

# Study 2B

November 02, 2025

## Items

Read Data . . . . .	2
Variable Names . . . . .	4
Demographics . . . . .	5
Primary Analysis . . . . .	6
Robustness . . . . .	7
Secondary Analysis . . . . .	9
Figure 2B Figure Code . . . . .	12
System of Equations . . . . .	15

## Read Data

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

if(USE_API) {
  ## Pull directly from Qualtrics API
  qual_data <- fetch_survey(surveyID='SV_eFIgYjv3jONWknQ',
    label = T,
    convert = F,
    start_date = "2023-12-14",
    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
race_list <- c('Salem', 'Harriet', 'Ali', '42')
budget <- c('Oppenheimer', 'Moneyball', 'Ali', 'Braveheart', 'A Beautiful Mind', 'The
  Aviator', 'The King\'s Speech', 'Rocketman', 'The Greatest Showman', 'Walk the Line')
year <- c('A Beautiful Day in the Neighborhood', 'Oppenheimer', 'Salem', 'Moneyball',
  'The Imitation Game', 'Tolkien', 'Jobs', 'J. Edgar', 'Hitchcock', 'LBJ', 'On The Basis
  of Sex', 'The Founder', 'Chappaquiddick', 'Rocketman', 'The Greatest Showman', 'Walk
  the Line', 'Harriet')
duration <- c('Moneyball', 'Chaplin', 'W. A Life Misunderstood', 'The Aviator', 'J.
  Edgar', 'Oppenheimer', 'Selma', 'The Doors', 'A Beautiful Mind', 'Harriet', 'Nixon',
  'On The Basis of Sex', 'Jobs', 'Braveheart', 'Ali', '42', 'Patton', 'Rocketman',
  'Walk the Line')

if(USE_API) {
  # Process the API data
  d0 <- qual_data %>%
    dplyr::filter(!is.na(`choice-8`), Finished==1) %>%
    dplyr::mutate(
      race_feedback = as.numeric(grepl("featured a racial minority protagonist",
        feedbackItem1) |
        grepl("featured a racial minority protagonist", feedbackItem2) |
        grepl("featured a racial minority 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)),
      duration_shown = as.numeric(grepl("had a runtime over 2 hrs", feedbackItem1) |
        grepl("had a runtime over 2 hrs", feedbackItem2) |
        grepl("had a runtime over 2 hrs", feedbackItem3)),
      race_pick = case_when(`choice-8` %in% race_list ~ 1,
        TRUE ~ 0),
      budget_pick = case_when(`choice-8` %in% budget ~ 1,
```

```

        TRUE ~ 0),
year_pick = case_when(`choice-8` %in% year ~ 1,
                      TRUE ~ 0),
duration_pick = case_when(`choice-8` %in% duration ~ 1,
                           TRUE ~ 0),
gender_code = case_when(gender=="Man" ~ 1, TRUE ~ 0),
race_code = case_when(race=="White / Caucasian" ~ 1, TRUE ~ 0),
base_race = rowSums(across(`choice-1`:`choice-7`, ~ . %in% race_list))) %>%
dplyr::select(race_feedback:duration_pick, gender, base_race, `choice-1`:`choice-8`,
→ race, age, gender_code, race_code) %>%
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
<code>race_feedback</code>	Binary indicator of whether participant was randomly assigned to race feedback condition.
<code>race_pick</code>	Binary indicator of whether participant selected a racial minority protagonist for their seventh film selection
<code>budget_shown</code>	Binary indicator of whether participant was randomly assigned to receive budget feedback.
<code>budget_pick</code>	Binary indicator of whether participant selected a film with a big budget for their seventh selection.
<code>year_shown</code>	Binary indicator of whether participant was randomly assigned to receive film year feedback.
<code>year_pick</code>	Binary indicator of whether participant selected a film released after 2010 for their seventh selection.
<code>duration_shown</code>	Binary indicator of whether participant was randomly assigned to receive film budget feedback
<code>duration_pick</code>	Binary indicator of whether participant selected a film with a big budget for their seventh selection.
<code>base_race</code>	Count of the number of films with racial minority protagonists selected in the initial seven films.
<code>choice-1 to choice-7</code>	The selected films
<code>gender</code>	Self-selected gender.
<code>race</code>	Self-selected race.
<code>age</code>	Self-entered age.
<code>gender_code</code>	Dummy code for gender (male = 1).
<code>race_code</code>	Dummy code for race (white = 1).

## Demographics

```
## Excluded Participants: 138

##                                     Percentage gender
## 1 Another gender not listed here:      0.1
## 2                                         Man    46.1
## 3                                         Non-binary    2.4
## 4                                         Woman   51.4

##                                     Percentage Race
## 1 American Indian or Alaskan Native  0.7
## 2           Asian / Pacific Islander  8.8
## 3           Black or African American 12.0
## 4           Hispanic / Latinx       6.2
## 5           White / Caucasian     72.3

##   mean_age    sd_age
## 1  42.199 13.10791

## Mean (num of initial women selected):  1.13

## SD (num of initial women selected):  0.86

## Percentage (initial women selected):  0.1614286

## SD (initial women selected):  0.1228571

## # A tibble: 2 x 2
##   race_feedback  mean
##   <int> <dbl>
## 1 0  0.166
## 2 1  0.158

##
## Welch Two Sample t-test
##
## data: base_race/7 by race_feedback
## t = 1.0065, df = 997.76, p-value = 0.3144
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
## -0.007452731  0.023146696
## sample estimates:
## mean in group 0 mean in group 1
##          0.1659078        0.1580608
```

## Primary Analysis

```
# primary model
r1 <- lm(race_pick ~ race_feedback, data=d0)

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

## 
## Call:
## lm(formula = race_pick ~ race_feedback, data = d0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.2711 -0.2711 -0.1773 -0.1773  0.8227
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.17729   0.01708 10.38   < 2e-16 ***
## race_feedback 0.09379   0.02627  3.57 0.000373 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4147 on 998 degrees of freedom
## Multiple R-squared:  0.01265,    Adjusted R-squared:  0.01166
## F-statistic: 12.79 on 1 and 998 DF,  p-value: 0.0003654

robust_confint(r1)

##                  2.5 %    97.5 %
## (Intercept) 0.14377458 0.2108071
## race_feedback 0.04224332 0.1453437
```

## Robustness

```
## which feedback was shown with gender, remove constant due to collinearity
r2 <- lm(race_pick ~ race_feedback + budget_shown + year_shown + duration_shown - 1,
         data=d0)

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

```
##
## Call:
## lm(formula = race_pick ~ race_feedback + budget_shown + year_shown +
##     duration_shown - 1, data = d0)
##
## Residuals:
##    Min      1Q  Median      3Q     Max 
## -0.3459 -0.2312 -0.1773 -0.1773  0.8227 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## race_feedback   0.15450   0.02300  6.716 3.13e-11 ***
## budget_shown    0.10058   0.02790  3.605 0.000328 *** 
## year_shown     -0.01412   0.03020 -0.467 0.640293    
## duration_shown  0.09083   0.02835  3.204 0.001399 **  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.4135 on 996 degrees of freedom
## Multiple R-squared:  0.2397, Adjusted R-squared:  0.2366 
## F-statistic: 78.5 on 4 and 996 DF, p-value: < 2.2e-16
```

```
robust_confint(r2)
```

```
##                  2.5 %    97.5 %
## race_feedback    0.10936106 0.19964428
## budget_shown    0.04582953 0.15532974
## year_shown      -0.07338967 0.04515278
## duration_shown  0.03519875 0.14646055
```

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

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

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

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

```

## 
## Residuals:
##   Min     1Q Median     3Q    Max 
## -0.3367 -0.2555 -0.1915 -0.1248  0.9042 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 0.259405  0.051570  5.030 5.81e-07 ***
## race_feedback 0.094981  0.026362  3.603  0.00033 ***
## gender_code   -0.029398  0.026583 -1.106  0.26904    
## race_code      0.022992  0.029049  0.792  0.42883    
## age           -0.002033  0.001028 -1.977  0.04829 *  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## Residual standard error: 0.4142 on 995 degrees of freedom 
## Multiple R-squared:  0.01779, Adjusted R-squared:  0.01384 
## F-statistic: 4.505 on 4 and 995 DF, p-value: 0.001304

```

```
robust_confint(r3)
```

```

##              2.5 %      97.5 % 
## (Intercept) 0.158205407 3.606037e-01 
## race_feedback 0.043249782 1.467125e-01 
## gender_code   -0.081563722 2.276770e-02 
## race_code      -0.034011207 7.999584e-02 
## age           -0.004050017 -1.529919e-05 

## logistic regression
# Fit the logistic regression model
r4 <- glm(race_pick ~ race_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)  0.215     0.117    -13.1  2.12e-39    0.170     0.270
## 2 race_feedback 1.73      0.154     3.54  4.07e- 4    1.28      2.34

```

## 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
## -0.3699 -0.3676 -0.3676  0.6324  0.6324
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.369942  0.036919 10.020  <2e-16 ***
## budget_shown -0.002348  0.040556 -0.058    0.954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4827 on 998 degrees of freedom
## Multiple R-squared:  3.393e-06, Adjusted R-squared:  -0.0009986
## F-statistic: 0.003386 on 1 and 998 DF,  p-value: 0.9536

robust_confint(r_budget)

##                  2.5 %      97.5 %
## (Intercept)  0.29749412 0.44239027
## budget_shown -0.08193365 0.07723668

## 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
## -0.5434 -0.5434  0.4566  0.4566  0.5220
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.47799   0.03986 11.990  <2e-16 ***
```

```

## year_shown   0.06541    0.04342    1.507    0.132
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4988 on 998 degrees of freedom
## Multiple R-squared:  0.002299,  Adjusted R-squared:  0.001299
## F-statistic: 2.299 on 1 and 998 DF,  p-value: 0.1297

```

```
robust_confint(r_year)
```

```

##                   2.5 %     97.5 %
## (Intercept)  0.39975887 0.5562160
## year_shown  -0.01978351 0.1506101

```

```

## duration feedback
r_duration <- lm(duration_pick ~ duration_shown, data=d0)

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

```

```

##
## Call:
## lm(formula = duration_pick ~ duration_shown, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7711 -0.7038  0.2962  0.2962  0.2962
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.77108   0.03281 23.504   <2e-16 ***
## duration_shown -0.06725   0.03643 -1.846   0.0652 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4512 on 998 degrees of freedom
## Multiple R-squared:  0.003072,  Adjusted R-squared:  0.002073
## F-statistic: 3.076 on 1 and 998 DF,  p-value: 0.07978

```

```
robust_confint(r_duration)
```

```

##                   2.5 %     97.5 %
## (Intercept)  0.7067069 0.835461807
## duration_shown -0.1387264 0.004231546

```

```

## interaction of base gender
# primary model
r_interaction <- lm(race_pick ~ race_feedback*base_race, data=d0)

```

```

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

```

```

##
## Call:
## lm(formula = race_pick ~ race_feedback * base_race, data = d0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39468 -0.28297 -0.17126 -0.00706  0.90035
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.28481   0.03094   9.204 < 2e-16 ***
## race_feedback             0.10987   0.04586   2.396   0.0168 *
## base_race                 -0.09258   0.01679  -5.513  4.5e-08 ***
## race_feedback:base_race -0.01913   0.02618  -0.731   0.4651
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4056 on 996 degrees of freedom
## Multiple R-squared:  0.05743,    Adjusted R-squared:  0.05459
## F-statistic: 20.23 on 3 and 996 DF,  p-value: 9.916e-13

```

Figure 2B Figure Code

```
##### race

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

##### budget

d_budget_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

d_year_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)

##### duration
```

```

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

## Combine plots

df_combined <- bind_rows(
  d_budget_plot %>% mutate(Category = "\nAbove a\n$40M Budget"),
  d_year_plot %>% mutate(Category = "\nReleased After\n2010"),
  d_duration_plot %>% mutate(Category = "\nOver Two Hours"),
  d_race_plot %>% mutate(Category = "\nAbout a Racial\nMinority Protagonist")
, .id = "id") %>%
  mutate(Category = factor(Category, levels = c('\nAbove a\n$40M Budget', '\nReleased
→ After\n2010', '\nOver Two Hours', '\nAbout a Racial\nMinority Protagonist')))

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=3, 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',
  '\nOver Two Hours', '\nAbout a Racial\nMinority Protagonist')), nrow = 1,
  strip.position = "bottom") +
  geom_segment(data = df_combined %>% filter(Condition == "\"Treatment\""),
               aes(x = 1, xend = 2, y = freq*100 + se + 10, yend = freq*100 + se + 10),
               inherit.aes = FALSE) +
  geom_text(data = df_combined %>% filter(Category %in% c('
  \nAbove a\n$40M Budget',
  '\nReleased After\n2010') & Condition == "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 12, yend = freq*100 + se + 12,
                label = "n.s."),
            inherit.aes = FALSE, vjust = 0, size = 7) +
  geom_text(data = df_combined %>% filter(Category %in% c('
  \nAbout a Racial\nMinority
  Protagonist') & Condition == "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 12, yend = freq*100 + se + 12,
                label = "***"),
            inherit.aes = FALSE, vjust = 0, size = 7) +
  geom_text(data = df_combined %>% filter(Category %in% c('
  \nOver Two Hours') &
    Condition == "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 12, yend = freq*100 + se + 12,
                label = "+"),
            inherit.aes = FALSE, vjust = 0, size = 7) +
  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 = "none",
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_B



# Save the plot as an R object so we can re-load it later



#saveRDS(p_combined_B, file = "p_combined_B.rds")


```

## System of Equations

```
##  
## systemfit results  
## method: OLS  
##  
##           N   DF      SSR  detRCov   OLS-R2 McElroy-R2  
## system 2000 1992 402.162 0.039798 0.010427  0.011688  
##  
##           N   DF      SSR       MSE      RMSE       R2   Adj R2  
## budgeteq 1000 996 231.854 0.232785 0.482478 0.003106 0.000104  
## raceeq   1000 996 170.309 0.170993 0.413513 0.020222 0.017271  
##  
## The covariance matrix of the residuals  
##          budgeteq     raceeq  
## budgeteq  0.23278471 -0.00262701  
## raceeq   -0.00262701  0.17099282  
##  
## The correlations of the residuals  
##          budgeteq     raceeq  
## budgeteq  1.0000000 -0.0131673  
## raceeq   -0.0131673  1.0000000  
##  
##  
## OLS estimates for 'budgeteq' (equation 1)  
## Model Formula: as.numeric(budget_pick) ~ race_feedback + budget_shown +  
##                 duration_shown - 1  
##  
##           Estimate Std. Error t value Pr(>|t|)  
## race_feedback  0.0882644  0.0259632 3.39959 0.00070163 ***  
## budget_shown   0.1067685  0.0311134 3.43159 0.00062471 ***  
## year_shown     0.1182201  0.0317417 3.72444 0.00020674 ***  
## duration_shown 0.1634577  0.0314159 5.20303 2.3802e-07 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 0.482478 on 996 degrees of freedom  
## Number of observations: 1000 Degrees of Freedom: 996  
## SSR: 231.853569 MSE: 0.232785 Root MSE: 0.482478  
## Multiple R-Squared: 0.003106 Adjusted R-Squared: 0.000104  
##  
##  
## OLS estimates for 'raceeq' (equation 2)  
## Model Formula: as.numeric(race_pick) ~ race_feedback + budget_shown + year_shown +  
##                 duration_shown - 1  
##  
##           Estimate Std. Error t value Pr(>|t|)  
## race_feedback   0.1545027  0.0222521  6.94329 6.9009e-12 ***  
## budget_shown    0.1005796  0.0266661  3.77182 0.00017161 ***  
## year_shown     -0.0141184  0.0272046 -0.51897 0.60389485  
## duration_shown  0.0908296  0.0269253  3.37339 0.00077108 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##
```

```
## Residual standard error: 0.413513 on 996 degrees of freedom
## Number of observations: 1000 Degrees of Freedom: 996
## SSR: 170.308849 MSE: 0.170993 Root MSE: 0.413513
## Multiple R-Squared: 0.020222 Adjusted R-Squared: 0.017271

##          Test Wald.Coefficient     P_Value
## 1   Race Feedback - Budget    7.713366 0.0055326811
## 2   Race Feedback - Year    13.743929 0.0002152213
## 3 Race Feedback - Duration  6.916283 0.0086071867
```