Modeling_Thesis

OliviaShin

6/5/2022

```
library(lme4)
library(ggplot2)
library(lmerTest)
library(ggeffects)
```

```
library(emmeans)
library(sjPlot)
library(formattable)
```

```
library(tidyverse)
```

```
sample_df <- read.csv("sample_df.csv")
sample_df
```

```
sample_df_thesis <- sample_df %>% select(-X.1, -X, -race_ver1, -race_ver2, -cohort_ve
r2, -gender_ver2, -mean_close_base_C, -temp_totalfriends_Base_C, -enrollment_code, -r
ace_Inter, -race_URM, -gender_ver3, -totfr_postint)
sample_df_thesis
```

```
str(sample_df)
```

```
sample_df$program <- factor(sample_df$program)

sample_df$condition <- factor(sample_df$condition)
sample_df$condition <- relevel(sample_df$condition, ref="Control")

sample_df$raceURM <- factor(sample_df$raceURM)

sample_df$school <- factor(sample_df$school)

sample_df$cohort_ver1 <- factor(sample_df$cohort_ver1)

sample_df$gender_ver1 <- factor(sample_df$gender_ver1)

sample_df$gender_ver3 <- ifelse(sample_df$gender_ver1 == "M", 0, 1)
```

```
str(sample_df)
```

```
# for summary statistics of variables #

sample_summary <- sample_df %>% select(condition, school, cohort_ver1, gender_ver1, p
rogram, mean_close_base, mean_close_w5_EOW, temp_totalfriends_Base, temp_totalfriends
_w5_EOW, enroll_full)

sample_summary_d <- data.frame(unclass(summary(sample_summary)), check.names = FALSE,
stringsAsFactors = FALSE)


sample_summary_d [is.na(sample_summary_d )] <- ""
```

```
names(sample_summary_d) <- c("Condition", "School", "Cohort", "Gender", "Program", "M
ean Close Base", "Mean Close EOW 5", "Total number of Friends Base", "Total number of
Friends EOW 5", "Enrollment")

sample_summary_d
```

```
# Summary Statistics of Independent Variables
IV <- sample_summary_d %>% select(Gender, Condition, School, Cohort, Program, "Mean C
lose Base", "Total number of Friends Base")
formattable(IV)
```

```
# Summary Statistics of Dependent Variables
DV <- sample_summary_d %>% select("Mean Close EOW 5", "Total number of Friends EOW
5", "Enrollment")
formattable(DV)
```

```
sample_df %>%
  summarise_all(funs(sum(is.na(.))))
```

```
total <- sample_df %>% summarise(enroll_mean = mean(enroll_full), enroll_sd = sd(enro
ll_full), mean_close_w5_mean = mean(mean_close_w5_EOW), mean_close_w5_sd = sd(mean_cl
ose_w5_EOW), totfriends_w5_mean = mean(temp_totalfriends_w5_EOW), totfriends_w5_sd =
sd(temp_totalfriends_w5_EOW))

by_condition <- sample_df %>% group_by(condition) %>% summarise(enroll_mean = mean(en
roll_full), enroll_sd = sd(enroll_full), mean_close_w5_mean = mean(mean_close_w5_EO
W), mean_close_w5_sd = sd(mean_close_w5_EOW), totfriends_w5_mean = mean(temp_totalfri
ends_w5_EOW), totfriends_w5_sd = sd(temp_totalfriends_w5_EOW))

# for gender
by_gender <- sample_df %>% group_by(condition, gender_ver1) %>% summarise(enroll_mean
= mean(enroll_full), enroll_sd = sd(enroll_full), mean_close_w5_mean = mean(mean_clos
e_w5_EOW), mean_close_w5_sd = sd(mean_close_w5_EOW), totfriends_w5_mean = mean(temp_t
otalfriends_w5_EOW), totfriends_w5_sd = sd(temp_totalfriends_w5_EOW))

# for race
by_race <- sample_df %>% group_by(condition, raceURM) %>% summarise(enroll_mean = mea
n(enroll_full), enroll_sd = sd(enroll_full), mean_close_w5_mean = mean(mean_close_w5_
EOW), mean_close_w5_sd = sd(mean_close_w5_EOW), totfriends_w5_mean = mean(temp_totalf
riends_w5_EOW), totfriends_w5_sd = sd(temp_totalfriends_w5_EOW))
```

```
total <- rename(total,
       "Enrollment - Mean" = enroll_mean,
       "Enrollment - SD" = enroll_sd,
       "Mean Close EOW 5 - Mean" = mean_close_w5_mean,
       "Mean Close EOW 5 - SD" = mean_close_w5_sd,
       "Total number of Friends EOW 5 - Mean" = totfriends_w5_mean,
       "Total number of Friends EOW 5 - SD" = totfriends_w5_sd
       )

total
```

```
# entire sample, summary statistics

formattable(as.data.frame(t(total)))
```

```
by_condition
```

```
by_gender <- rename(by_gender,
       "Condition" = condition,
       "Gender" = gender_ver1,
        "Enrollment - Mean" = enroll_mean,
        "Enrollment - SD" = enroll_sd,
        "Mean Close EOW 5 - Mean" = mean_close_w5_mean,
        "Mean Close EOW 5 - SD" = mean_close_w5_sd,
        "Total number of Friends EOW 5 - Mean" = totfriends_w5_mean,
        "Total number of Friends EOW 5 - SD" = totfriends_w5_sd
        )


formattable(by_gender)
```

Mean Closeness W5

```
# set conditionControl  as a reference group, so that we see Belonging, Affirmation.

fit1_close_w5 <- lmerTest::lmer(mean_close_w5_EOW ~ condition + mean_close_base_C + c
ohort_ver2 + (1|school/program), data=sample_df)
#summary(fit1_close_w5, correlation = FALSE)
```

```
fit2_close_w5 <- lmerTest::lmer(mean_close_w5_EOW ~ condition*gender_ver2 + mean_clos
e_base_C + cohort_ver2 + (1|school/program), data=sample_df)
summary(fit2_close_w5, correlation = FALSE)
# gender_ver2: "M" = -0.5, "F" = 0.5
# cohort_ver2: 1 => -0.5, 2 => 0.5
```

```
# with sjPlot package

tab_model(fit2_close_w5,
          pred.labels = c("Intercept", "Condition_Affirmation","Condition_Belonging",
"Gender", "Mean Close Base", "Cohort", "Condition_Affirmation * Gender" ,"Condition_B
elonging * Gender"),
          dv.labels = c("Mean Close EOW 5"))
```

```
## Estimated marginal means: Condition x Gender
emm <- emmeans(fit2_close_w5, specs = ~ condition | gender_ver2)
emm
#"M" = -0.5, "F" = 0.5
```

```
cont <- contrast(emm, interaction = "pairwise")
cont_d <- as.data.frame(cont)
cont_d$gender_ver2 <- as.factor(cont_d$gender_ver2)
cont_d$gender_ver2 <- fct_recode(cont_d$gender_ver2, "Male" = "-0.5", "Female" = "0.
5")
cont_d <- rename(cont_d, Gender = gender_ver2, Condition_Pairwise = condition_pairwis
e)
formattable(cont_d)
```

```
# Contrast: Male vs. Female -------------------

## Pairwise comparisons of male and female in each condition

male <- c(1,0)
female <- c(0,1)

cont_gender <- contrast(emm, method = list("Female - Male" = female - male), by = "co
ndition")
cont_gender_d <- as.data.frame(cont_gender)
formattable(cont_gender_d)
```

```
#Emmeans: these are the estimated marginal means for those six groups

emm_d <- as.data.frame(emm)
emm_d$gender_ver2 <- as.factor(emm_d$gender_ver2) # gender_ver2: "M" = -0.5, "F" = 0.
5
emm_d$gender_ver2 <- fct_recode(emm_d$gender_ver2, "Male" = "-0.5", "Female" = "0.5")
emm_d <- rename(emm_d, Gender = gender_ver2, Condition = condition)
emm_d
```

```
# with formattable package
formattable(emm_d)
```

Total number of friends W5

```
fit1_total_w5 <- lmerTest::lmer(temp_totalfriends_w5_EOW ~ condition + temp_totalfrie
nds_Base_C + cohort_ver2 + (1|school/program), data=sample_df)
#summary(fit1_total_w5, correlation = FALSE)
```

```
fit2_total_w5 <- lmerTest::lmer(temp_totalfriends_w5_EOW ~ condition*gender_ver2  + t
emp_totalfriends_Base_C + cohort_ver2 + (1|school/program), data=sample_df)
summary(fit2_total_w5, correlation = FALSE)
```

```
# with sjPlot package

tab_model(fit2_total_w5,
          pred.labels = c("Intercept", "Condition_Affirmation","Condition_Belonging",
"Gender", "Total number of friends Base", "Cohort", "Condition_Affirmation * Gender"
,"Condition_Belonging * Gender"),
          dv.labels = c("Total number of friends EOW 5"))
```

```
## Estimated marginal means: Condition x Gender
emm2 <- emmeans(fit2_total_w5, specs = ~ condition | gender_ver2)
emm2

emm2_d <- as.data.frame(emm2)
emm2_d$gender_ver2 <- as.factor(emm2_d$gender_ver2) # gender_ver2: "M" = -0.5, "F" = 0.5
emm2_d$gender_ver2 <- fct_recode(emm2_d$gender_ver2, "Male" = "-0.5", "Female" = "0.5")
emm2_d <- rename(emm2_d, Gender = gender_ver2, Condition = condition)
emm2_d
```

```
# with formattable package
formattable(emm2_d)
```

```
cont2 <- contrast(emm2, interaction = "pairwise")
cont_d2 <- as.data.frame(cont2)
cont_d2$gender_ver2 <- as.factor(cont_d2$gender_ver2)
cont_d2$gender_ver2 <- fct_recode(cont_d2$gender_ver2, "Male" = "-0.5", "Female" = "0.5")
cont_d2 <- rename(cont_d2, Gender = gender_ver2, Condition_Pairwise = condition_pairwise)
formattable(cont_d2)
```

```
# Contrast: Male vs. Female ------------------

## Pairwise comparisons of male and female in each condition

male <- c(1,0)
female <- c(0,1)

cont_gender2 <- contrast(emm2, method = list("Female - Male" = female - male), by = "condition")
cont_gender_d2 <- as.data.frame(cont_gender2)
formattable(cont_gender_d2)
```

```
as.data.frame(ggemmeans(fit2_total_w5, c("condition", "gender_ver2"))) %>% arrange(group)
```

Enrollment

```
#enroll_full
fit1_enroll <- glmer(enroll_full ~ condition + cohort_ver2 + school + (1|program), data=sample_df, family=binomial(link="logit"), control = glmerControl(optimizer ="bobyqa"))
#summary(fit1_enroll, correlation = FALSE)
```

```
# gender_ver2: "M" = -0.5, "F" = 0.5
#enroll_full
fit2_enroll <- glmer(enroll_full ~ condition*gender_ver2 + cohort_ver2 + school + (1|
program), data=sample_df, family=binomial(link="logit"), control = glmerControl(optim
izer ="bobyqa"))
summary(fit2_enroll, correlation = FALSE)
```

```
# with sjPlot package

tab_model(fit2_enroll,
          pred.labels = c("Intercept", "Condition_Affirmation","Condition_Belonging",
"Gender", "Cohort", "School_PennState", "School_Stanford", "Condition_Affirmation * G
ender" ,"Condition_Belonging * Gender"),
          dv.labels = c("Enrollment"))
```

```
## Estimated marginal means: Condition x Gender
emm3 <- emmeans(fit2_enroll, specs = ~ condition | gender_ver2)
emm3

emm3_d <- as.data.frame(emm3)
emm3_d$gender_ver2 <- as.factor(emm3_d$gender_ver2) # gender_ver2: "M" = -0.5, "F" =
0.5
emm3_d$gender_ver2 <- fct_recode(emm3_d$gender_ver2, "Male" = "-0.5", "Female" = "0.
5")
emm3_d <- rename(emm3_d, Gender = gender_ver2, Condition = condition)
emm3_d
```

```
cont3 <- emm3 %>%
regrid() %>%
contrast(interaction = "pairwise", type="response")
```

```
cont_d3 <- as.data.frame(cont3)
cont_d3$gender_ver2 <- as.factor(cont_d3$gender_ver2)
cont_d3$gender_ver2 <- fct_recode(cont_d3$gender_ver2, "Male" = "-0.5", "Female" =
"0.5")
cont_d3 <- rename(cont_d3, Gender = gender_ver2, Condition_Pairwise = condition_pairw
ise)
formattable(cont_d3)
```

```
# Contrast: Male vs. Female -------------------

## Pairwise comparisons of male and female in each condition

male <- c(1,0)
female <- c(0,1)

cont_gender3 <- contrast(emm3, method = list("Female - Male" = female - male), by =
"condition", type="response")
cont_gender_d3 <- as.data.frame(cont_gender3)
formattable(cont_gender_d3)
```

```
cont_gender3
```

```
ggemmeans(fit2_enroll, c("condition", "gender_ver2"))
plot(ggemmeans(fit2_enroll, c("condition", "gender_ver2"))) +
  labs(x="Condition", y = "P(Enrollment)") +
  ggtitle("Predicted Probabilities of Enrollment") +
  scale_colour_discrete(name  ="Gender",
                            breaks=c("-0.5", "0.5"),
                            labels=c("Male", "Female"))
```

```
emm_enroll <- as.data.frame(ggemmeans(fit2_enroll, c("condition", "gender_ver2"))) %
>% arrange(group)

emm_enroll$group <- fct_recode(emm_enroll$group, "Male" = "-0.5", "Female" = "0.5")

emm_enroll <- rename(emm_enroll, Condition = x)

emm_enroll <- rename(emm_enroll, Gender = group)

emm_enroll <- emm_enroll %>% select(Condition, Gender, predicted, std.error, conf.lo
w, conf.high)

emm_enroll

formattable(emm_enroll)
```

ego network - post intervention friends visualization

OliviaShin

5/11/2022

Total number of Friends

```
library(dplyr)
```

```
match <- read.csv("more_than_one_friend_760.csv", header = FALSE)

match <- as.data.frame(t(match))
head(match)
```

```
library(janitor)
```

```
match <- match %>% row_to_names(row_number = 1)

colnames(match)[1] <- "PID"
```

```
match
```

```
matchT <- as.data.frame(t(match)) %>% row_to_names(row_number = 1)
matchT
```

```
# Exclude Baseline survey friends - Only for the post_intervention
post_int_df <- matchT[-c(1:6),]
post_int_df
```

```
library(stringr)
library(tidyr)
library(tidyverse)
```

```
firsttwo <- reduce(seq_along(post_int_df),
       .init = post_int_df,
       ~ .x %>% separate(names(post_int_df)[.y],
                         sep = ' ',
                         into = paste0(names(post_int_df)[.y], '_col_' , 1:2),
                         fill = 'right'
                         )
       )
firsttwo
```

```
# to lower case

firsttwo_post_intervention <- data.frame(lapply(firsttwo, function(v) {
  if (is.character(v)) return(tolower(v))
  else return(v)
}))
```

```
firsttwo_post_intervention
```

```
firstthree_post_intervention <- data.frame(lapply(firsttwo_post_intervention, functio
n(v) {
  if (is.character(v)) return(substr(v, 1, 3))
  else return(v)
}))

firstthree_post_intervention
```

```
i <- seq.int(1L, length(firstthree_post_intervention), by = 2L)

post_intervention <- data.frame(mapply(paste0, firstthree_post_intervention[i], first
three_post_intervention[i + 1]))
```

```
colnames(post_intervention) <- substr(colnames(post_intervention), 2, 9)

post_intervention
```

```
post_intervention <- post_intervention %>% na_if("NANA")
post_intervention
```

```
df <- as.data.frame(t(post_intervention))
df$PID <- rownames(df)
df
```

```
df2 <- df %>% pivot_longer(-PID, names_to = "friend_num", values_to = "friend_names")
%>% group_by(PID) %>% distinct(friend_names) %>% na.omit() %>% mutate(frid = paste0
("fr",as.character(row_number(PID))), totfr_postint = n_distinct(friend_names))
df2
```

```
temp_df2 <- df2 %>% select(PID, frid)
temp_df2
```

```
frid <- temp_df2 %>%
  pivot_wider(names_from  = frid, values_from = frid)
frid <- as.data.frame(t(frid))
frid
```

```
# totfr_postint = total friends post intervention
colnames(frid) <- frid[1,]
frid <- frid[-1,]
frid
```

```
#write.csv(frid, "confidential_postint_friends.csv")
```

```
# totfr_postint = total friends post intervention
post_intervention
```

```
# Calculate total number of friends, post intervention. = total number of friends, ex
cept for the baseline survey friends
df3 <- df2 %>% select(PID, totfr_postint) %>% distinct()
df3
```

```
#write.csv(df3, "unique_total_friends.csv")
```

Visualize Ego-Network Examples

```
library(igraph)
library(ggraph)
library(patchwork)
```

```
fr_df <- temp_df2 %>% filter(PID %in% c("1C012522"))
fr_graph <- igraph::graph_from_data_frame(
  d = fr_df, directed = TRUE
)
g <- ggraph(fr_graph, layout = "fr") +
  geom_edge_link(color = "black", alpha = 0.7) +
   geom_node_label(aes(label = name), color = "#CC79A7") +
  theme_void()

fr_df1 <- temp_df2 %>% filter(PID %in% c("1C787697"))
fr_graph1 <- igraph::graph_from_data_frame(
  d = fr_df1, directed = TRUE
)
g1 <- ggraph(fr_graph1, layout = "fr") +
  geom_edge_link(color = "black", alpha = 0.7) +
   geom_node_label(aes(label = name), color = "#D55E00") +
  theme_void()

fr_df2 <- temp_df2 %>% filter(PID %in% c("2S069175"))
fr_graph2 <- igraph::graph_from_data_frame(
  d = fr_df2, directed = TRUE
)
g2 <- ggraph(fr_graph2, layout = "fr") +
  geom_edge_link(color = "black", alpha = 0.7) +
   geom_node_label(aes(label = name), color = "#0072B2") +
  theme_void()

fr_df3 <- temp_df2 %>% filter(PID %in% c("1P206167"))
fr_graph3 <- igraph::graph_from_data_frame(
  d = fr_df3, directed = TRUE
)
g3 <- ggraph(fr_graph3, layout = "fr") +
  geom_edge_link(color = "black", alpha = 0.7) +
   geom_node_label(aes(label = name), color = "#009E73") +
  theme_void()

fr_df4 <- temp_df2 %>% filter(PID %in% c("1P214797"))
fr_graph4 <- igraph::graph_from_data_frame(
  d = fr_df4, directed = TRUE
)
```

```
# Ego-Network Examples
g1 + plot_spacer() + g + plot_spacer() + g2 + plot_spacer() + g3
```