labs(  
 x = "Party", y = "Mean Absolute Error", fill = "Method"  
 ) +  
 theme\_minimal(base\_family = "Times New Roman") +  
 theme(  
 axis.text.x = element\_text(angle = 0, hjust = 1),   
 panel.grid.major.x = element\_blank(),   
 panel.grid.minor.x = element\_blank(),  
 )  
  
figure\_4\_8



ggsave("visual exports/figure\_4\_8.png", figure\_4\_8,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Table 4.6

# avg\_result  
avg\_results\_2023 <- data\_2023\_consensus %>%  
 group\_by(abbrev\_candidacies) %>%  
 summarise(  
 avg\_prediction = mean(estimated\_voting, na.rm = TRUE),  
 .groups = "drop"  
 )  
  
results\_consensus <- data\_2023\_consensus %>%   
 group\_by(abbrev\_candidacies) %>%   
 summarise(   
   
 # resultados imaginarios = ponderamos el resultado de cada encuesta en funcion de nuestros weights   
 weighted\_results\_MAE\_rating = weighted.mean(estimated\_voting, combined\_weight\_MAE, na.rm = TRUE),   
 weighted\_results\_RMSE\_rating = weighted.mean(estimated\_voting, combined\_weight\_RMSE, na.rm = TRUE),   
 .groups = "drop"  
 )   
  
  
# added to the benchmark vs actual dataset  
bench\_vs\_actual <- results\_consensus %>%  
 left\_join(  
 vis\_2023 %>% distinct(abbrev\_candidacies, voting\_results\_pct),  
 by = c("abbrev\_candidacies")  
 ) %>%  
 left\_join(avg\_results\_2023, by = c("abbrev\_candidacies"))  
  
table\_4\_6 <- bench\_vs\_actual %>%  
 select(Parties = abbrev\_candidacies, "Weighted results (MAE)" = weighted\_results\_MAE\_rating, "Weighted results (RMSE)"= weighted\_results\_RMSE\_rating, "Actual vote share" = voting\_results\_pct, "Avg predictions" = avg\_prediction)  
  
table\_4\_6

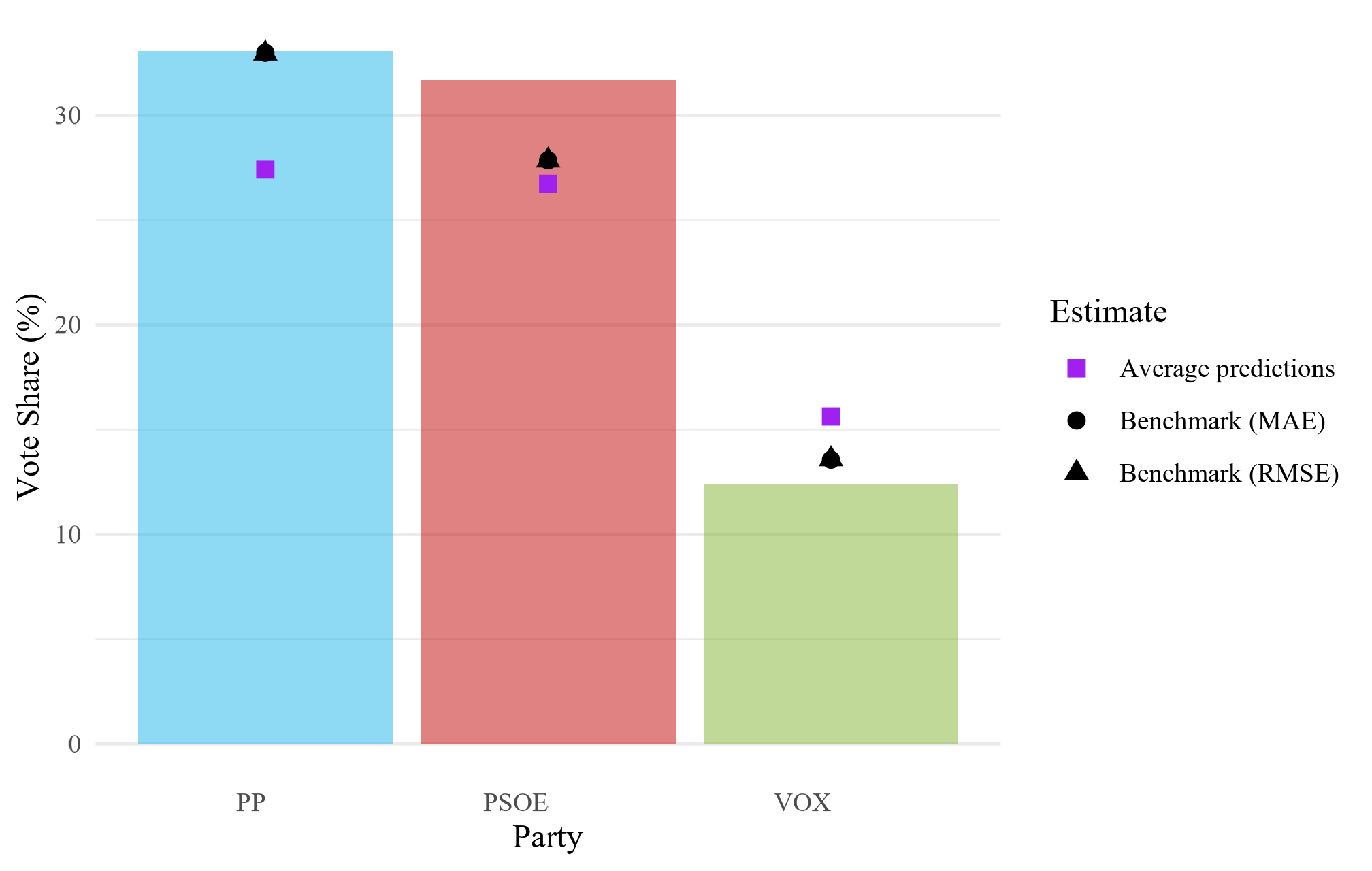
# A tibble: 3 × 5  
 Parties `Weighted results (MAE)` `Weighted results (RMSE)` `Actual vote share`  
 <chr> <dbl> <dbl> <dbl>  
1 PP 33.0 33.0 33.1  
2 PSOE 27.8 27.8 31.7  
3 VOX 13.6 13.6 12.4  
# ℹ 1 more variable: `Avg predictions` <dbl>

flextable(table\_4\_6) %>%  
 autofit()

| Parties | Weighted results (MAE) | Weighted results (RMSE) | Actual vote share | Avg predictions |
| --- | --- | --- | --- | --- |
| PP | 32.98772 | 32.95434 | 33.057 | 27.42037 |
| PSOE | 27.84859 | 27.84361 | 31.682 | 26.72832 |
| VOX | 13.56560 | 13.58912 | 12.383 | 15.62950 |

# Figure 4.9

figure\_4\_9 <- ggplot(bench\_vs\_actual %>%  
 mutate(abbrev\_candidacies = fct\_reorder(abbrev\_candidacies, voting\_results\_pct, .desc = TRUE)),  
 aes(x = abbrev\_candidacies, group = abbrev\_candidacies)) +  
 # actual voteshare  
 geom\_col(aes(y = voting\_results\_pct, fill = abbrev\_candidacies),  
 alpha = 0.5, position = "dodge") +  
   
 # MAE pollster rating estimations  
 geom\_point(aes(y = weighted\_results\_MAE\_rating, shape = "Benchmark (MAE)"),  
 position = position\_dodge(width = 0.9), size = 2.5, colour = "black") +  
   
 # RMSE pollster rating estimations  
 geom\_point(aes(y = weighted\_results\_RMSE\_rating, shape = "Benchmark (RMSE)"),  
 position = position\_dodge(width = 0.9), size = 2.5, colour = "black") +  
   
 # Average estimation based on all polls per election (possibly reducing partisan effect?)  
 geom\_point(aes(y = avg\_prediction, shape = "Average predictions"),  
 position = position\_dodge(width = 0.9), size = 2.5, colour = "purple") +  
   
 scale\_fill\_manual(values = party\_colors, guide = "none") +  
 scale\_shape\_manual(values = c("Benchmark (MAE)" = 16, # circle  
 "Benchmark (RMSE)" = 17, # triangle  
 "Average predictions" = 15), # square  
 name = "Estimate") +  
   
 labs(  
 x = "Party", y = "Vote Share (%)"  
 ) +  
 theme\_minimal(base\_family = "Times New Roman") +  
 theme(  
 axis.text.x = element\_text(angle = 0, hjust = 1),   
 panel.grid.major.x = element\_blank(),   
 panel.grid.minor.x = element\_blank(),  
 )  
  
  
  
figure\_4\_9



ggsave("visual exports/figure\_4\_9.png", figure\_4\_9,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

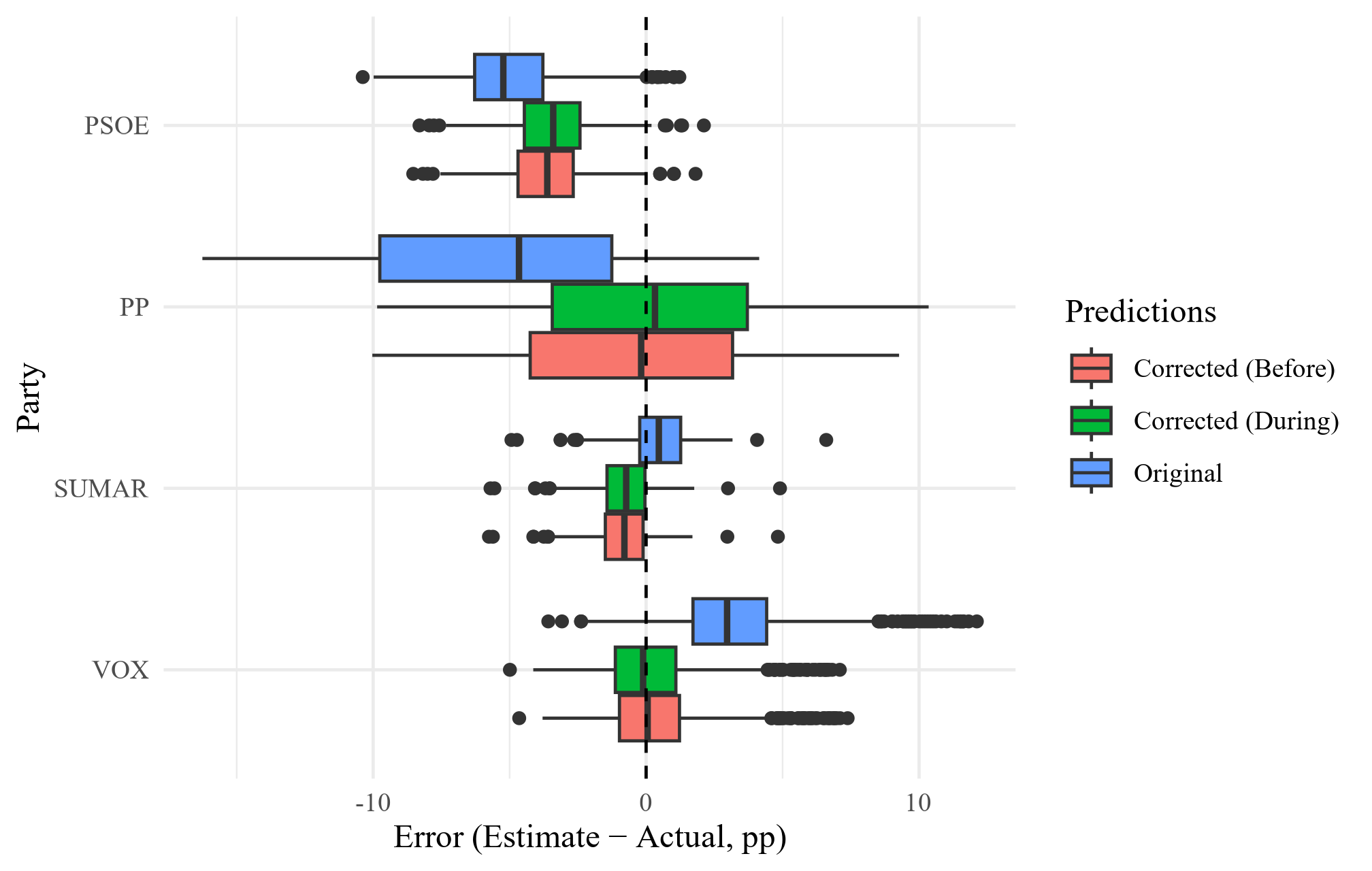
# Table 4.7

# general   
results\_reg\_2\_pct <- pred\_store\_2\_pct %>%  
 group\_by(model) %>%  
 summarise(  
 RMSE = sqrt(mean((pred - obs)^2, na.rm = TRUE)),  
 MAE = mean(abs(pred - obs), na.rm = TRUE),  
 Rsquared = cor(pred, obs, use = "complete.obs")^2,  
 .groups = "drop"  
 )  
  
# best of each in general  
rankings <- results\_reg\_2\_pct %>%  
 mutate(  
 rank\_mae = rank(MAE, ties.method = "min"),  
 rank\_rmse = rank(RMSE, ties.method = "min"),  
 rank\_r2 = rank(-Rsquared, ties.method = "min") # higher is better  
 )  
  
table\_4\_7 <- rankings %>%  
 mutate(  
 MAE = sprintf("%.3f%s", MAE, ifelse(rank\_mae == 1, " ★", ifelse(rank\_mae == 2, " ☆", ""))),  
 RMSE = sprintf("%.3f%s", RMSE, ifelse(rank\_rmse == 1, " ★", ifelse(rank\_rmse == 2, " ☆", ""))),  
 Rsquared = sprintf("%.3f%s", Rsquared, ifelse(rank\_r2 == 1, " ★", ifelse(rank\_r2 == 2, " ☆", "")))  
 )  
  
table\_4\_7 <- table\_4\_7 %>%  
 select(Model = model, MAE = MAE, RMSE = RMSE, Rsquared = Rsquared)  
   
  
flextable(table\_4\_7) %>%  
 autofit()

| Model | MAE | RMSE | Rsquared |
| --- | --- | --- | --- |
| m\_knn\_before | 3.619 | 4.878 | 0.850 |
| m\_knn\_during | 3.777 | 5.234 | 0.828 |
| m\_lm\_baseline | 3.635 | 4.964 | 0.842 |
| m\_lm\_before | 3.489 ★ | 4.678 ★ | 0.860 ★ |
| m\_lm\_during | 3.502 ☆ | 4.689 ☆ | 0.859 ☆ |
| m\_nn\_before | 4.196 | 5.536 | 0.804 |
| m\_nn\_during | 4.401 | 6.020 | 0.773 |
| m\_rf\_before | 3.682 | 4.864 | 0.848 |
| m\_rf\_during | 3.651 | 4.822 | 0.851 |

# Figure 4.10

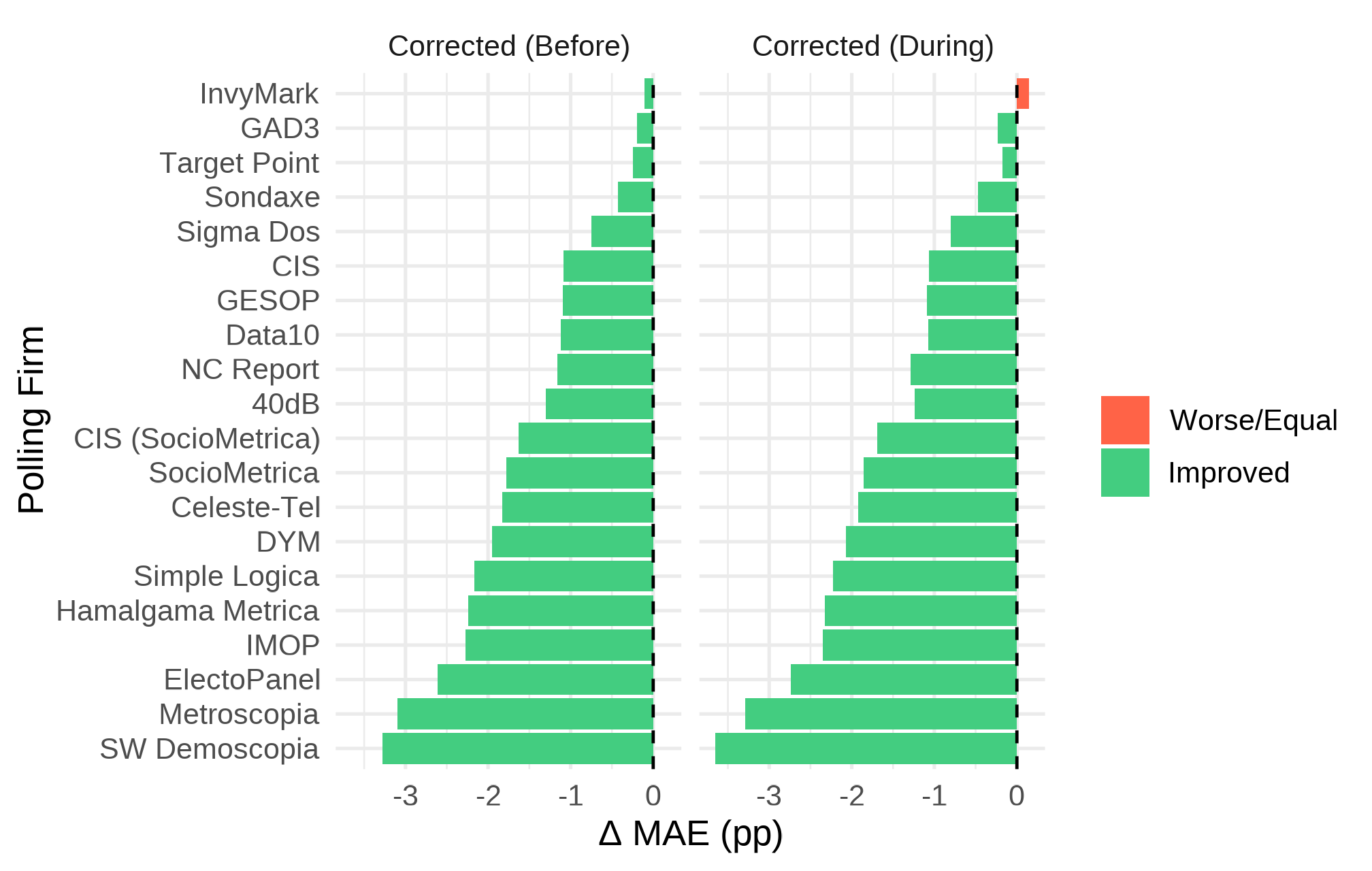
figure\_4\_10 <- complete\_2023\_2\_pct %>%  
 select(abbrev\_candidacies, error\_original, error\_lm\_before, error\_lm\_during) %>%  
 pivot\_longer(  
 c(error\_original, error\_lm\_before, error\_lm\_during),  
 names\_to = "type", values\_to = "error"  
 ) %>%  
 mutate(type = recode(type,  
 error\_original = "Original ",  
 error\_lm\_before = "Corrected (Before)",  
 error\_lm\_during = "Corrected (During)")) %>%  
 ggplot(aes(x = fct\_reorder(abbrev\_candidacies, error, .fun = median, .desc = TRUE),  
 y = error, fill = type)) +  
 geom\_boxplot(position = position\_dodge(width = 0.8)) +  
 geom\_hline(yintercept = 0, linetype = "dashed") +  
 labs(  
 x = "Party",  
 y = "Error (Estimate − Actual, pp)",  
 fill = "Predictions"  
 ) +  
 coord\_flip() +  
 theme\_minimal(base\_family = "Times New Roman")  
  
  
figure\_4\_10



ggsave("visual exports/figure\_4\_10.png", figure\_4\_10,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Figure 4.11

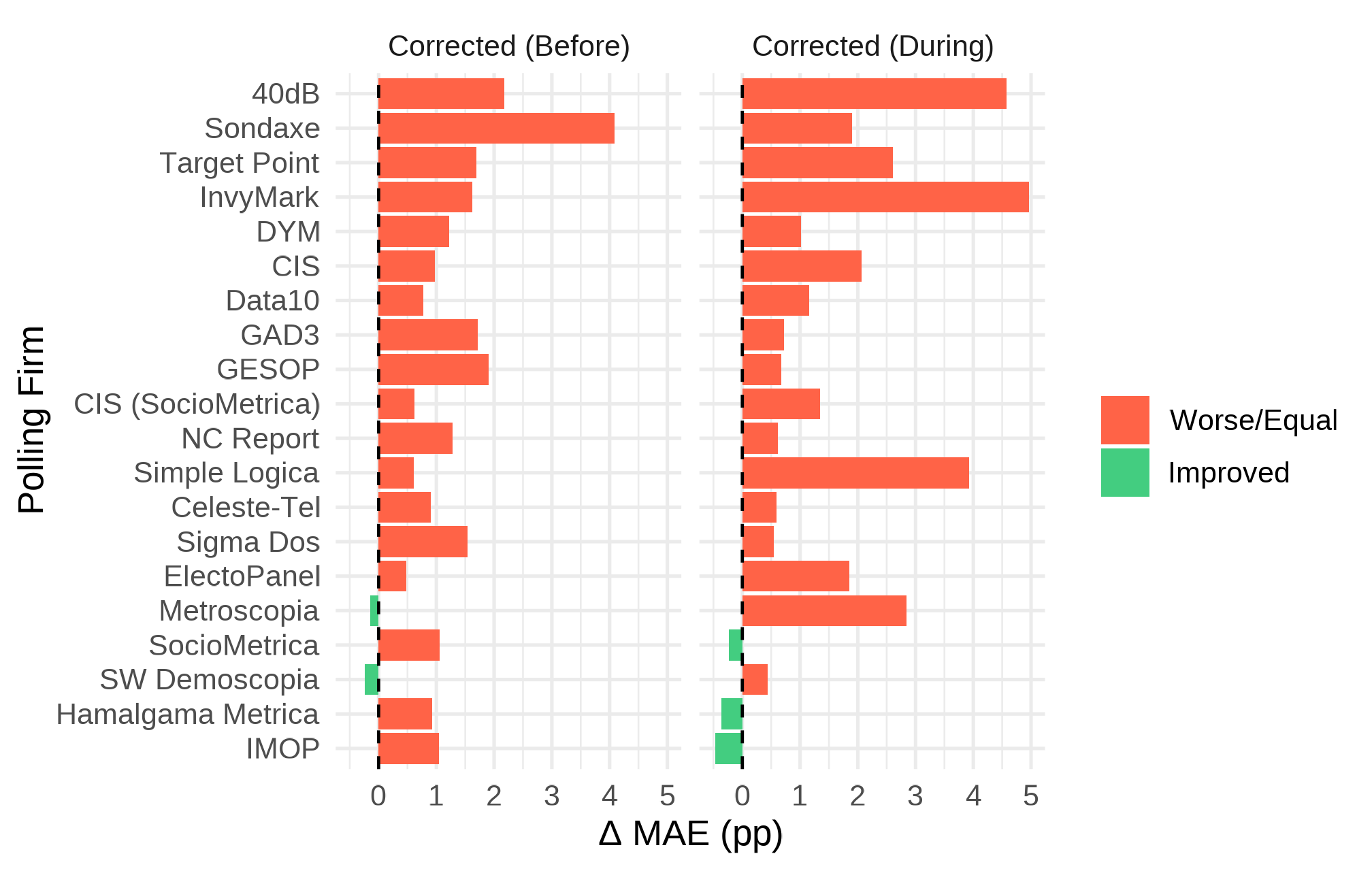
# function for per firm mae  
delta\_mae\_by\_firm <- function(df, pred\_col, label) {  
 df %>%  
 group\_by(polling\_firm) %>%  
 summarise(  
 MAE\_original = mean(abs(estimated\_voting - voting\_results\_pct), na.rm = TRUE),  
 MAE\_corrected = mean(abs(.data[[pred\_col]] - voting\_results\_pct), na.rm = TRUE),  
 .groups = "drop"  
 ) %>%  
 mutate(  
 delta = MAE\_corrected - MAE\_original, # negative = improvement  
 improved = delta < 0,  
 type = label  
 )  
}  
  
# table  
d\_before\_2\_pct <- delta\_mae\_by\_firm(complete\_2023\_2\_pct, "pred\_lm\_before", "Corrected (Before)")  
d\_during\_2\_pct <- delta\_mae\_by\_firm(complete\_2023\_2\_pct, "pred\_lm\_during", "Corrected (During)")  
  
beat\_pollsters\_both\_2\_pct <- bind\_rows(d\_before\_2\_pct, d\_during\_2\_pct)  
  
  
# ranking  
order\_by\_best\_2\_pct <- beat\_pollsters\_both\_2\_pct %>%  
 group\_by(polling\_firm) %>%  
 summarise(best\_delta = min(delta, na.rm = TRUE), .groups = "drop") %>%  
 arrange(best\_delta) %>%  
 pull(polling\_firm)  
  
  
  
# Plot  
figure\_4\_11 <- ggplot(  
 beat\_pollsters\_both\_2\_pct %>%  
 mutate(polling\_firm = factor(polling\_firm, levels = order\_by\_best\_2\_pct)),  
 aes(x = polling\_firm, y = delta, fill = improved)  
) +  
 geom\_col(position = position\_dodge(width = 0.8)) +  
 facet\_wrap(~ type) +  
 geom\_hline(yintercept = 0, linetype = "dashed") +  
 coord\_flip() +  
 scale\_fill\_manual(values = c("TRUE" = "seagreen3", "FALSE" = "tomato"),  
 labels = c("TRUE" = "Improved",  
 "FALSE" = "Worse/Equal")) +  
 labs(  
 x = "Polling Firm", y = "Δ MAE (pp)", fill = ""  
 ) +  
 theme\_minimal(base\_size = 12)  
  
figure\_4\_11



ggsave("visual exports/figure\_4\_11.png", figure\_4\_11,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Figure 5.1

# table  
d\_before\_1\_pct <- delta\_mae\_by\_firm(complete\_2023\_1\_pct, "pred\_rf\_before", "Corrected (Before)")  
d\_during\_1\_pct <- delta\_mae\_by\_firm(complete\_2023\_1\_pct, "pred\_knn\_during", "Corrected (During)")  
  
beat\_pollsters\_both\_1\_pct <- bind\_rows(d\_before\_1\_pct, d\_during\_1\_pct)  
  
  
# ranking  
order\_by\_best\_1\_pct <- beat\_pollsters\_both\_1\_pct %>%  
 group\_by(polling\_firm) %>%  
 summarise(best\_delta = min(delta, na.rm = TRUE), .groups = "drop") %>%  
 arrange(best\_delta) %>%  
 pull(polling\_firm)  
  
  
# Plot  
figure\_5\_1 <- ggplot(  
 beat\_pollsters\_both\_1\_pct %>%  
 mutate(polling\_firm = factor(polling\_firm, levels = order\_by\_best\_1\_pct)),  
 aes(x = polling\_firm, y = delta, fill = improved)  
) +  
 geom\_col(position = position\_dodge(width = 0.8)) +  
 facet\_wrap(~ type) +  
 geom\_hline(yintercept = 0, linetype = "dashed") +  
 coord\_flip() +  
 scale\_fill\_manual(values = c("TRUE" = "seagreen3", "FALSE" = "tomato"),  
 labels = c("TRUE" = "Improved",  
 "FALSE" = "Worse/Equal")) +  
 labs(  
 x = "Polling Firm", y = "Δ MAE (pp)", fill = ""  
 ) +  
 theme\_minimal(base\_size = 12)  
  
figure\_5\_1



ggsave("visual exports/figure\_5\_1.png", figure\_5\_1,  
 width = 6.3, height = 4, dpi = 320,  
 device = ragg::agg\_png)

# Appendix A

# general   
results\_reg\_1\_pct <- pred\_store\_1\_pct %>%  
 group\_by(model) %>%  
 summarise(  
 RMSE = sqrt(mean((pred - obs)^2, na.rm = TRUE)),  
 MAE = mean(abs(pred - obs), na.rm = TRUE),  
 Rsquared = cor(pred, obs, use = "complete.obs")^2,  
 .groups = "drop"  
 )  
  
results\_reg\_1\_pct %>%  
 arrange(MAE)

# A tibble: 9 × 4  
 model RMSE MAE Rsquared  
 <chr> <dbl> <dbl> <dbl>  
1 m\_rf\_during 4.86 3.33 0.857  
2 m\_knn\_before 4.92 3.35 0.854  
3 m\_knn\_during 5.14 3.41 0.841  
4 m\_rf\_before 4.87 3.44 0.856  
5 m\_lm\_before 5.40 3.69 0.824  
6 m\_lm\_during 5.42 3.69 0.823  
7 m\_lm\_baseline 5.52 3.73 0.816  
8 m\_nn\_before 5.44 4.00 0.821  
9 m\_nn\_during 6.67 5.17 0.736

# best of each in general  
best\_mae\_1\_pct <- results\_reg\_1\_pct %>%   
 slice\_min(MAE, n = 1, with\_ties = FALSE) %>%  
 pull(model)  
  
best\_rmse\_1\_pct <- results\_reg\_1\_pct %>%  
 slice\_min(RMSE, n = 1, with\_ties = FALSE) %>%  
 pull(model)  
  
best\_r2\_1\_pct <- results\_reg\_1\_pct %>%   
 slice\_max(Rsquared, n = 1, with\_ties = FALSE) %>%   
 pull(model)  
  
  
table\_appendix\_A <- results\_reg\_1\_pct %>%  
 mutate(  
 MAE = ifelse(model == best\_mae\_1\_pct,   
 sprintf("%.3f ★", MAE),   
 sprintf("%.3f", MAE)),  
   
 RMSE = ifelse(model == best\_rmse\_1\_pct,   
 sprintf("%.3f ★", RMSE),   
 sprintf("%.3f", RMSE)),  
   
 Rsquared = ifelse(model == best\_r2\_1\_pct,   
 sprintf("%.3f ★", Rsquared),  
 sprintf("%.3f", Rsquared))  
 )  
  
table\_appendix\_A

# A tibble: 9 × 4  
 model RMSE MAE Rsquared  
 <chr> <chr> <chr> <chr>   
1 m\_knn\_before 4.923 3.351 0.854   
2 m\_knn\_during 5.139 3.409 0.841   
3 m\_lm\_baseline 5.521 3.728 0.816   
4 m\_lm\_before 5.400 3.691 0.824   
5 m\_lm\_during 5.420 3.695 0.823   
6 m\_nn\_before 5.435 4.000 0.821   
7 m\_nn\_during 6.672 5.168 0.736   
8 m\_rf\_before 4.870 3.441 0.856   
9 m\_rf\_during 4.858 ★ 3.328 ★ 0.857 ★

flextable(table\_appendix\_A) %>%  
 autofit()

| model | RMSE | MAE | Rsquared |
| --- | --- | --- | --- |
| m\_knn\_before | 4.923 | 3.351 | 0.854 |
| m\_knn\_during | 5.139 | 3.409 | 0.841 |
| m\_lm\_baseline | 5.521 | 3.728 | 0.816 |
| m\_lm\_before | 5.400 | 3.691 | 0.824 |
| m\_lm\_during | 5.420 | 3.695 | 0.823 |
| m\_nn\_before | 5.435 | 4.000 | 0.821 |
| m\_nn\_during | 6.672 | 5.168 | 0.736 |
| m\_rf\_before | 4.870 | 3.441 | 0.856 |
| m\_rf\_during | 4.858 ★ | 3.328 ★ | 0.857 ★ |