Study S2A

January 28, 2025

Items

Read Data . . . 2
Variable Names . . . 4
Demographics . . . 6
Pooled Analysis . . . 7
Primary Analysis . . . 9
Secondary Analysis . . . 9
Figure S4A Code . . . 12
Figure S4B Code . . . 15
System of Equations . . . 19

Read Data

```r
# Set this to TRUE if you have API access, FALSE if using CSV
USE_API <- TRUE

if(USE_API) {
  ## Pull directly from Qualtrics API
  qual_data <- fetch_survey(surveyID='SV_3sK5vJ1hXsDPKom',
                     label = T,
                     convert = F,
                     start_date = "2022-12-01",
                     force_request = T)
} else {
  # Read the processed data directly from CSV
  d0 <- read.csv('StudyS2A.csv', check.names = F)
}

# Define the categories
females <- c('Mary Barra (CEO of General Motors)', 'Oprah Winfrey (CEO of Oprah Winfrey
↪   Network)',
             'Janet Yellen (U.S. Secretary Treasury)', 'Jane Fraser (CEO of Citigroup)',
             'Rosalind Brewer (CEO of Walgreens)')

ceos <- c("Tim Cook (CEO of Apple)", "Elon Musk (CEO of Tesla)", "Oprah Winfrey (CEO of
↪   Oprah Winfrey Network)",
          "Mary Barra (CEO of General Motors)", "Jane Fraser (CEO of Citigroup)", "Marvin
          ↪   Ellison (CEO of Lowe's)",
          "Warren Buffet (CEO of Berkshire Hathaway)", "Charles Koch (CEO of Koch
          ↪   Industries)",
          "Stewart Butterfield (CEO of Slack)", "Jack Ma (CEO of Alibaba)", "Rosalind
          ↪   Brewer (CEO of Walgreens)",
          "Jeff Weiner (CEO of LinkedIn)")

techs <- c("Bill Gates (Co-founder of Microsoft)", "Mark Zuckerberg (Co-founder of
↪   Facebook)",
           "Tim Cook (CEO of Apple)", "Elon Musk (CEO of Tesla)", "Larry Page (Co-founder
           ↪   of Google)",
           "Sergey Brin (Co-founder of Google)", "Jeff Bezos (Founder of Amazon)",
           "Stewart Butterfield (CEO of Slack)", "Jack Ma (CEO of Alibaba)",
           "Jeff Weiner (CEO of LinkedIn)")

founders <- c("Bill Gates (Co-founder of Microsoft)", "Mark Zuckerberg (Co-founder of
↪   Facebook)",
              "Larry Page (Co-founder of Google)", "Sergey Brin (Co-founder of Google)",
              "Sean Combs (Founder of Bad Boy Entertainment)", "Daymond John (Founder of
              ↪   FUBU)",
              "Jeff Bezos (Founder of Amazon)", "Michael Bloomberg (Co-founder of
              ↪   Bloomberg LP)",
              "Phil Knight (Co-founder of Nike)")

if(USE_API) {
  d0 <- qual_data |>
    mutate(ec_2 = tolower(ec_2)) |>
```

```
filter(workerId!="", selection_6!="", ec_2 %in% c("one one", "\"one one\""),
↪   Finished==1)  |>
mutate(fem_choice = across(c(bonus_ctrl2_7, bonus_ctrl1_7, bonus_trt_7),
              ~ case_when(. %in% females ~ 1,
                          TRUE ~ 0)),
       ceo_choice = across(c(bonus_ctrl2_7, bonus_ctrl1_7, bonus_trt_7),
              ~ case_when(. %in% ceos ~ 1,
                          TRUE ~ 0)),
        tech_choice = across(c(bonus_ctrl2_7, bonus_ctrl1_7, bonus_trt_7),
              ~ case_when(. %in% techs ~ 1,
                TRUE ~ 0)),
        founder_choice = across(c(bonus_ctrl2_7, bonus_ctrl1_7, bonus_trt_7),
              ~ case_when(. %in% founders ~ 1,
                TRUE ~ 0)),
       condition = case_when(group %in% c("control1", "control2") ~ "Control",
                             TRUE ~ "Treatment")) |>
mutate(fem_score = case_when(`female_count_type-1` == "are female" ~ female_count_1,
                          `female_count_type-2` == "are female" ~ female_count_2,
                          `female_count_type-3` == "are female" ~ female_count_3),
       gender_feedback = case_when(condition=="Control" ~ 0, condition=="Treatment" ~
       ↪   1),
       founder = case_when((group == "control1" & (`count_type-1` == "are founders" |
       ↪   `count_type-2` == "are founders")) | (group=="control2") |
       ↪   (group=="treatment" & (`female_count_type-1` == "are founders" |
       ↪   `female_count_type-2` == "are founders" | `female_count_type-3` == "are
       ↪   founders")) ~ 1, TRUE ~ 0),
       tech = case_when((group == "control1" & (`count_type-1` == "are in the
       ↪   technology industry" | `count_type-2` == "are in the technology
       ↪   industry")) | (group=="control2") | (group=="treatment" &
       ↪   (`female_count_type-1` == "are in the technology industry" |
       ↪   `female_count_type-2` == "are in the technology industry" |
       ↪   `female_count_type-3` == "are in the technology industry")) ~ 1, TRUE ~
       ↪   0),
       ceo = case_when((group == "control1" & (`count_type-1` == "are CEOs" |
       ↪   `count_type-2` == "are CEOs")) | (group=="control2") | (group=="treatment"
       ↪   & (`female_count_type-1` == "are CEOs" | `female_count_type-2` == "are
       ↪   CEOs" | `female_count_type-3` == "are CEOs")) ~ 1, TRUE ~ 0),
       list_two = case_when(group=="control1" ~ 1, group=="control2" ~ 0, TRUE ~
       ↪   NA_real_)) |>
mutate(female = case_when(fem_choice$bonus_ctrl2_7==1 | fem_choice$bonus_ctrl1_7==1 |
↪   fem_choice$bonus_trt_7==1 ~ 1, TRUE ~ 0),
       ceo_pick = case_when(ceo_choice$bonus_ctrl2_7==1 | ceo_choice$bonus_ctrl1_7==1
       ↪   |  ceo_choice$bonus_trt_7==1 ~ 1, TRUE ~ 0),
       founder_pick = case_when(founder_choice$bonus_ctrl2_7==1 |
       ↪   founder_choice$bonus_ctrl1_7==1 |  founder_choice$bonus_trt_7==1 ~ 1, TRUE
       ↪   ~ 0),
       tech_pick = case_when(tech_choice$bonus_ctrl2_7==1 |
       ↪   tech_choice$bonus_ctrl1_7==1 |  tech_choice$bonus_trt_7==1 ~ 1, TRUE ~ 0),
       gender_code = case_when(gender=="Man" ~ 1, TRUE ~ 0),
       race_code = case_when(race=="White / Caucasian" ~ 1, TRUE ~ 0),
       age = as.numeric(age),
       gender = case_when(gender=="" ~ "N/A",
                          TRUE ~ gender),
           race = case_when(race=="" ~ "N/A",
                            TRUE ~ race),
           encouragement = case_when(vignette=="demand" ~ 1,
                                     vignette=="base" ~ 0,
                                  TRUE ~ NA),
           base_gender = rowSums(across(selection_1:selection_6, ~ . %in% females))) |>
    select(
      list_two, encouragement, gender_feedback, female, ceo, ceo_pick, founder,
↪   founder_pick, tech, tech_pick, base_gender, gender, race, age, gender_code,
↪   race_code) |>
    slice(1:1000) # pre-registered sample size

  # Write the API-pulled data into a CSV file
  write.csv(d0, 'StudyS2A.csv', row.names = FALSE, quote = TRUE)
}

# Create the encouragement-specific dataframes
d0_w_encouragement <- d0 |>
  filter(encouragement == 1)

d0_no_encouragement <- d0 |>
  filter(encouragement == 0)
```

Variable Names

Variable	**Description**
`list_two`	Binary indicator of whether the control received a list of two attributes (list_two=1) or not (list_two=0).
`encouragement`	Binary indicator of whether a participant was randomly assigned to the encouragement condition.
`gender_feedback`	Binary indicator of whether a participant was randomly assigned to gender feedback condition.
`female`	Binary indicator of whether a participant selected a female business leader for their seventh selection.
`ceo`	Binary indicator of whether a participant was randomly assigned to receive CEO feedback.
`ceo_pick`	Binary indicator of whether a participant selected a CEO business leader for their seventh selection.
`founder`	Binary indicator of whether a participant was randomly assigned to receive founder feedback.
`founder_pick`	Binary indicator of whether a participant selected a founder business leader for their seventh selection.
`tech`	Binary indicator of whether a participant was randomly assigned to receive technologist feedback.
`tech_pick`	Binary indicator of whether a participant selected a technologist business leader for their seventh selection.
`base_gender`	Count of the number of female panelists selected in the initial six selections.
`gender`	Self-selected gender.
`race`	Self-selected race.
`age`	Self-entered age.
`gender_code`	Dummy code for gender (male = 1).
`race_code`	Dummy code for race (white = 1).

Demographics

```
## Excluded Participants: 265
```

```
##                        Percentage gender
## 1 Another gender not listed here:    0.2
## 2                             Man   47.2
## 3                      Non-binary    1.1
## 4                           Woman   51.5
```

```
##                          Percentage Race
## 1 American Indian or Alaskan Native  0.5
## 2          Asian / Pacific Islander  8.5
## 3         Black or African American 11.2
## 4                 Hispanic / Latinx  5.7
## 5                 White / Caucasian 74.1
```

```
## # A tibble: 1 x 2
##   mean_age sd_age
##      <dbl>  <dbl>
## 1     40.4   12.0
```

Pooled Analysis

```
## no gender feedback
r0_0 <- lm(female ~ list_two, data=d0_no_encouragement)

# Calculate robust standard errors
robust_summary(r0_0)
```

```
##
## Call:
## lm(formula = female ~ list_two, data = d0_no_encouragement)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.2845 -0.2845 -0.2823  0.7155  0.7177
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.282258   0.040749   6.927 3.99e-11 ***
## list_two    0.002225   0.058701   0.038     0.97
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4525 on 238 degrees of freedom
##   (256 observations deleted due to missingness)
## Multiple R-squared:  6.087e-06,  Adjusted R-squared:  -0.004196
## F-statistic: 0.001449 on 1 and 238 DF,  p-value: 0.9697
```

```
## no diversity encouragement
r0_1 <- lm(female ~ list_two, data=d0_w_encouragement)

# Calculate robust standard errors
robust_summary(r0_1)
```

```
##
## Call:
## lm(formula = female ~ list_two, data = d0_w_encouragement)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3621 -0.3621 -0.2707  0.6379  0.7293
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.36207    0.04501   8.044 3.63e-14 ***
## list_two    -0.09139    0.05944  -1.538    0.125
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4634 on 247 degrees of freedom
##   (255 observations deleted due to missingness)
## Multiple R-squared:  0.009661,   Adjusted R-squared:  0.005652
## F-statistic:  2.41 on 1 and 247 DF,  p-value: 0.1219
```

Primary Analysis

No Encouragement

```
# primary model, no encouragement
r1 <- lm(female ~ gender_feedback*encouragement, data=d0)

robust_summary(r1)
```

```
## 
## Call:
## lm(formula = female ~ gender_feedback * encouragement, data = d0)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -0.5020 -0.4766 -0.2833  0.5234  0.7167 
## 
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                    0.283333   0.029209   9.700  < 2e-16 ***
## gender_feedback                0.193229   0.042840   4.511 7.24e-06 ***
## encouragement                  0.029920   0.041522   0.721    0.471    
## gender_feedback:encouragement -0.004521   0.060781  -0.074    0.941    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.4804 on 996 degrees of freedom
## Multiple R-squared:  0.03878,    Adjusted R-squared:  0.03589 
## F-statistic:  13.4 on 3 and 996 DF,  p-value: 1.417e-08
```

```
robust_confint(r1)
```

```
##                                    2.5 %    97.5 %
## (Intercept)                    0.2260153 0.3406514
## gender_feedback                0.1091629 0.2772955
## encouragement                 -0.0515615 0.1114009
## gender_feedback:encouragement -0.1237942 0.1147514
```

Secondary Analysis

```
## ceo feedback
r_ceo <- lm(ceo_pick ~ ceo*encouragement, data=d0)

# Display the robust_summary with robust standard errors
robust_summary(r_ceo)
```

```
## 
## Call:
```

```
## lm(formula = ceo_pick ~ ceo * encouragement, data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5134 -0.4744 -0.4394  0.5255  0.5606
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.47445    0.04298  11.040   <2e-16 ***
## ceo               -0.03156    0.05038  -0.626    0.531
## encouragement     -0.03506    0.06117  -0.573    0.567
## ceo:encouragement  0.10560    0.07147   1.478    0.140
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4992 on 996 degrees of freedom
## Multiple R-squared:  0.004345,   Adjusted R-squared:  0.001346
## F-statistic: 1.449 on 3 and 996 DF,  p-value: 0.2271
```

```r
robust_confint(r_ceo)
```

```
##                         2.5 %     97.5 %
## (Intercept)        0.39011910 0.55878601
## ceo               -0.13041707 0.06730583
## encouragement     -0.15509324 0.08497601
## ceo:encouragement -0.03464641 0.24585148
```

```r
## founder feedback
r_founder <- lm(founder_pick ~ founder*encouragement, data=d0)

# Display the robust_summary with robust standard errors
robust_summary(r_founder)
```

```
##
## Call:
## lm(formula = founder_pick ~ founder * encouragement, data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3692 -0.3351 -0.3019  0.6308  0.7266
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.301887   0.045014   6.706 3.34e-11 ***
## founder                0.067344   0.051250   1.314    0.189
## encouragement         -0.028449   0.060024  -0.474    0.636
## founder:encouragement -0.005675   0.069274  -0.082    0.935
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4725 on 996 degrees of freedom
## Multiple R-squared:  0.004719,   Adjusted R-squared:  0.001721
## F-statistic: 1.574 on 3 and 996 DF,  p-value: 0.194
```

```
robust_confint(r_founder)
```

```
##                              2.5 %     97.5 %
## (Intercept)             0.21355343 0.39022016
## founder                -0.03322561 0.16791357
## encouragement          -0.14623784 0.08933926
## founder:encouragement -0.14161516 0.13026497
```

```
## tech feedback
r_tech <- lm(tech_pick ~ tech*encouragement, data=d0)

# Display the robust_summary with robust standard errors
robust_summary(r_tech)
```

```
##
## Call:
## lm(formula = tech_pick ~ tech * encouragement, data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3488 -0.3488 -0.3191  0.6512  0.7344
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.488e-01  4.229e-02   8.249 5.03e-16 ***
## tech                -6.337e-05  4.910e-02  -0.001    0.999
## encouragement       -8.321e-02  5.776e-02  -1.441    0.150
## tech:encouragement  5.359e-02  6.738e-02   0.795    0.427
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4693 on 996 degrees of freedom
## Multiple R-squared:  0.003366,   Adjusted R-squared:  0.0003645
## F-statistic: 1.121 on 3 and 996 DF,  p-value: 0.3393
```

```
robust_confint(r_tech)
```

```
##                           2.5 %     97.5 %
## (Intercept)          0.26584884 0.43182558
## tech                -0.09641335 0.09628661
## encouragement       -0.19656290 0.03013848
## tech:encouragement -0.07863192 0.18580653
```

```
## interaction of base gender
# primary model
r_interaction <- lm(female ~ gender_feedback*base_gender, data=d0)


# Display the summary with robust standard errors
robust_summary(r_interaction)
```

```
##
## Call:
## lm(formula = female ~ gender_feedback * base_gender, data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6618 -0.3351 -0.2810  0.4634  0.7461
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  0.33511    0.03439   9.743  < 2e-16 ***
## gender_feedback              0.32671    0.04896   6.673 4.14e-11 ***
## base_gender                 -0.02707    0.01940  -1.395 0.163303
## gender_feedback:base_gender -0.09819    0.02605  -3.770 0.000173 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4703 on 996 degrees of freedom
## Multiple R-squared:  0.07907,    Adjusted R-squared:  0.0763
## F-statistic: 28.51 on 3 and 996 DF,  p-value: < 2.2e-16
```

Figure S4A Code

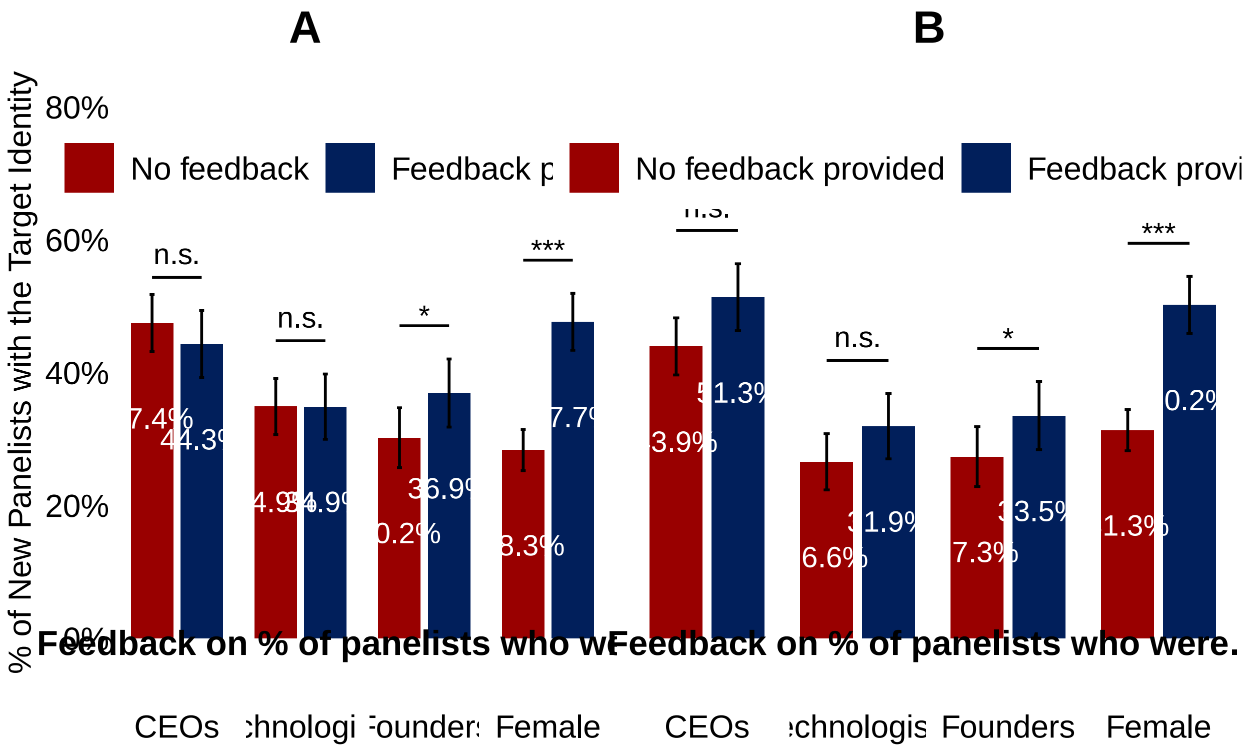
```r
dgender_plot <- d0_no_encouragement |>
  select(gender_feedback, female) |>
  group_by(gender_feedback, female) |>
  summarise(n = n()) |>
  mutate(freq = n / sum(n)) |>
  filter(female == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(gender_feedback==0 ~ coef(summary(r1))[, "Std. Error"][1]*100,
                        TRUE ~ coef(robust_summary(r1))[, "Std. Error"][2]*100)) |>
  mutate(gender_feedback = case_when(gender_feedback==1 ~ "\"Treatment\"",
                                     TRUE ~ "\"Control\"")) |>
  rename(Condition = gender_feedback)

## dataframe for CEO information

dceo_plot <- d0_no_encouragement |>
  select(ceo, ceo_pick) |>
  group_by(ceo, ceo_pick) |>
  summarise(n = n()) |>
  mutate(freq = n / sum(n)) |>
  filter(ceo_pick == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(ceo==0 ~ coef(robust_summary(r_ceo))[, "Std. Error"][1]*100,
                        TRUE ~ coef(robust_summary(r_ceo))[, "Std. Error"][2]*100)) |>
  mutate(ceo = case_when(ceo==1 ~ "\"Treatment\"",
                         TRUE ~ "\"Control\"")) |>
  rename(Condition = ceo)

## dataframe for Founder information
```

```
dfounder_plot <- d0_no_encouragement |>
  select(founder, founder_pick) |>
  group_by(founder, founder_pick) |>
  summarise(n = n()) |>
  mutate(freq = n / sum(n)) |>
  filter(founder_pick == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(founder==0 ~ coef(robust_summary(r_founder))[, "Std.
          ↪  Error"][1]*100,
                        TRUE ~ coef(robust_summary(r_founder))[, "Std. Error"][2]*100))
                         ↪  |>
  mutate(founder = case_when(founder==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = founder)

## dataframe for Tech information

dtech_plot <- d0_no_encouragement |>
  select(tech, tech_pick) |>
  group_by(tech, tech_pick) |>
  summarise(n = n()) |>
  mutate(freq = n / sum(n)) |>
  filter(tech_pick == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(tech==0 ~ coef(robust_summary(r_tech))[, "Std. Error"][1]*100,
                        TRUE ~ coef(robust_summary(r_tech))[, "Std. Error"][2]*100)) |>
  mutate(tech = case_when(tech==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = tech)

df_combined <- bind_rows(
  dceo_plot %>% mutate(Category = "\nCEOs"),
  dtech_plot %>% mutate(Category = "\nTechnologists"),
  dfounder_plot %>% mutate(Category = "\nFounders"),
  dgender_plot %>% mutate(Category = "\nFemale")
, .id = "id") %>%
  mutate(Category = factor(Category, levels = c('\nCEOs', '\nTechnologists',
  ↪  '\nFounders', '\nFemale')))

p_combined_A <- ggplot(df_combined, aes(x = Condition, y = freq*100, fill = Condition)) +
  geom_bar(stat="identity", width = 0.85, position = position_dodge(width = 0.7)) +
  geom_text(aes(label=paste0(sprintf("%.1f", freq*100),"%")),
            position=position_dodge(width=0.7), vjust=5, size = 4, color = "white") +
  geom_errorbar(aes(ymin=freq*100-se, ymax=freq*100+se), width = .1, position =
  ↪  position_dodge(width = 0.7)) +
  facet_wrap(~factor(Category, c('\nCEOs', '\nTechnologists', '\nFounders', '\nFemale')),
  ↪  nrow = 1, strip.position = "bottom") +
  geom_segment(data = df_combined %>% filter(Condition == "\"Treatment\""),
               aes(x = 1, xend = 2, y = freq*100 + se + 5, yend = freq*100 + se + 5),
               inherit.aes = FALSE) +
  geom_text(data = df_combined %>% filter(Category %in% c('\nCEOs', '\nTechnologists') &
  ↪  Condition == "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 7, yend = freq*100 + se + 7,
            ↪   label = "n.s."),
            inherit.aes = FALSE, vjust = 0) +
    geom_text(data = df_combined %>% filter(Category %in% c('\nFounders') & Condition ==
    ↪   "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 5, yend = freq*100 + se + 5,
            ↪   label = "*"),
            inherit.aes = FALSE, vjust = 0) +
  geom_text(data = df_combined %>% filter(Category == '\nFemale' & Condition ==
  ↪   "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 5, yend = freq*100 + se + 5,
            ↪   label = "***"),
            inherit.aes = FALSE, vjust = 0) +
  theme_bw() +
  scale_fill_manual(values = c("#990000", "#011F5B"), labels = c("No feedback", "Feedback
  ↪   provided"), "Feedback") +
  scale_y_continuous(labels = function(x) paste0(x,"%"), limits = c(0,80)) +
  scale_x_discrete(labels = c("\"Control\"" = "Not\nShown", "\"Treatment\"" = "Shown")) +
  labs(x = "Feedback on % of panelists who were...", y = "% of New Panelists with the
  ↪   Target Identity",
       caption = 'Note: Error Bars are SEs', title = "The Effect of Getting Feedback on
       ↪   Your Panel's Composition") +
  theme(plot.caption = element_text(face = "italic"),
        legend.position = c(0.5, 0.85),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.text = element_text(size = 12),
        legend.key.size = unit(7, 'mm'),
        legend.background = element_rect(fill = "white"),
        panel.grid.minor = element_blank(),
        panel.grid = element_blank(),
        panel.border = element_rect(fill= NA, color = "white"),
        plot.background = element_rect(fill = "white"),
        panel.background = element_rect(fill = "white"),
        axis.title.x = element_text(face="bold", size = 13, vjust = 19),
        plot.title = element_blank(),
        axis.title.y = element_text(size = 12, color = "black"),
        axis.text.x = element_blank(),
        axis.ticks = element_blank(),
        axis.text.y = element_text(size = 12, color = "black"),
        strip.text = element_text(size = 12, color = "black"),
        strip.background = element_rect(color = "white", fill = "white"))
#p_combined_A
```

Figure S4B Code

```
dgender_plot <- d0_w_encouragement |>
  select(gender_feedback, female) |>
  group_by(gender_feedback, female) |>
  summarise(n = n()) |>
  mutate(freq = n / sum(n)) |>
  filter(female == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(gender_feedback==0 ~ coef(summary(r1))[, "Std. Error"][1]*100,
                        TRUE ~ coef(robust_summary(r1))[, "Std. Error"][2]*100)) |>
  mutate(gender_feedback = case_when(gender_feedback==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = gender_feedback)

## dataframe for CEO information

dceo_plot <- d0_w_encouragement |>
  select(ceo, ceo_pick) |>
  group_by(ceo, ceo_pick) |>
  summarise(n = n()) |>
  mutate(freq = n / sum(n)) |>
  filter(ceo_pick == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(ceo==0 ~ coef(robust_summary(r_ceo))[, "Std. Error"][1]*100,
                        TRUE ~ coef(robust_summary(r_ceo))[, "Std. Error"][2]*100)) |>
  mutate(ceo = case_when(ceo==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = ceo)

## dataframe for Founder information

dfounder_plot <- d0_w_encouragement |>
  select(founder, founder_pick) |>
  group_by(founder, founder_pick) |>
  summarise(n = n()) |>
  mutate(freq = n / sum(n)) |>
  filter(founder_pick == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(founder==0 ~ coef(robust_summary(r_founder))[, "Std.
         ↪  Error"][1]*100,
                        TRUE ~ coef(robust_summary(r_founder))[, "Std. Error"][2]*100))
                        ↪  |>
  mutate(founder = case_when(founder==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = founder)

## dataframe for Tech information

dtech_plot <- d0_w_encouragement |>
  select(tech, tech_pick) |>
  group_by(tech, tech_pick) |>
  summarise(n = n()) |>
```

```r
  mutate(freq = n / sum(n)) |>
  filter(tech_pick == 1) |>
  mutate(sd = sqrt((freq*(1-freq))/n)*100,
         se = case_when(tech==0 ~ coef(robust_summary(r_tech))[, "Std. Error"][1]*100,
                        TRUE ~ coef(robust_summary(r_tech))[, "Std. Error"][2]*100)) |>
  mutate(tech = case_when(tech==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = tech)

df_combined <- bind_rows(
  dceo_plot %>% mutate(Category = "\nCEOs"),
  dtech_plot %>% mutate(Category = "\nTechnologists"),
  dfounder_plot %>% mutate(Category = "\nFounders"),
  dgender_plot %>% mutate(Category = "\nFemale")
, .id = "id") %>%
  mutate(Category = factor(Category, levels = c('\nCEOs', '\nTechnologists',
  ↪  '\nFounders', '\nFemale')))

p_combined_B <- ggplot(df_combined, aes(x = Condition, y = freq*100, fill = Condition)) +
  geom_bar(stat="identity", width = 0.85, position = position_dodge(width = 0.7)) +
  geom_text(aes(label=paste0(sprintf("%.1f", freq*100),"%")),
            position=position_dodge(width=0.7), vjust=5, size = 4, color = "white") +
  geom_errorbar(aes(ymin=freq*100-se, ymax=freq*100+se), width = .1, position =
  ↪  position_dodge(width = 0.7)) +
  facet_wrap(~factor(Category, c('\nCEOs', '\nTechnologists', '\nFounders', '\nFemale')),
  ↪  nrow = 1, strip.position = "bottom") +
  geom_segment(data = df_combined %>% filter(Condition == "\"Treatment\""),
               aes(x = 1, xend = 2, y = freq*100 + se + 5, yend = freq*100 + se + 5),
               inherit.aes = FALSE) +
  geom_text(data = df_combined %>% filter(Category %in% c('\nCEOs', '\nTechnologists') &
  ↪  Condition == "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 7, yend = freq*100 + se + 7,
            ↪  label = "n.s."),
            inherit.aes = FALSE, vjust = 0) +
    geom_text(data = df_combined %>% filter(Category %in% c('\nFounders') & Condition ==
    ↪  "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 5, yend = freq*100 + se + 5,
            ↪  label = "*"),
            inherit.aes = FALSE, vjust = 0) +
  geom_text(data = df_combined %>% filter(Category == '\nFemale' & Condition ==
  ↪  "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 5, yend = freq*100 + se + 5,
            ↪  label = "***"),
            inherit.aes = FALSE, vjust = 0) +
  theme_bw() +
  scale_fill_manual(values = c("#990000", "#011F5B"), labels = c("No feedback provided",
  ↪  "Feedback provided"), "Feedback") +
  scale_y_continuous(labels = function(x) paste0(x,"%"), limits = c(0,80)) +
  scale_x_discrete(labels = c("\"Control\"" = "Not\nShown", "\"Treatment\"" = "Shown")) +
  labs(x = "Feedback on % of panelists who were...", y = "% of New Panelists with the
  ↪  Target Identity",
       caption = 'Note: Error Bars are SEs', title = "The Effect of Getting Feedback on
       ↪  Your Panel's Composition") +
  theme(plot.caption = element_text(face = "italic"),
        legend.position = c(0.5, 0.85),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.text = element_text(size = 12),
        legend.key.size = unit(7, 'mm'),
        legend.background = element_rect(fill = "white"),
       panel.grid.minor = element_blank(),
       panel.grid = element_blank(),
       panel.border = element_rect(fill= NA, color = "white"),
       plot.background = element_rect(fill = "white"),
       panel.background = element_rect(fill = "white"),
       axis.title.x = element_text(face="bold", size = 13, vjust = 19),
       plot.title = element_blank(),
       axis.title.y = element_blank(),  # Remove y-axis title
       axis.text.x = element_blank(),
       axis.ticks = element_blank(),
       axis.text.y = element_blank(),  # Remove y-axis text
       axis.ticks.y = element_blank(),  # Remove y-axis ticks
       strip.text = element_text(size = 12, color = "black"),
       strip.background = element_rect(color = "white", fill = "white"))

#p_combined_B
# Create separate plots for the labels
label_A <- ggplot() +
  annotate("text", x = 0, y = 0, label = "A", size = 6, fontface = "bold") +
  theme_void()

label_B <- ggplot() +
  annotate("text", x = 0, y = 0, label = "B", size = 6, fontface = "bold") +
  theme_void()

# Combine the main plots and labels using grid.arrange()
combined_plot <- grid.arrange(
  arrangeGrob(label_A, p_combined_A, ncol = 1, heights = c(0.1, 1)),
  arrangeGrob(label_B, p_combined_B, ncol = 1, heights = c(0.1, 1)),
  ncol = 2
)
```



Note: Error Bars are SEs

Note: Error Bars are SEs

```
# ggsave("../Supplemental_Figures/Figure-S4.pdf", combined_plot, width = 16, height = 7,
↪  units = "in", device = cairo_pdf, family = "Times New Roman")
```

System of Equations

```
##
## Call:
## lm(formula = as.numeric(founder_pick) ~ founder + tech + ceo +
##     gender_feedback - 1, data = d0_no_encouragement)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4765 -0.3464 -0.2835  0.5235  0.7273
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## founder          0.18701    0.03808   4.911 1.24e-06 ***
## tech             0.11329    0.03728   3.039   0.0025 **
## ceo              0.17620    0.03633   4.849 1.66e-06 ***
## gender_feedback -0.01683    0.03846  -0.438   0.6619
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4777 on 492 degrees of freedom
## Multiple R-squared:  0.3622, Adjusted R-squared:  0.357
## F-statistic: 69.85 on 4 and 492 DF,  p-value: < 2.2e-16
```

```
##                                    Wald.Coefficient      P_Value
## Gender Feedback - Founder Feedback         7.099297 0.0078378268
## Gender Feedback - Tech Feedback           10.457233 0.0012624375
## Gender Feedback - CEO Feedback            12.381159 0.0004535653
```