Study 2B - New Attributes

October 08, 2025

Items

Read Data

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

if(USE_API) {
  ## Pull directly from Qualtrics API
  qual_data <- fetch_survey(surveyID='SV_cGCFBxyTGbzsZvw',  # New Study ID
                            start_date = "2025-10-08",
                     force_request = T)
} else {
  # Read the processed data directly from CSV
  d0 <- read.csv('Study2B.csv', check.names = F)
  num_excluded <- unique(d0$num_excluded_total)
}

# Define the categories based on the JavaScript feedback code
minority <- c(
  'Alice Walker', 'Gabriel Garcia Marquez', 'Isabel Allende',
  'Sandra Cisneros', 'Toni Morrison', 'WEB Du Bois'
)

multi_genre <- c(
  'Agatha Christie', 'Emily Bronte', 'Gabriel Garcia Marquez', 'George Orwell',
  'Herman Melville', 'Isabel Allende', 'Joyce Carol Oates', 'J.R.R. Tolkien',
  'Louisa May Alcott', 'Lucy Maud', 'Ray Bradbury', 'Sandra Cisneros',
  'Toni Morrison', 'WEB Du Bois'
)

sold_30m <- c(
  'Charles Dickens', 'Gabriel Garcia Marquez', 'George Orwell', 'Isabel Allende',
  'J.D. Salinger', 'J.R.R. Tolkien', 'Lucy Maud', 'F. Scott Fitzgerald'
)

classic_50plus <- c(
  'Agatha Christie', 'Charles Dickens', 'Emily Bronte', 'Ernest Hemingway',
  'F. Scott Fitzgerald', 'Gabriel Garcia Marquez', 'George Orwell', 'Herman Melville',
  'J.D. Salinger', 'J.R.R. Tolkien', 'Jack London', 'Jane Austen', 'John Steinbeck',
  'Louisa May Alcott', 'Lucy Maud', 'Nathaniel Hawthorne', 'Ray Bradbury',
  'Sylvia Plath', 'WEB Du Bois'
)

if(USE_API) {
  # Process the API data with new category definitions
  d0 <- qual_data |>
    filter(!is.na(`choice-7`), !is.na(PROLIFIC_PID)) |>
    mutate(
      # Minority feedback detection
      minority_feedback = as.numeric(grepl("were racial minorities", feedbackItem1) |
                  grepl("were racial minorities", feedbackItem2) |
                  grepl("were racial minorities", feedbackItem3)),

      # Attribute feedback detection based on JavaScript code
      forms_shown = as.numeric(grepl("spanning multiple genres", feedbackItem1) |
```

```r
                grepl("spanning multiple genres", feedbackItem2) |
                grepl("spanning multiple genres", feedbackItem3)),

    sold30m_shown = as.numeric(grepl("30M\\+ copies sold", feedbackItem1) |
                    grepl("30M\\+ copies sold", feedbackItem2) |
                    grepl("30M\\+ copies sold", feedbackItem3)),

    classic_shown = as.numeric(grepl("remained in continuous print for over 50 years",
↪    feedbackItem1) |
                    grepl("remained in continuous print for over 50 years",
                    ↪    feedbackItem2) |
                    grepl("remained in continuous print for over 50 years",
                    ↪    feedbackItem3)),

    # Picks based on categories
    minority_pick = case_when(`choice-7` %in% minority ~ 1, TRUE ~ 0),
    forms_pick = case_when(`choice-7` %in% multi_genre ~ 1, TRUE ~ 0),
    sold30m_pick = case_when(`choice-7` %in% sold_30m ~ 1, TRUE ~ 0),
    classic_pick = case_when(`choice-7` %in% classic_50plus ~ 1, TRUE ~ 0),

    # Demographics
    gender_code = case_when(gender=="Man" ~ 1, TRUE ~ 0),
    race_code = case_when(str_detect(race, "White / Caucasian") ~ 1, TRUE ~ 0),

    # Process political ideology (1-7 scale)
    poli_numeric = case_when(
      poli == "1Extremely liberal" ~ 1,
      poli == "2Liberal" ~ 2,
      poli == "3Slightly liberal" ~ 3,
      poli == "4Moderate; middle of the road" ~ 4,
      poli == "5Slightly conservative" ~ 5,
      poli == "6Conservative" ~ 6,
      poli == "7Extremely conservative" ~ 7,
      TRUE ~ NA_integer_
    ),

    # Center political ideology for interaction
    poli_centered = poli_numeric - 4,  # Center at moderate (4)

    # Process political party affiliation
    party_democrat = case_when(poli_party == "Democrat" ~ 1, TRUE ~ 0),
    party_independent = case_when(poli_party == "Independent" ~ 1, TRUE ~ 0),
    party_republican = case_when(poli_party == "Republican" ~ 1, TRUE ~ 0),

    base_minority = rowSums(across(`choice-1`:`choice-6`, ~ . %in% minority))
  ) |>
  mutate(
    across(c(I1:E3),
            ~ case_when(
              . == "Strongly disagree" ~ 1, . == "Disagree" ~ 2, . == "Somewhat
↪    disagree" ~ 3,
              . == "Neither agree nor disagree" ~ 4, . == "Somewhat agree" ~ 5, . ==
↪    "Agree" ~ 6,
              . == "Strongly agree" ~ 7, TRUE ~ NA_integer_))) |>
```

```r
    mutate(
      internal1Z = (I1 - mean(I1, na.rm = TRUE)) / sd(I1, na.rm = TRUE),
      internal2Z = (I2 - mean(I2, na.rm = TRUE)) / sd(I2, na.rm = TRUE),
      internal3Z = (I3 - mean(I3, na.rm = TRUE)) / sd(I3, na.rm = TRUE),
      internal4Z = (I4 - mean(I4, na.rm = TRUE)) / sd(I4, na.rm = TRUE),
      internal = (internal1Z + internal2Z + internal3Z + internal4Z) / 4,
      external1Z = (E1 - mean(E1, na.rm = TRUE)) / sd(E1, na.rm = TRUE),
      external2Z = (E2 - mean(E2, na.rm = TRUE)) / sd(E2, na.rm = TRUE),
      external3Z = (E3 - mean(E3, na.rm = TRUE)) / sd(E3, na.rm = TRUE),
      external = (external1Z + external2Z + external3Z) / 3
    ) |>
    dplyr::select(minority_feedback, forms_shown, sold30m_shown, classic_shown,
                  minority_pick, forms_pick, sold30m_pick, classic_pick, base_minority,
↪  `choice-1`:`choice-7`,
                  race, gender, age, gender_code, race_code, poli, poli_numeric,
↪  poli_centered,
                  poli_party, party_democrat, party_independent, party_republican,
                  internal1Z:external) |>
    slice(1:1000) # pre-registered sample size

  # Calculate the number of excluded participants
  num_excluded <- nrow(qual_data) - nrow(d0)

  # Save num_excluded in d0
  d0$num_excluded_total <- num_excluded  # As a column

  # Write the API-pulled data into a CSV file
  write.csv(d0, 'Study2B.csv', row.names = FALSE, quote = TRUE)
}
```

Variable Names

Variable	Description
`minority_feedback`	Binary indicator of whether a participant was randomly assigned to racial minority feedback condition.
`minority_pick`	Binary indicator of whether a participant selected a racial minority author for their seventh selection
`forms_pick`	Binary indicator of whether a participant selected an author who wrote in multiple genres for their seventh selection.
`sold30m_pick`	Binary indicator of whether a participant selected an author with at least one book with 30M+ copies sold for their seventh selection.
`classic_pick`	Binary indicator of whether a participant selected an author with a book in print for 50+ years for their seventh selection.
`base_minority`	Count of the number of racial minority authors selected in the initial six authors.
`choice-1 to choice-7`	The selected authors
`gender`	Self-selected gender.
`race`	Self-selected race.
`age`	Self-entered age.
`poli`	Political ideology on 7-point scale (1 = extremely liberal, 7 = extremely conservative).
`poli_centered`	Centered political ideology (0 = moderate).
`poli_party`	Political party affiliation (Democrat, Republican, Independent).
`party_democrat`	Indicator for Democrat party affiliation.
`party_independent`	Indicator for Independent party affiliation.
`party_republican`	Indicator for Republican party affiliation (reference category).
`gender_code`	Dummy code for gender (male = 1).
`race_code`	Dummy code for race (white = 1).
`internal1Z-4Z`	Individual standardized scale items for Internal Motivation to Respond Without Prejudice.
`external1Z-3Z`	Individual standardized scale items for External Motivation to Respond Without Prejudice.
`internal`	Aggregated scale items for Internal Motivation to Respond Without Prejudice.
`external`	Aggregated scale items for External Motivation to Respond Without Prejudice.

Demographics

```
## Excluded Participants: 36


##                              Percentage gender
## 1                                Woman  66.67
## 2                                  Man  31.79
## 3                           Non-binary   1.03
## 4 Another gender not listed here:        0.51


##                                 Percentage  Race
## 1 American Indian or Alaskan Native  0.51
## 2           Asian / Pacific Islander  5.64
## 3          Black or African American 12.82
## 4               Hispanic / Latinx  8.72
## 5               White / Caucasian 72.31


## # A tibble: 1 x 2
##   mean_age sd_age
##      <dbl>  <dbl>
## 1     43.4   14.0


## Mean Political Ideology (1-7 scale):  3.54


## SD Political Ideology:  1.85


##      Percentage Party
## 1     Democrat 43.59
## 2   Republican 22.56
## 3 Independent 33.85


## Mean (num of initial racial minorities selected):  1.41


## SD (num of initial racial minorities selected):  1.25


## Percentage (initial racial minorities selected):  0.235


## SD (initial racial minorities selected):  0.2083333


## # A tibble: 2 x 2
##   minority_feedback  mean
##              <dbl> <dbl>
## 1                 0 0.244
## 2                 1 0.224


##
##  Welch Two Sample t-test
##
## data:  base_minority/6 by minority_feedback
## t = 0.67666, df = 192.83, p-value = 0.4994
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
```

```
## 95 percent confidence interval:
##  -0.03858260  0.07888142
## sample estimates:
## mean in group 0 mean in group 1
##       0.2441077       0.2239583
```

Cronbach's Alpha

```r
# Calculating Cronbach's Alpha for the Internal subscale
internal_items <- d0[, c("internal1Z", "internal2Z", "internal3Z", "internal4Z")]
alpha_internal <- alpha(internal_items)

cat("Cronbach's Alpha for Internal Subscale: ", alpha_internal$total$raw_alpha, "\n")
```

```
## Cronbach's Alpha for Internal Subscale:  0.9298854
```

```r
# Calculating Cronbach's Alpha for the External subscale
external_items <- d0[, c("external1Z", "external2Z", "external3Z")]
alpha_external <- alpha(external_items)
cat("Cronbach's Alpha for External Subscale: ", alpha_external$total$raw_alpha, "\n")
```

```
## Cronbach's Alpha for External Subscale:  0.9133347
```

Primary Analysis

```
# primary model
r1 <- lm(minority_pick ~ minority_feedback, data=d0)

# Display the summary with robust standard errors
robust_summary(r1)
```

```
##
## Call:
## lm(formula = minority_pick ~ minority_feedback, data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3333 -0.3333 -0.2222  0.6667  0.7778
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.22222    0.04221   5.265 3.72e-07 ***
## minority_feedback  0.11111    0.06439   1.726    0.086 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4463 on 193 degrees of freedom
## Multiple R-squared:  0.01541,    Adjusted R-squared:  0.01031
## F-statistic: 3.021 on 1 and 193 DF,  p-value: 0.0838
```

```
robust_confint(r1)
```

```
##                        2.5 %     97.5 %
## (Intercept)       0.13897055 0.3054739
## minority_feedback -0.01587809 0.2381003
```

Robustness

```
## which feedback was shown with minority, remove constant due to collinearity
r2 <- lm(minority_pick ~ minority_feedback + classic_shown + sold30m_shown + forms_shown
↪   - 1, data=d0)

# Display the summary with robust standard errors
robust_summary(r2)
```

```
##
## Call:
## lm(formula = minority_pick ~ minority_feedback + classic_shown +
##     sold30m_shown + forms_shown - 1, data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4048 -0.2692 -0.2222  0.5952  0.7778
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## minority_feedback  0.17175    0.05693   3.017   0.0029 **
## classic_shown     -0.01079    0.06822  -0.158   0.8745
## sold30m_shown      0.10826    0.07253   1.493   0.1372
## forms_shown        0.12475    0.07332   1.701   0.0905 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4464 on 191 degrees of freedom
## Multiple R-squared:  0.2952, Adjusted R-squared:  0.2804
## F-statistic:     20 on 4 and 191 DF,  p-value: 9.041e-14
```

```
robust_confint(r2)
```

```
##                          2.5 %    97.5 %
## minority_feedback  0.05946145 0.2840469
## classic_shown     -0.14534322 0.1237722
## sold30m_shown     -0.03479246 0.2513167
## forms_shown       -0.01986918 0.2693604
```

```
## robust to demographic controls

r3 <- lm(minority_pick ~ minority_feedback + gender_code + race_code + age, data=d0)

# Display the summary with robust standard errors
robust_summary(r3)
```

```
##
## Call:
## lm(formula = minority_pick ~ minority_feedback + gender_code +
##     race_code + age, data = d0)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3746 -0.3152 -0.2247  0.6384  0.8203
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.274230   0.122274   2.243   0.0261 *
## minority_feedback 0.113000   0.064886   1.742   0.0832 .
## gender_code      -0.028830   0.068070  -0.424   0.6724
## race_code         0.021626   0.075077   0.288   0.7736
## age              -0.001369   0.002517  -0.544   0.5872
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4492 on 190 degrees of freedom
## Multiple R-squared:  0.01826,    Adjusted R-squared:  -0.002407
## F-statistic: 0.8835 on 4 and 190 DF,  p-value: 0.4748
```

robust_confint(r3)

```
##                        2.5 %      97.5 %
## (Intercept)       0.033041018 0.515418242
## minority_feedback -0.014989720 0.240990324
## gender_code       -0.163099280 0.105440119
## race_code         -0.126466439 0.169718274
## age               -0.006332811 0.003595746
```

```
## logistic regression
# Fit the logistic regression model
r4 <- glm(minority_pick ~ minority_feedback, family = binomial, data=d0)
summary(r4)
```

```
##
## Call:
## glm(formula = minority_pick ~ minority_feedback, family = binomial,
##     data = d0)
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -1.2528     0.2417  -5.182 2.19e-07 ***
## minority_feedback   0.5596     0.3245   1.724   0.0846 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 230.11  on 194  degrees of freedom
## Residual deviance: 227.09  on 193  degrees of freedom
## AIC: 231.09
##
## Number of Fisher Scoring iterations: 4
```

```
# Odds ratio
tidy_r4 <- tidy(r4, exponentiate = TRUE, conf.int = T)
print(tidy_r4)
```

```
## # A tibble: 2 x 7
##   term               estimate std.error statistic   p.value conf.low conf.high
##   <chr>                 <dbl>     <dbl>     <dbl>     <dbl>    <dbl>     <dbl>
## 1 (Intercept)           0.286     0.242     -5.18 0.000000219    0.174     0.450
## 2 minority_feedback     1.75      0.325      1.72 0.0846         0.931     3.34
```

Interaction Analysis

```
## interaction of base minority
# primary model
r_interaction <- lm(minority_pick ~ minority_feedback*base_minority, data=d0)


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

```
##
## Call:
## lm(formula = minority_pick ~ minority_feedback * base_minority,
##     data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3333 -0.3333 -0.2442  0.6667  0.8504
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    0.29153    0.06719   4.339 2.32e-05 ***
## minority_feedback              0.04180    0.09731   0.430    0.668
## base_minority                 -0.04732    0.02864  -1.652    0.100
## minority_feedback:base_minority  0.04732    0.04780   0.990    0.323
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4466 on 191 degrees of freedom
## Multiple R-squared:  0.02416,    Adjusted R-squared:  0.008835
## F-statistic: 1.576 on 3 and 191 DF,  p-value: 0.1965
```

Moderation by Political Ideology

```r
# Test for moderation by political ideology
r_poli_moderation <- lm(minority_pick ~ minority_feedback * poli_centered, data=d0)

# Display the summary with robust standard errors
cat("Moderation by Political Ideology\n")
```

```
## Moderation by Political Ideology
```

```r
robust_summary(r_poli_moderation)
```

```
##
## Call:
## lm(formula = minority_pick ~ minority_feedback * poli_centered,
##     data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4379 -0.3115 -0.2044  0.5621  0.9562
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      0.20438    0.04020   5.083  8.8e-07 ***
## minority_feedback                0.10372    0.06318   1.642   0.1023
## poli_centered                   -0.05354    0.02389  -2.241   0.0262 *
## minority_feedback:poli_centered  0.01028    0.03482   0.295   0.7683
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4394 on 191 degrees of freedom
## Multiple R-squared:  0.05551,    Adjusted R-squared:  0.04068
## F-statistic: 3.742 on 3 and 191 DF,  p-value: 0.01207
```

```r
robust_confint(r_poli_moderation)
```

```
##                                       2.5 %       97.5 %
## (Intercept)                      0.12507471  0.283679500
## minority_feedback               -0.02089656  0.228339533
## poli_centered                   -0.10066439 -0.006406315
## minority_feedback:poli_centered -0.05841390  0.078965490
```

```r
# Simple slopes analysis at different levels of political ideology
# Liberal (1 SD below mean, approximately liberal)
liberal_slope <- coef(r_poli_moderation)["minority_feedback"] +
                coef(r_poli_moderation)["minority_feedback:poli_centered"] * (-2)

# Moderate (at mean, centered = 0)
moderate_slope <- coef(r_poli_moderation)["minority_feedback"]
```

```
# Conservative (1 SD above mean, approximately conservative)
conservative_slope <- coef(r_poli_moderation)["minority_feedback"] +
                      coef(r_poli_moderation)["minority_feedback:poli_centered"] * (2)

cat("\n\nSimple Slopes Analysis:\n")
```

```
##
##
## Simple Slopes Analysis:
```

```
cat("Effect of minority feedback for liberals (ideology = 2): ", round(liberal_slope *
→   100, 2), "%\n")
```

```
## Effect of minority feedback for liberals (ideology = 2):  8.32 %
```

```
cat("Effect of minority feedback for moderates (ideology = 4): ", round(moderate_slope *
→   100, 2), "%\n")
```

```
## Effect of minority feedback for moderates (ideology = 4):  10.37 %
```

```
cat("Effect of minority feedback for conservatives (ideology = 6): ",
→   round(conservative_slope * 100, 2), "%\n")
```

```
## Effect of minority feedback for conservatives (ideology = 6):  12.43 %
```

```
# Visualization of interaction
library(ggplot2)
# Create prediction data
pred_data <- expand.grid(
  minority_feedback = c(0, 1),
  poli_centered = seq(-3, 3, by = 0.5)  # From very liberal to very conservative
)
pred_data$minority_pick <- predict(r_poli_moderation, newdata = pred_data)
pred_data$poli_label <- pred_data$poli_centered + 4  # Convert back to 1-7 scale

# Plot interaction
p_interaction <- ggplot(pred_data, aes(x = poli_label, y = minority_pick * 100,
                                       color = factor(minority_feedback))) +
  geom_line(size = 1.2) +
  scale_color_manual(values = c("0" = "#990000", "1" = "#011F5B"),
                     labels = c("0" = "Control", "1" = "Minority Feedback")) +
  scale_x_continuous(breaks = 1:7,
                     labels = c("Extremely\nLiberal", "Liberal", "Slightly\nLiberal",
                                "Moderate", "Slightly\nConservative", "Conservative",
                                "Extremely\nConservative")) +
  labs(x = "Political Ideology",
       y = "% Selecting Racial Minority Author",
       title = "Moderation by Political Ideology",
       color = "Condition") +
  theme_bw() +
```

```
    theme(legend.position = "top",
          axis.text.x = element_text(angle = 45, hjust = 1))

print(p_interaction)
```

Moderation by Political Ideology
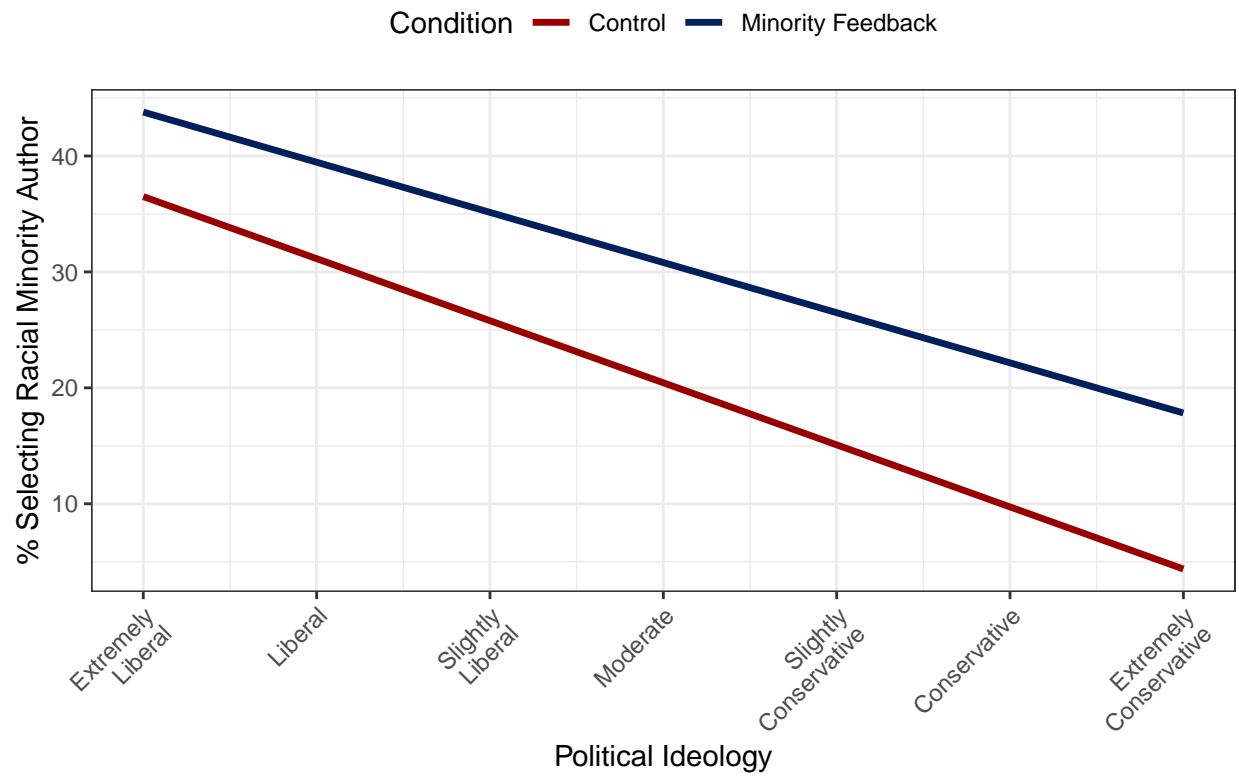
Moderation by Political Party

```r
# Test for moderation by political party affiliation
# Republican is the reference category
r_party_moderation <- lm(minority_pick ~ minority_feedback * party_democrat +
                                        minority_feedback * party_independent, data=d0)

# Display the summary with robust standard errors
cat("Moderation by Political Party Affiliation\n")
```

```
## Moderation by Political Party Affiliation
```

```r
robust_summary(r_party_moderation)
```

```
##
## Call:
## lm(formula = minority_pick ~ minority_feedback * party_democrat +
##      minority_feedback * party_independent, data = d0)
##
## Residuals:
##     Min     1Q  Median     3Q     Max
## -0.4000 -0.3095 -0.2222  0.6000  0.9524
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        0.047619   0.048795   0.976  0.33036
## minority_feedback                  0.169772   0.102302   1.660  0.09867 .
## party_democrat                     0.261905   0.087868   2.981  0.00325 **
## party_independent                  0.174603   0.086373   2.021  0.04464 *
## minority_feedback:party_democrat  -0.130459   0.146091  -0.893  0.37299
## minority_feedback:party_independent 0.008006  0.155262   0.052  0.95893
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4426 on 189 degrees of freedom
## Multiple R-squared:  0.0516, Adjusted R-squared:  0.02651
## F-statistic: 2.057 on 5 and 189 DF,  p-value: 0.07266
```

```r
robust_confint(r_party_moderation)
```

```
##                                          2.5 %     97.5 %
## (Intercept)                        -0.048633736 0.1438718
## minority_feedback                  -0.032028683 0.3715732
## party_democrat                      0.088577237 0.4352323
## party_independent                   0.004223965 0.3449824
## minority_feedback:party_democrat   -0.418637672 0.1577200
## minority_feedback:party_independent -0.298263955 0.3142750
```

```
# Simple effects analysis for each party
# Effect for Republicans (reference category)
republican_effect <- coef(r_party_moderation)["minority_feedback"]

# Effect for Democrats
democrat_effect <- coef(r_party_moderation)["minority_feedback"] +
                   coef(r_party_moderation)["minority_feedback:party_democrat"]

# Effect for Independents
independent_effect <- coef(r_party_moderation)["minority_feedback"] +
                      coef(r_party_moderation)["minority_feedback:party_independent"]

cat("\n\nSimple Effects Analysis:\n")
```

```
##
##
## Simple Effects Analysis:
```

```
cat("Effect of minority feedback for Republicans: ", round(republican_effect * 100, 2),
↪    "%\n")
```

```
## Effect of minority feedback for Republicans:  16.98 %
```

```
cat("Effect of minority feedback for Democrats: ", round(democrat_effect * 100, 2),
↪    "%\n")
```

```
## Effect of minority feedback for Democrats:  3.93 %
```

```
cat("Effect of minority feedback for Independents: ", round(independent_effect * 100, 2),
↪    "%\n")
```

```
## Effect of minority feedback for Independents:  17.78 %
```

```
# Compute means for each group and condition
party_means <- d0 |>
  group_by(poli_party, minority_feedback) |>
  summarize(
    mean_minority_pick = mean(minority_pick, na.rm = TRUE),
    n = n()
  ) |>
  filter(!is.na(poli_party))

print(party_means)
```

```
## # A tibble: 6 x 4
## # Groups:   poli_party [3]
##   poli_party  minority_feedback mean_minority_pick     n
##   <ord>                   <dbl>              <dbl> <int>
## 1 Democrat                    0              0.310    42
```

```
## 2 Democrat          1          0.349     43
## 3 Republican        0          0.0476    21
## 4 Republican        1          0.217     23
## 5 Independent       0          0.222     36
## 6 Independent       1          0.4       30
```

```r
# Visualization of interaction
p_party_interaction <- ggplot(party_means, aes(x = poli_party, y = mean_minority_pick *
→   100,
                                               fill = factor(minority_feedback))) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.8), width = 0.7) +
  scale_fill_manual(values = c("0" = "#990000", "1" = "#011F5B"),
                    labels = c("0" = "Control", "1" = "Minority Feedback")) +
  labs(x = "Political Party",
       y = "% Selecting Racial Minority Author",
       title = "Moderation by Political Party Affiliation",
       fill = "Condition") +
  theme_bw() +
  theme(legend.position = "top")

print(p_party_interaction)
```


Moderation by Political Party Affiliation
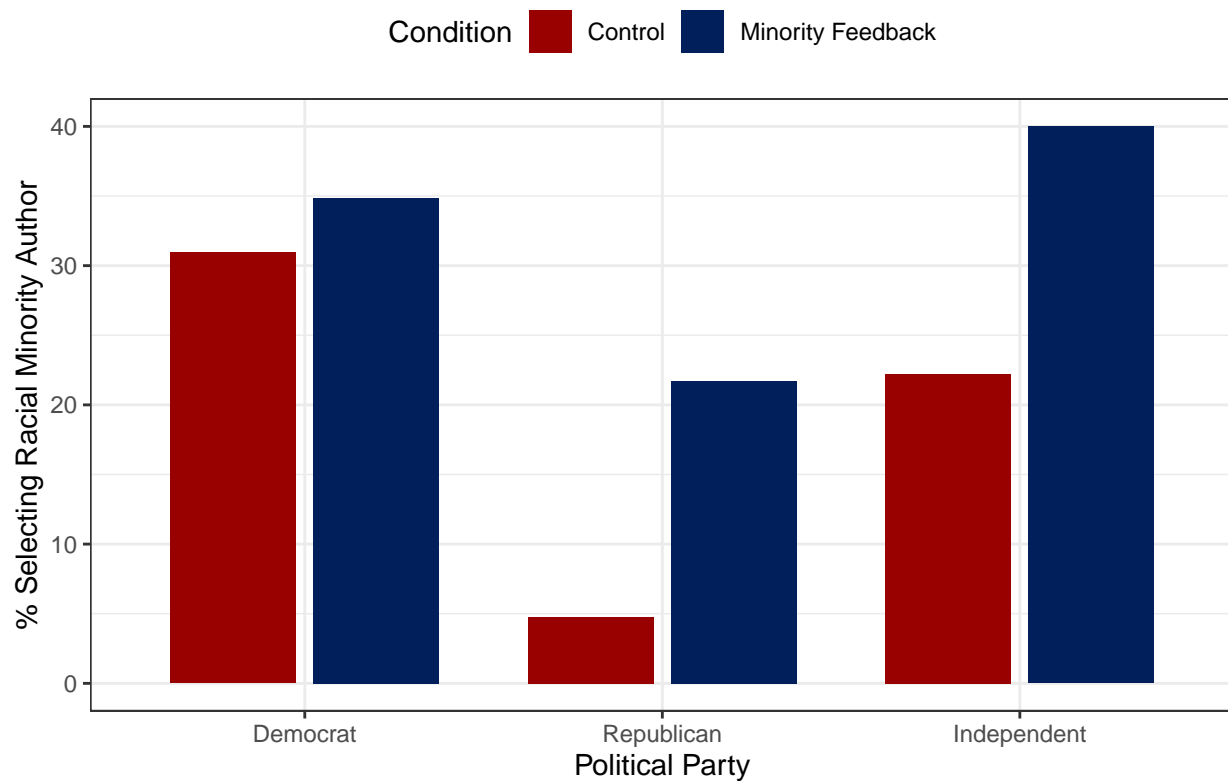
Secondary Analysis

Mediation

```
## Sobel test for Internal Motivation


## $statistic
## internal
##  5.93249
##
## $p_value
##      internal
## 2.983746e-09
##
## $se
##    internal
## 0.03508805


##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                 Estimate 95% CI Lower 95% CI Upper p-value
## ACME           0.0590377    0.0058273    0.1210501  0.0290 *
## ADE            0.0520734   -0.0637463    0.1671711  0.3886
## Total Effect   0.1111111   -0.0151099    0.2347076  0.0802 .
## Prop. Mediated 0.5313396   -1.1197025    3.2854882  0.0992 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 195
##
##
## Simulations: 10000
```

ACME(ρ)
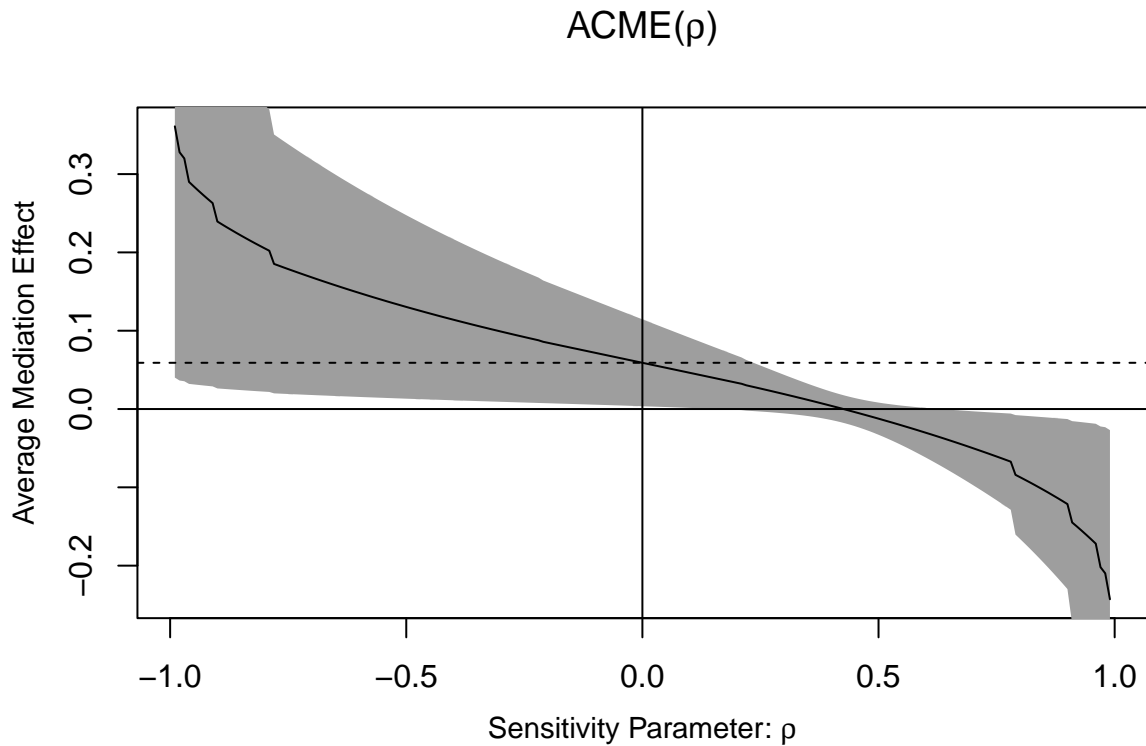

```
## Sobel test for External Motivation


## $statistic
## external
## 2.495304
##
## $p_value
##    external
## 0.01258493
##
## $se
##    external
## 0.03506313


##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                 Estimate 95% CI Lower 95% CI Upper p-value
## ACME            0.010848    -0.010581     0.045476  0.3422
## ADE             0.100264    -0.025273     0.224713  0.1186
## Total Effect    0.111111    -0.014737     0.237472  0.0826 .
## Prop. Mediated  0.097628    -0.399471     0.888477  0.3804
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 195
##
##
## Simulations: 10000
```

ACME(ρ)
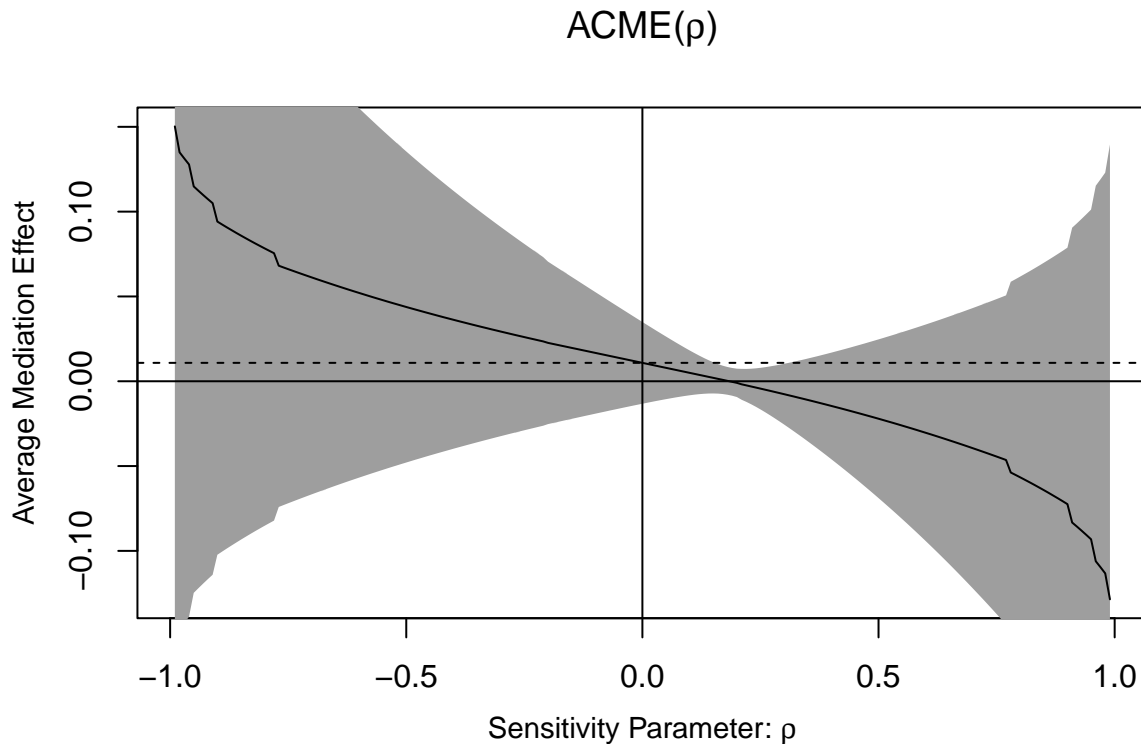

```
## Correlation Between Internal and External:  0.7779532


## P-value:  8.230305e-41


## Combined Multiple Mediation Model Results


##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                Estimate 95% CI Lower 95% CI Upper p-value
## ACME           0.099637     0.010048     0.195592  0.0268 *
## ADE            0.033687    -0.081354     0.147645  0.5848
## Total Effect   0.133324    -0.012979     0.271616  0.0740 .
## Prop. Mediated 0.747328    -0.846779     3.282953  0.0780 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Sample Size Used: 195
##
##
## Simulations: 10000


##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                 Estimate 95% CI Lower 95% CI Upper p-value
## ACME           -0.022213    -0.069096     0.026776  0.3424
## ADE             0.033687    -0.082177     0.149982  0.5904
## Total Effect    0.011474    -0.114074     0.136358  0.8678
## Prop. Mediated -1.935855    -6.293533     7.028423  0.9486
##
## Sample Size Used: 195
##
##
## Simulations: 10000
```

Figure S1 Code

```
# Get p-values from regression models
p_minority <- robust_summary(r1)$coefficients["minority_feedback", "Pr(>|t|)"]
p_sold30m <- robust_summary(r_sold30m)$coefficients["sold30m_shown", "Pr(>|t|)"]
p_forms <- robust_summary(r_forms)$coefficients["forms_shown", "Pr(>|t|)"]
p_classic <- robust_summary(r_classic)$coefficients["classic_shown", "Pr(>|t|)"]

# Function to convert p-value to significance stars
get_sig_stars <- function(p) {
  if (p < 0.001) return("***")
  else if (p < 0.01) return("**")
  else if (p < 0.05) return("*")
  else return("n.s.")
}

# Get significance labels
sig_minority <- get_sig_stars(p_minority)
sig_sold30m <- get_sig_stars(p_sold30m)
sig_forms <- get_sig_stars(p_forms)
sig_classic <- get_sig_stars(p_classic)

dminority_plot <- d0 |>
  dplyr::select(minority_feedback, minority_pick) |>
  dplyr::group_by(minority_feedback) |>
  dplyr::summarise(
    n = n(),
    freq = mean(minority_pick),
    sd = sd(minority_pick) * 100,
    se = (sd(minority_pick) / sqrt(n())) * 100
```

```r
  ) |>
  dplyr::mutate(
    minority_feedback = case_when(
      minority_feedback == 1 ~ "\"Treatment\"",
      TRUE ~ "\"Control\""
    )
  ) |>
  dplyr::rename(Condition = minority_feedback)


##### sold 30M+

dsold30m_plot <- d0 |>
  dplyr::select(sold30m_shown, sold30m_pick) |>
  dplyr::group_by(sold30m_shown) |>
  dplyr::summarise(
    n = n(),
    freq = mean(sold30m_pick),
    sd = sd(sold30m_pick) * 100,
    se = (sd(sold30m_pick) / sqrt(n())) * 100
  )|>
  mutate(sold30m_shown = case_when(sold30m_shown==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = sold30m_shown)


##### forms

dforms_plot <- d0 |>
  dplyr::select(forms_shown, forms_pick) |>
  dplyr::group_by(forms_shown) |>
  dplyr::summarise(
    n = n(),
    freq = mean(forms_pick),
    sd = sd(forms_pick) * 100,
    se = (sd(forms_pick) / sqrt(n())) * 100
  )|>
  mutate(forms_shown = case_when(forms_shown==1 ~ "\"Treatment\"",
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = forms_shown)


#### classic

dclassic_plot <- d0 |>
  dplyr::select(classic_shown, classic_pick) |>
  dplyr::group_by(classic_shown) |>
  dplyr::summarise(
    n = n(),
    freq = mean(classic_pick),
    sd = sd(classic_pick) * 100,
    se = (sd(classic_pick) / sqrt(n())) * 100
  )|>
  mutate(classic_shown = case_when(classic_shown==1 ~ "\"Treatment\"",
```

```r
                          TRUE ~ "\"Control\"")) |>
  rename(Condition = classic_shown)

## Combine plots

df_combined <- bind_rows(
  dforms_plot %>% mutate(Category = "\nWrote Multiple\nGenres", sig_label = sig_forms),
  dsold30m_plot %>% mutate(Category = "\n30M+ Copies\nSold", sig_label = sig_sold30m),
  dclassic_plot %>% mutate(Category = "\n50+ Years\nin Print", sig_label = sig_classic),
  dminority_plot %>% mutate(Category = "\nWere Racial\nMinorities", sig_label =
↪  sig_minority)
, .id = "id") %>%
  mutate(Category = factor(Category, levels = c('\nWrote Multiple\nGenres', '\n30M+
  ↪  Copies\nSold', '\n50+ Years\nin Print', '\nWere Racial\nMinorities')))

p_combined <- 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 = 5, 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('\nWrote Multiple\nGenres', '\n30M+ Copies\nSold',
  ↪  '\n50+ Years\nin Print', '\nWere Racial\nMinorities')), 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(Condition == "\"Treatment\""),
            aes(x = 1.5, y = freq*100 + se + 7, label = sig_label),
            inherit.aes = FALSE, vjust = 0, size = 5) +
  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(0,100)) +
  scale_x_discrete(labels = c("\"Control\"" = "Not\nShown", "\"Treatment\"" = "Shown")) +
  labs(x = "Feedback on % of authors who...", y = "% of New Authors with the Target
  ↪  Attribute",title = "The Effect of Getting Feedback on Your Author Selections") +
  theme(plot.caption = element_text(face = "italic"),
        legend.position = c(0.5, 0.95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.text = element_text(size = 20),
        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 = 21, vjust = 17),
        plot.title = element_blank(),
        axis.title.y = element_text(size = 20, color = "black"),
        axis.text.x = element_blank(),
        axis.ticks = element_blank(),
```
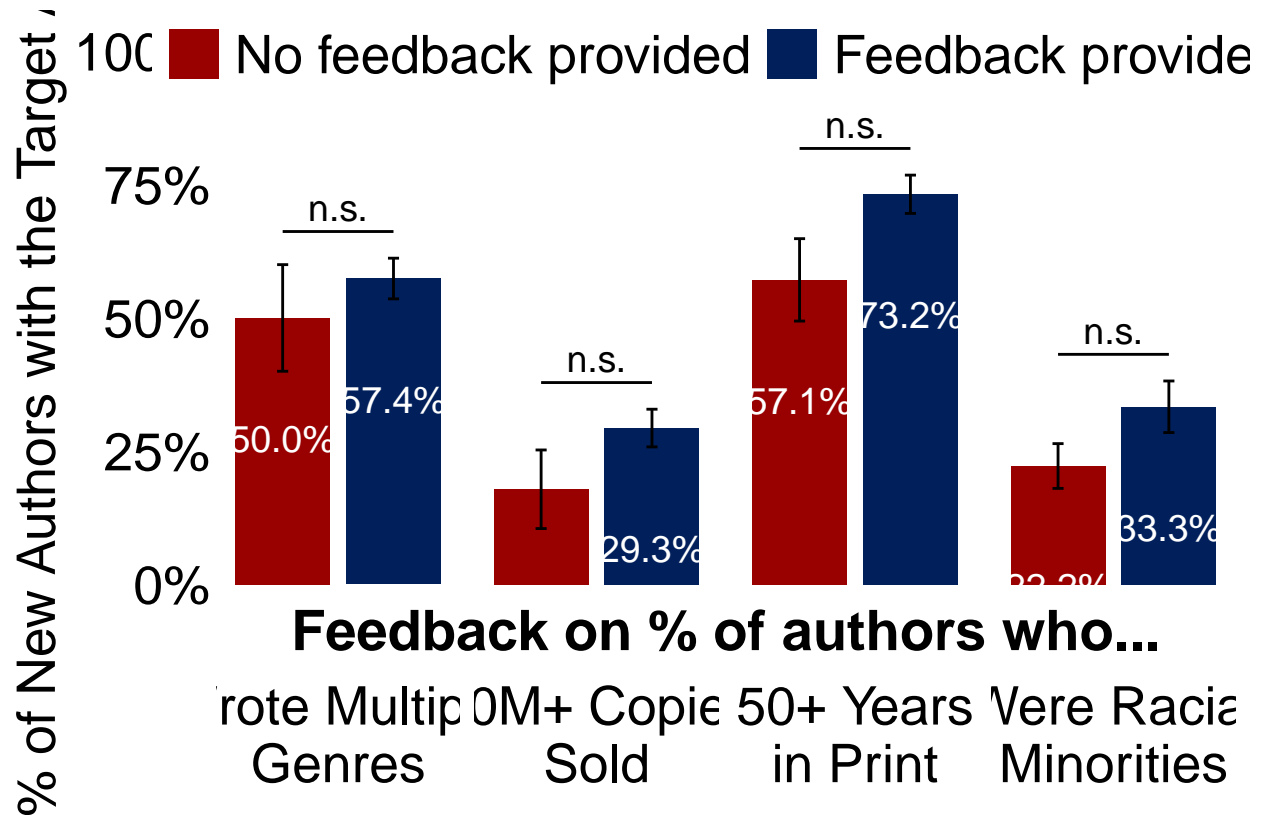
```
        axis.text.y = element_text(size = 20, color = "black"),
        strip.text = element_text(size = 20, color = "black"),
        strip.background = element_rect(colour = "white", fill = "white"))

p_combined
```



```
# Save the plot with Times New Roman font
ggsave("Figure-S1.pdf", plot = p_combined, width = 10, height = 8, units = "in", device =
↪  cairo_pdf, family = "Times New Roman")
```

System of Simultaneous Equations

```
##                                Wald.Coefficient      P_Value
## Minority Feedback - Sold30M Feedback      0.9163005 3.390536e-01
## Minority Feedback - Forms Feedback        2.2018939 1.386659e-01
## Minority Feedback - Classic Feedback     16.8003680 5.074734e-05
```