

# Study S2B

December 31, 2025

## Items

Read Data . . . . .	2
Variable Names . . . . .	4
Demographics . . . . .	5
Primary Analysis . . . . .	6
Figure S6A Code . . . . .	10
Figure S6B Code . . . . .	13

## 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_2tycLNwc08A12cK',
    label = T,
    convert = F,
    start_date = "2024-01-22",
    force_request = T)
} else {
  # Read the processed data directly from CSV
  d0 <- read.csv('StudyS2B.csv', check.names = F)
  num_excluded <- unique(d0$num_excluded_total)
}

# Define the categories
women <- c('The Broski Report with Brittany Broski', 'Would You Believe...? with Rebecca
  ↵ Rogers', 'Wiser Than Me with Julia Louis-Dreyfus', 'Where Should We Begin? with
  ↵ Esther Perel', 'Money Rehab with Nicole Lapin')
episodes <- c('The Diary Of A CEO with Steven Bartlett', 'On Purpose with Jay Shetty',
  ↵ 'Lex Fridman Podcast', 'The School of Greatness', 'Passion Struck with John R.
  ↵ Miles', 'The Joe Budden Podcast', 'The Bill Simmons Podcast', 'This Past Weekend w/
  ↵ Theo Von', 'Money Rehab with Nicole Lapin', 'The Peter Attia Drive')
duration <- c('Huberman Lab', 'Shawn Ryan Show', 'Lex Fridman Podcast', 'The Joe Budden
  ↵ Podcast', 'Dan Carlin\\'s Hardcore History')
years <- c('What Now? with Trevor Noah', 'The Ultimate Human with Gary Brecka', 'The
  ↵ Broski Report with Brittany Broski', 'Would You Believe...? with Rebecca Rogers',
  ↵ 'The Wayback with Ryan Sickler', 'Wiser Than Me with Julia Louis-Dreyfus', 'All There
  ↵ Is with Anderson Cooper', 'The Ezra Klein Show', 'Passion Struck with John R. Miles',
  ↵ 'Money Rehab with Nicole Lapin', 'The Money Mondays')

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 host", feedbackItem1) |
        grepl("feature a female host", feedbackItem2) |
        grepl("feature a female host", feedbackItem3)),
      episodes_shown = as.numeric(grepl("have more than 300 episodes", feedbackItem1) |
        grepl("have more than 300 episodes", feedbackItem2) |
        grepl("have more than 300 episodes", feedbackItem3)),
      duration_shown = as.numeric(grepl("have an average episode length greater than 2
        ↵ hours", feedbackItem1) |
        grepl("have an average episode length greater than 2 hours",
        ↵ feedbackItem2) |
        grepl("have an average episode length greater than 2 hours",
        ↵ feedbackItem3)),
      years_shown = as.numeric(grepl("started over 3 years ago", feedbackItem1) |
        grepl("started over 3 years ago", feedbackItem2) |
        grepl("started over 3 years ago", feedbackItem3)),
      female = case_when(`choice-7` %in% women ~ 1,
```

```

        TRUE ~ 0),
episodes_pick = case_when(`choice-7` %in% episodes ~ 1,
                           TRUE ~ 0),
duration_pick = case_when(`choice-7` %in% duration ~ 1,
                           TRUE ~ 0),
years_pick = case_when(`choice-7` %in% years ~ 1,
                           TRUE ~ 0),
encouragement = case_when(group=="encourage" ~ 1,
                           group=="base" ~ 0,
                           TRUE ~ NA),
race_code = case_when(race=="White / Caucasian" ~ 1, TRUE ~ 0),
age = as.numeric(age),
gender_code = case_when(gender=="Man" ~ 1, TRUE ~ 0),
base_gender = rowSums(across(`choice-1`:`choice-6`, ~ . %in% women))) |>
select(
  encouragement, gender_feedback, female, episodes_pick, episodes_shown,
→ duration_shown, duration_pick, years_pick, years_shown, base_gender, gender, 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, 'StudyS2B.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
<code>encouragement</code>	Binary indicator of whether the participant was randomly assigned to the encouragement condition.
<code>gender_feedback</code>	Binary indicator of whether the participant was randomly assigned to gender feedback condition.
<code>female</code>	Binary indicator of whether the participant selected a female podcast host for their seventh selection.
<code>episodes_shown</code>	Binary indicator of whether the participant was randomly assigned to receive feedback on the number of episodes in the podcast.
<code>episodes_pick</code>	Binary indicator of whether the participant selected a podcast with over 300 episodes.
<code>duration_shown</code>	Binary indicator of whether the participant was randomly assigned to receive feedback on the duration of podcasts.
<code>duration_pick</code>	Binary indicator of whether the participant selected a podcast with an average length of over 2 hours.
<code>years_shown</code>	Binary indicator of whether the participant was randomly assigned to receive feedback about when the podcast started.
<code>years_pick</code>	Binary indicator of whether the participant selected a podcast that started over 3 years ago.
<code>base_gender</code>	Count of the number of podcasts with a female host selected in the initial six podcasts.
<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: 99

##                                     Percentage gender
## 1 Another gender not listed here:      0.2
## 2                               Man    52.8
## 3                Non-binary     0.7
## 4                           Woman   46.3

##                                     Percentage Race
## 1 American Indian or Alaskan Native  0.3
## 2          Asian / Pacific Islander  7.5
## 3    Black or African American    8.4
## 4           Hispanic / Latinx     4.6
## 5        White / Caucasian  79.2

## # A tibble: 1 x 2
##   mean_age sd_age
##       <dbl>  <dbl>
## 1     43.8   12.1
```

## Primary Analysis

```
# 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 
## -0.4980 -0.4257 -0.2360  0.5020  0.7640 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 0.23600   0.02696  8.753 < 2e-16 ***
## gender_feedback 0.18970   0.04143  4.578 5.28e-06 ***
## encouragement  0.05306   0.03919  1.354  0.176    
## gender_feedback:encouragement 0.01919   0.05962  0.322  0.748  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4698 on 996 degrees of freedom
## Multiple R-squared:  0.04712,    Adjusted R-squared:  0.04425 
## F-statistic: 16.42 on 3 and 996 DF,  p-value: 2.032e-10
```

## Secondary Analysis

```
## episodes feedback
r_episodes <- lm(episodes_pick ~ episodes_shown*encouragement, data=d0)

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

```
##
## Call:
## lm(formula = episodes_pick ~ episodes_shown * encouragement,
##      data = d0)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -0.3864 -0.3552 -0.3261  0.6448  0.7381
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.38636   0.05250   7.359 3.87e-13 ***
## episodes_shown            -0.03113   0.05759  -0.541  0.5889
## encouragement             -0.12446   0.07151  -1.740  0.0821 .
## episodes_shown:encouragement  0.09537   0.07876   1.211  0.2262
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4731 on 996 degrees of freedom
## Multiple R-squared:  0.003901, Adjusted R-squared:  0.000901
## F-statistic:  1.3 on 3 and 996 DF, p-value: 0.273
```

```
robust_confint(r_episodes)
```

```
##                                2.5 %    97.5 %
## (Intercept)               0.28333643 0.48939084
## episodes_shown            -0.14414163 0.08187665
## encouragement             -0.26478483 0.01586708
## episodes_shown:encouragement -0.05918771 0.24992135
```

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

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

```
##
## Call:
## lm(formula = duration_pick ~ duration_shown * encouragement,
##      data = d0)
##
## Residuals:
```

```

##      Min     1Q   Median     3Q     Max
## -0.1832 -0.1832 -0.1667 -0.1667  0.9333
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.085714  0.048711  1.760  0.0788 .
## duration_shown            0.097475  0.051929  1.877  0.0608 .
## encouragement             -0.019048  0.061798 -0.308  0.7580
## duration_shown:encouragement 0.002525  0.066699  0.038  0.9698
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3726 on 996 degrees of freedom
## Multiple R-squared:  0.005793, Adjusted R-squared:  0.002799
## F-statistic: 1.935 on 3 and 996 DF, p-value: 0.1223

robust_confint(r_duration)

##                               2.5 %    97.5 %
## (Intercept)           -0.009872717 0.1813013
## duration_shown        -0.004426834 0.1993776
## encouragement          -0.140316871 0.1022216
## duration_shown:encouragement -0.128362575 0.1334118

## years feedback
r_years <- lm(years_pick ~ years_shown*encouragement, data=d0)

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

##
## Call:
## lm(formula = years_pick ~ years_shown * encouragement, data = d0)
##
## Residuals:
##      Min     1Q   Median     3Q     Max
## -0.6638 -0.5610  0.3362  0.4390  0.5013
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.57937   0.04433 13.069 <2e-16 ***
## years_shown              -0.08071   0.05137 -1.571   0.116
## encouragement             0.08443   0.06263  1.348   0.178
## years_shown:encouragement -0.02205   0.07238 -0.305   0.761
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4956 on 996 degrees of freedom
## Multiple R-squared:  0.01066, Adjusted R-squared:  0.007683
## F-statistic: 3.578 on 3 and 996 DF, p-value: 0.01357

```

```

robust_confint(r_years)

##                               2.5 %      97.5 %
## (Intercept)            0.49237289 0.66635727
## years_shown           -0.18151474 0.02010361
## encouragement          -0.03847683 0.20733288
## years_shown:encouragement -0.16409280 0.11999564

## 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
## -0.6215 -0.3277 -0.2363  0.4977  0.8552
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                0.32773   0.03658   8.959 < 2e-16 ***
## gender_feedback             0.29373   0.05213   5.634 2.29e-08 ***
## base_gender                 -0.04573   0.02031  -2.252   0.0245 *
## gender_feedback:base_gender -0.07343   0.02879  -2.550   0.0109 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4629 on 996 degrees of freedom
## Multiple R-squared:  0.07478,    Adjusted R-squared:  0.07199
## F-statistic: 26.83 on 3 and 996 DF,  p-value: < 2.2e-16

```

Figure S6A Code

```
dgender_plot <- d0_no_encouragement |>
  select(gender_feedback, female) |>
  group_by(gender_feedback) |>
  summarise(
    n = n(),
    freq = mean(female),
    sd = sd(female) * 100,
    se = (sd(female) / sqrt(n())) * 100
  ) |>
  mutate(gender_feedback = case_when(gender_feedback==1 ~ "\"Treatment\"",
                                       TRUE ~ "\"Control\"")) |>
  rename(Condition = gender_feedback)

## dataframe for duration information

dduration_plot <- d0_no_encouragement |>
  select(duration_shown, duration_pick) |>
  group_by(duration_shown) |>
  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)

## dataframe for episodes information

depisodes_plot <- d0_no_encouragement |>
  select(episodes_shown, episodes_pick) |>
  group_by(episodes_shown) |>
  summarise(
    n = n(),
    freq = mean(episodes_pick),
    sd = sd(episodes_pick) * 100,
    se = (sd(episodes_pick) / sqrt(n())) * 100
  ) |>
  mutate(episodes_shown = case_when(episodes_shown==1 ~ "\"Treatment\",
                                       TRUE ~ "\"Control\"")) |>
  rename(Condition = episodes_shown)

## dataframe for years information

dyears_plot <- d0_no_encouragement |>
  select(years_shown, years_pick) |>
  group_by(years_shown) |>
  summarise(
    n = n(),
    freq = mean(years_pick),
    sd = sd(years_pick) * 100,
```

```

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

df_combined <- bind_rows(
  dduration_plot %>% mutate(Category = "\nOver 3\nHours"),
  dyears_plot %>% mutate(Category = "\nStarted 3\nYears Ago"),
  depisodes_plot %>% mutate(Category = "\nWith Over\n300 Episodes"),
  dgender_plot %>% mutate(Category = "\nWith Female\nHost")
, .id = "id") %>%
  mutate(Category = factor(Category, levels = c("\nOver 3\nHours", '\nStarted 3\nYears
  ↵ Ago', "\nWith Over\n300 Episodes", '\nWith Female\nHost')))

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(data = df_combined %>% filter(!Category == "\nOver 3\nHours" & Condition ==
  ↵ "\"Control\""),
            aes(label=paste0(sprintf("%.1f", freq*100), "%")),
            position=position_dodge(width=0.7), vjust=5, size = 5, 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("\nOver 3\nHours", '\nStarted 3\nYears Ago', "\nWith
  ↵ Over\n300 Episodes", '\nWith Female\nHost')), nrow = 1, strip.position = "bottom") +
  +
  geom_segment(data = df_combined %>% filter(Category %in% c("\nOver 3\nHours", "\nWith
  ↵ Over\n300 Episodes", '\nWith Female\nHost') & Condition == "\"Treatment\""),
               aes(x = 1, xend = 2, y = freq*100 + se + 5, yend = freq*100 + se + 5),
               inherit.aes = FALSE) +
  geom_segment(data = df_combined %>% filter(Category %in% c('\nStarted 3\nYears Ago')
  ↵ & Condition == "\"Control\""),
               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("\nWith Over\n300 Episodes") &
  ↵ 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, size = 5, family = "Times New Roman") +
  geom_text(data = df_combined %>% filter(Category %in% c("\nOver 3\nHours") &
  ↵ Condition == "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 7, yend = freq*100 + se + 7,
            ↵ label = "+"),
            inherit.aes = FALSE, vjust = 0, size = 5, family = "Times New Roman") +
  geom_text(data = df_combined %>% filter(Category %in% c('\nStarted 3\nYears Ago') &
  ↵ Condition == "\"Control\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 5, yend = freq*100 + se + 5,
            ↵ label = "*"),
            inherit.aes = FALSE, vjust = 0, size = 5, family = "Times New Roman") +
  geom_text(data = df_combined %>% filter(Category == '\nWith Female\nHost' & 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, size = 5, family = "Times New Roman")

```

```

    inherit.aes = FALSE, vjust = 0, size = 5, family = "Times New Roman") +
geom_text(data = df_combined %>% filter(Category == "\nOver 3\nHours" & Condition ==
  \\"Control\"),  

  aes(label = paste0(sprintf("%.1f", freq*100), "%"),  

      x = Condition, y = freq*100 + se + 4),  

  inherit.aes = FALSE, vjust = -0.5, size = 5, color = "black", family = "Times  

  New Roman") +  

theme_bw() +  

scale_fill_manual(values = c("#011F5B", "#990000"), labels = c("No feedback provided",  

  \\"Feedback provided\", "Feedback") +  

scale_y_continuous(labels = function(x) paste0(x,"%"), limits = c(0,85)) +  

scale_x_discrete(labels = c("\\"Control\" = "Not\nShown", "\\"Treatment\" = "Shown")) +  

labs(x= "Feedback on % of podcasts...", y = "% of New Podcasts with the Target  

  \\"Attribute\") +  

theme(plot.caption = element_text(face = "italic", family = "Times New Roman"),  

  legend.position = c(0.5, 0.95),  

  legend.title = element_blank(),  

  legend.direction = "horizontal",  

  legend.text = element_text(size = 14, 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 = 14, vjust = 17, family = "Times  

  New Roman"),  

  plot.title = element_blank(),  

  axis.title.y = element_text(size = 14, color = "black", family = "Times New  

  Roman"),  

  axis.text.x = element_blank(),  

  axis.ticks = element_blank(),  

  axis.text.y = element_text(size = 14, color = "black", family = "Times New  

  Roman"),  

  strip.text = element_text(size = 14, color = "black", family = "Times New  

  Roman"),  

  strip.background = element_rect(colour = "white", fill = "white"))

```

#p\_combined\_A

Figure S6B Code

```
dgender_plot <- d0_w_encouragement |>
  select(gender_feedback, female) |>
  group_by(gender_feedback) |>
  summarise(
    n = n(),
    freq = mean(female),
    sd = sd(female) * 100,
    se = (sd(female) / sqrt(n())) * 100
  ) |>
  mutate(gender_feedback = case_when(gender_feedback==1 ~ "\"Treatment\"",
                                       TRUE ~ "\"Control\"")) |>
  rename(Condition = gender_feedback)

## dataframe for duration information

dduration_plot <- d0_w_encouragement |>
  select(duration_shown, duration_pick) |>
  group_by(duration_shown) |>
  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)

## dataframe for episodes information

depisodes_plot <- d0_w_encouragement |>
  select(episodes_shown, episodes_pick) |>
  group_by(episodes_shown) |>
  summarise(
    n = n(),
    freq = mean(episodes_pick),
    sd = sd(episodes_pick) * 100,
    se = (sd(episodes_pick) / sqrt(n())) * 100
  ) |>
  mutate(episodes_shown = case_when(episodes_shown==1 ~ "\"Treatment\",
                                       TRUE ~ "\"Control\"")) |>
  rename(Condition = episodes_shown)

## dataframe for years information

dyears_plot <- d0_w_encouragement |>
  select(years_shown, years_pick) |>
  group_by(years_shown) |>
  summarise(
    n = n(),
    freq = mean(years_pick),
    sd = sd(years_pick) * 100,
```

```

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

df_combined <- bind_rows(
  dduration_plot %>% mutate(Category = "\nOver 3\nHours"),
  dyears_plot %>% mutate(Category = "\nStarted 3\nYears Ago"),
  depisodes_plot %>% mutate(Category = "\nWith Over\n300 Episodes"),
  dgender_plot %>% mutate(Category = "\nWith Female\nHost")
, .id = "id") %>%
  mutate(Category = factor(Category, levels = c("\nOver 3\nHours", '\nStarted 3\nYears
→ Ago', "\nWith Over\n300 Episodes", '\nWith Female\nHost')))

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(data = df_combined %>% filter(!(Category == "\nOver 3\nHours")),
            aes(label=paste0(sprintf("%.1f", freq*100), "%")),
            position=position_dodge(width=0.7), vjust=5, size = 5, 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("\nOver 3\nHours", '\nStarted 3\nYears Ago', "\nWith
  ← Over\n300 Episodes", '\nWith Female\nHost')), nrow = 1, strip.position = "bottom")
  ← +
  geom_segment(data = df_combined %>% filter(Category %in% c("\nOver 3\nHours", "\nWith
  ← Over\n300 Episodes", '\nWith Female\nHost') & Condition == "\"Treatment\""),
               aes(x = 1, xend = 2, y = freq*100 + se + 5, yend = freq*100 + se + 5),
               inherit.aes = FALSE) +
  geom_segment(data = df_combined %>% filter(Category %in% c('\nStarted 3\nYears Ago')
  ← & Condition == "\"Control\""),
               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("\nWith Over\n300 Episodes") &
  ← 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, size = 5, family = "Times New Roman") +
  geom_text(data = df_combined %>% filter(Category %in% c("\nOver 3\nHours") &
  ← Condition == "\"Treatment\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 7, yend = freq*100 + se + 7,
            ← label = "*"),
            inherit.aes = FALSE, vjust = 0, size = 5, family = "Times New Roman") +
  geom_text(data = df_combined %>% filter(Category %in% c('\nStarted 3\nYears Ago') &
  ← Condition == "\"Control\""),
            aes(x = 1.5, xend = 1.5, y = freq*100 + se + 5, yend = freq*100 + se + 5,
            ← label = "*"),
            inherit.aes = FALSE, vjust = 0, size = 5, family = "Times New Roman") +
  geom_text(data = df_combined %>% filter(Category == '\nWith Female\nHost' & 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, size = 5, family = "Times New Roman") +

```

```

geom_text(data = df_combined %>% filter(Category == "\nOver 3\nHours"),
          aes(label = paste0(sprintf("%.1f", freq*100), "%"),
              x = Condition, y = 40),
          inherit.aes = FALSE, vjust = -0.5, size = 5, color = "black", family = "Times
          → New Roman") +
theme_bw() +
scale_fill_manual(values = c("#011F5B", "#990000"), labels = c("No feedback provided",
→ "Feedback provided"), "Feedback") +
scale_y_continuous(labels = function(x) paste0(x, "%"), limits = c(0,110)) +
scale_x_discrete(labels = c("\\"Control\\" = "Not\nShown", "\\"Treatment\\" = "Shown")) +
labs(x = "Feedback on % of podcasts...", y = "% of New Podcasts with the Target
→ Attribute") +
theme(plot.caption = element_text(face = "italic", family = "Times New Roman"),
      legend.position = "none",
      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 = 14, vjust = 17, family = "Times
      → New Roman"),
      plot.title = element_blank(),
      axis.title.y = element_text(size = 14, color = "black", family = "Times New
      → Roman"),
      axis.text.x = element_blank(),
      axis.ticks = element_blank(),
      axis.text.y = element_text(size = 14, color = "black", family = "Times New
      → Roman"),
      strip.text = element_text(size = 14, color = "black", family = "Times New
      → Roman"),
      strip.background = element_rect(colour = "white", fill = "white"))

#p_combined_B

# Save individual plot objects as RDS for combination in figure-S6.R
saveRDS(p_combined_A, file = "p_combined_S2B_A.rds")
saveRDS(p_combined_B, file = "p_combined_S2B_B.rds")

```