

# Study 1A

January 28, 2025

## Items

Read Data . . . . .	2
Variable Names . . . . .	4
Demographics . . . . .	5
Primary Analysis . . . . .	6
Robustness . . . . .	7
Secondary Analysis . . . . .	10
Figure 2A Code . . . . .	12
System of Equations . . . . .	16

## Read Data

```
# 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_3y2GHRkJ8RpoNka',
    label = T,
    convert = F,
    start_date = "2023-10-31",
    force_request = T)
} else {
  # Read the processed data directly from CSV
  d0 <- read.csv('Study1A.csv', check.names = F)
}

# Define movie categories
women <- c('On The Basis of Sex', 'Marie Antoinette', 'The Iron Lady', 'Judy', 'Coco
  ↳ Before Chanel', 'Rooney')
budget <- c('Oppenheimer', 'Moneyball', 'JFK', 'Braveheart', 'Lincoln', 'A Beautiful
  ↳ Mind', 'The Aviator', 'Marie Antoinette')
year <- c('A Beautiful Day in the Neighborhood', 'Oppenheimer', 'Moneyball', 'The
  ↳ Imitation Game', 'Tolkien', 'Jobs', 'J. Edgar', 'Hitchcock', 'Lincoln', 'The Darkest
  ↳ Hour', 'Judy', 'The Iron Lady', 'On The Basis of Sex')
poli <- c('JFK', 'Braveheart', 'J. Edgar', 'Nixon', 'Lincoln', 'W. A Life Misunderstood',
  ↳ 'The Darkest Hour', 'The Iron Lady', 'On The Basis of Sex', 'Marie Antoinette')

if(USE_API) {
  d0 <- qual_data |>
  filter(!is.na(`choice-7`), !is.na(workerId), Finished==1) |>
  mutate(
    gender_feedback = as.numeric(grepl("feature a female protagonist", feedbackItem1) |
      grepl("feature a female protagonist", feedbackItem2) |
      grepl("feature a female protagonist", feedbackItem3)),
    budget_shown = as.numeric(grepl("had a big budget \\\(>\\\$40 million\\)",
      ↳ feedbackItem1) |
      grepl("had a big budget \\\(>\\\$40 million\\)", feedbackItem2) |
      grepl("had a big budget \\\(>\\\$40 million\\)", feedbackItem3)),
    year_shown = as.numeric(grepl("were released after 2010", feedbackItem1) |
      grepl("were released after 2010", feedbackItem2) |
      grepl("were released after 2010", feedbackItem3)),
    poli_shown = as.numeric(grepl("feature a political leader", feedbackItem1) |
      grepl("feature a political leader", feedbackItem2) |
      grepl("feature a political leader", feedbackItem3)),
    female_pick = case_when(`choice-7` %in% women ~ 1,
      TRUE ~ 0),
    budget_pick = case_when(`choice-7` %in% budget ~ 1,
      TRUE ~ 0),
    year_pick = case_when(`choice-7` %in% year ~ 1,
      TRUE ~ 0),
    poli_pick = case_when(`choice-7` %in% poli ~ 1,
      TRUE ~ 0),
```

```

    gender_code = case_when(gender=="Man" ~ 1, TRUE ~ 0),
    race_code = case_when(race=="White / Caucasian" ~ 1, TRUE ~ 0),
    base_gender = rowSums(across(`choice-1`:`choice-6`, ~ . %in% women))) |>
select(gender_feedback:poli_pick, gender, base_gender, `choice-1`:`choice-7`,
  ↪ 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, 'Study1A.csv', row.names = FALSE, quote = TRUE)
}

```

## Variable Names

Variable	Description
gender_feedback	Binary indicator of whether a participant was randomly assigned to gender feedback condition.
female_pick	Binary indicator of whether a participant selected a biopic with a female protagonist for their seventh selection.
budget_shown	Binary indicator of whether a participant was randomly assigned to receive budget feedback.
budget_pick	Binary indicator of whether a participant selected a film with a big budget for their seventh selection.
year_shown	Binary indicator of whether a participant was randomly assigned to receive film year feedback.
year_pick	Binary indicator of whether a participant selected a film released after 2010 for their seventh selection.
poli_shown	Binary indicator of whether a participant was randomly assigned to receive political leader feedback
poli_pick	Binary indicator of whether a participant selected a film with a political protagonist for their seventh selection.
base_gender	Count of the number of films with female protagonists selected in the initial six films.
choice-1 to choice-7	The selected films
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: 127

##		Percentage	gender
## 1	Another gender not listed here:	0.1	
## 2	Man	51.0	
## 3	Non-binary	0.6	
## 4	Woman	48.3	

##		Percentage	Race
## 1	American Indian or Alaskan Native	0.5	
## 2	Asian / Pacific Islander	6.4	
## 3	Black or African American	9.7	
## 4	Hispanic / Latinx	5.5	
## 5	White / Caucasian	77.9	

## Mean (age): 43.6

## SD (age): 12.53

## Primary Analysis

```
# primary model
r1 <- lm(female_pick ~ gender_feedback, data=d0)

# robust standard errors
robust_summary(r1)

##
## Call:
## lm(formula = female_pick ~ gender_feedback, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4064 -0.4064 -0.2008  0.5936  0.7992
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.20080    0.01799  11.163 < 2e-16 ***
## gender_feedback 0.20557    0.02839   7.241 8.9e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4488 on 998 degrees of freedom
## Multiple R-squared:  0.04993,    Adjusted R-squared:  0.04898
## F-statistic: 52.45 on 1 and 998 DF,  p-value: 8.808e-13

robust_confint(r1)

##              2.5 %    97.5 %
## (Intercept)  0.1655056 0.2361008
## gender_feedback 0.1498595 0.2612831
```

## Robustness

```
## which feedback was shown with gender, remove constant due to oollinearity
r2 <- lm(female_pick ~ gender_feedback + budget_shown + year_shown + poli_shown - 1,
  ↪ data=d0)
```

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

```
##
## Call:
## lm(formula = female_pick ~ gender_feedback + budget_shown + year_shown +
##     poli_shown - 1, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4258 -0.3785 -0.2008  0.5742  0.7992
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## gender_feedback   0.26950    0.02828   9.529 < 2e-16 ***
## budget_shown      0.09177    0.03377   2.718  0.00668 **
## year_shown        0.06451    0.04349   1.483  0.13832
## poli_shown        0.04452    0.03183   1.399  0.16214
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.449 on 996 degrees of freedom
## Multiple R-squared:  0.3395, Adjusted R-squared:  0.3369
## F-statistic: 128 on 4 and 996 DF, p-value: < 2.2e-16
```

```
robust_confint(r2)
```

```
##              2.5 %    97.5 %
## gender_feedback 0.21400095 0.3250011
## budget_shown    0.02551315 0.1580331
## year_shown     -0.02083631 0.1498505
## poli_shown     -0.01793086 0.1069768
```

```
## robust to demographic controls
```

```
r3 <- lm(female_pick ~ gender_feedback + gender_code + race_code + age, data=d0)
```

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

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

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4859 -0.3765 -0.1965  0.5720  0.8593
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.124582   0.058790   2.119  0.0343 *
## gender_feedback 0.207858   0.028406   7.317 5.2e-13 ***
## gender_code    -0.014956   0.028537  -0.524  0.6003
## race_code      -0.014009   0.035194  -0.398  0.6907
## age            0.002147   0.001176   1.826  0.0681 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4486 on 995 degrees of freedom
## Multiple R-squared:  0.05361,    Adjusted R-squared:  0.0498
## F-statistic: 14.09 on 4 and 995 DF,  p-value: 3.447e-11
```

```
robust_confint(r3)
```

```
##              2.5 %      97.5 %
## (Intercept)  0.0092151226 0.239949331
## gender_feedback 0.1521162281 0.263600069
## gender_code    -0.0709568010 0.041044236
## race_code      -0.0830726707 0.055054152
## age            -0.0001600227 0.004454534
```

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

# 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)    0.251     0.112    -12.3  4.97e-35  0.201    0.311
## 2 gender_feedback 2.72      0.144     6.95  3.53e-12  2.06     3.62
```

```
summary(r4)
```

```
##
## Call:
## glm(formula = female_pick ~ gender_feedback, family = binomial,
##      data = d0)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
```



```

## (Intercept)      -1.3813      0.1119 -12.348 < 2e-16 ***
## gender_feedback  1.0023      0.1441   6.955 3.53e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1228.4  on 999  degrees of freedom
## Residual deviance: 1177.7  on 998  degrees of freedom
## AIC: 1181.7
##
## Number of Fisher Scoring iterations: 4

```

## Secondary Analysis

```
## budget feedback
r_budget <- lm(budget_pick ~ budget_shown, data=d0)

# Display the robust_summary with robust standard errors
robust_summary(r_budget)

##
## Call:
## lm(formula = budget_pick ~ budget_shown, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.3475 -0.3475 -0.3475  0.6525  0.6723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.32768    0.03548   9.236  <2e-16 ***
## budget_shown  0.01983    0.03918   0.506    0.613
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4755 on 998 degrees of freedom
## Multiple R-squared:  0.0002537, Adjusted R-squared:  -0.000748
## F-statistic: 0.2533 on 1 and 998 DF,  p-value: 0.6149

robust_confint(r_budget)
```

```
##              2.5 %      97.5 %
## (Intercept)  0.25805892 0.39730831
## budget_shown -0.05705832 0.09670931
```

```
## year feedback
r_year <- lm(year_pick ~ year_shown, data=d0)

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

```
##
## Call:
## lm(formula = year_pick ~ year_shown, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5797 -0.5768  0.4232  0.4232  0.4232
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.579710   0.060297   9.614  <2e-16 ***
```

```
## year_shown -0.002911 0.062438 -0.047 0.963
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4945 on 998 degrees of freedom
## Multiple R-squared: 2.23e-06, Adjusted R-squared: -0.0009998
## F-statistic: 0.002226 on 1 and 998 DF, p-value: 0.9624
```

```
robust_confint(r_year)
```

```
##           2.5 %    97.5 %
## (Intercept) 0.4613867 0.6980335
## year_shown -0.1254355 0.1196135
```

```
## poli feedback
r_poli <- lm(poli_pick ~ poli_shown, data=d0)
```

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

```
##
## Call:
## lm(formula = poli_pick ~ poli_shown, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4258 -0.3750 -0.3750  0.6250  0.6250
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.42578    0.03102   13.72  <2e-16 ***
## poli_shown  -0.05078    0.03575   -1.42   0.156
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4873 on 998 degrees of freedom
## Multiple R-squared: 0.002068, Adjusted R-squared: 0.001068
## F-statistic: 2.069 on 1 and 998 DF, p-value: 0.1507
```

```
robust_confint(r_poli)
```

```
##           2.5 %    97.5 %
## (Intercept) 0.3648995 0.48666296
## poli_shown -0.1209448 0.01938233
```

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

```

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

##
## Call:
## lm(formula = female_pick ~ gender_feedback * base_gender, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5000 -0.3209 -0.2009  0.5000  0.8037
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.203259   0.026338   7.717 2.88e-14 ***
## gender_feedback    0.296789   0.041415   7.166 1.50e-12 ***
## base_gender      -0.002316   0.017983  -0.129  0.8976
## gender_feedback:base_gender -0.087253   0.027237  -3.204  0.0014 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4449 on 996 degrees of freedom
## Multiple R-squared:  0.06829,    Adjusted R-squared:  0.06549
## F-statistic: 24.34 on 3 and 996 DF,  p-value: 3.355e-15

```

## Figure 2A Code

```

##### female

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

##### budget

dbudget_plot <- d0 |>
  select(budget_shown, budget_pick) |>
  group_by(budget_shown) |>
  dplyr::summarise(
    n = n(),
    freq = mean(budget_pick),

```

```

    sd = sd(budget_pick) * 100,
    se = (sd(budget_pick) / sqrt(n())) * 100
  ) |>
  mutate(budget_shown = case_when(budget_shown==1 ~ "\"Treatment\"",
                                   TRUE ~ "\"Control\"")) |>
  rename(Condition = budget_shown)

#### year

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

#### political leader

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

## Combine plots

# Load the Times New Roman font
loadfonts(device = "pdf")

# Your data-binding code
df_combined <- bind_rows(
  dbudget_plot %>% mutate(Category = "\"Above a $40M Budget\"",
  dyear_plot %>% mutate(Category = "\"Released After 2010\"",
  dpoli_plot %>% mutate(Category = "\"About a Political leader\"",
  dfemale_plot %>% mutate(Category = "\"About a Woman Protagonist\"",
  , .id = "id") %>%
  mutate(Category = factor(Category, levels = c('Above a $40M Budget', 'Released
↵ After 2010', 'About a Political leader', 'About a Woman Protagonist')))

```

```

# Create the plot
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 = 7, color = "white",
    family = "Times New Roman") +
  geom_errorbar(aes(ymin = freq * 100 - se, ymax = freq * 100 + se), width = .1, position
    = position_dodge(width = 0.7)) +
  facet_wrap(~factor(Category, c('\nAbove a\n$40M Budget', '\nReleased After\n2010',
    '\nAbout a\nPolitical leader', '\nAbout a\nWoman Protagonist')), nrow = 1,
    strip.position = "bottom") +
  geom_segment(data = df_combined %>% filter(Condition == "\"Treatment\""),
    aes(x = 1, xend = 2, y = freq * 100 + se + 7, yend = freq * 100 + se + 7),
    inherit.aes = FALSE) +
  geom_text(data = df_combined %>% filter(Category %in% c('\nAbove a\n$40M Budget',
    '\nReleased After\n2010', '\nAbout a\nPolitical leader') & Condition ==
    "\"Treatment\""),
    aes(x = 1.5, y = freq * 100 + se + 9, label = "n.s."),
    inherit.aes = FALSE, vjust = 0, size = 7, family = "Times New Roman") +
  geom_text(data = df_combined %>% filter(Category == '\nAbout a\nWoman Protagonist' &
    Condition == "\"Treatment\""),
    aes(x = 1.5, y = freq * 100 + se + 9, label = "***"),
    inherit.aes = FALSE, vjust = 0, size = 7, family = "Times New Roman") +
  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 films that were...", y = "% of Final Films Selected with the
    Target Attribute",
    title = "The Effect of Getting Feedback on Your Biopic Selections") +
  theme(plot.caption = element_text(face = "italic", family = "Times New Roman"),
    legend.position = c(0.5, 0.85),
    legend.title = element_blank(),
    legend.direction = "horizontal",
    legend.text = element_text(size = 20, family = "Times New Roman"),
    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 = 22, vjust = 17, family = "Times
    New Roman"),
    plot.title = element_blank(),
    axis.title.y = element_text(size = 20, color = "black", family = "Times New
    Roman"),
    axis.text.x = element_blank(),
    axis.ticks = element_blank(),
    axis.text.y = element_text(size = 20, color = "black", family = "Times New
    Roman"),

```

```
strip.text = element_text(size = 20, color = "black", family = "Times New  
↪ Roman"),  
strip.background = element_rect(colour = "white", fill = "white"))  
  
#p_combined_A  
# Save the plot as an R object so we can re-load it later  
saveRDS(p_combined_A, file = "p_combined_A.rds")
```

## System of Equations

```
##
## Call:
## lm(formula = as.numeric(female_pick) ~ gender_feedback + budget_shown +
##     year_shown + poli_shown - 1, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4258 -0.3785 -0.2008  0.5742  0.7992
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## gender_feedback  0.26950     0.02681  10.054 < 2e-16 ***
## budget_shown     0.09177     0.03104   2.957  0.00318 **
## year_shown       0.06451     0.03947   1.635  0.10247
## poli_shown       0.04452     0.02909   1.531  0.12618
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.449 on 996 degrees of freedom
## Multiple R-squared:  0.3395, Adjusted R-squared:  0.3369
## F-statistic: 128 on 4 and 996 DF, p-value: < 2.2e-16

##              Test Wald.Coefficient      P_Value
## 1   Female Feedback - Budget          65.06085 1.221245e-15
## 2   Female Feedback - Year           12.44103 4.295101e-04
## 3 Female Feedback - Political          57.73585 4.585221e-14
```