PC_English_Full

2024-11-18

Cleaning

```
library(readr)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v purrr     1.0.2
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)
library(ggsignif)
library(mlogit)
```

```
## Loading required package: dfidx
##
## Attaching package: 'dfidx'
##
## The following object is masked from 'package:stats':
##
##     filter
```

```
data <- read_csv("PC_Pilot-English_Prolific_December 12, 2024_17.20.csv")
```

```
## Rows: 403 Columns: 81
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (81): StartDate, EndDate, Status, IPAddress, Progress, Duration (in seco...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
View(data)

data_header_rm <- data[-c(1,2),] #remove qualtrics duplicate headers
```

```r
data_header_rm[data_header_rm == ""] <- NA #empty string = NA
View(data_header_rm)

# Prolific, pass consent, and prescreen
data_passed <- data_header_rm %>%
  filter(
    Finished == 1,
    consent == 1,
    str_length(Prolific_ID) == 24, # Prolific IDs
  )

#rows with fraud
data_fraud <- data_passed %>%
  filter(
    Q_BallotBoxStuffing == "True",
    Q_RecaptchaScore <= 0.5,
    Q_RelevantIDDuplicate == "True",
    Q_RelevantIDDuplicateScore >= 75,
    Q_RelevantIDFraudScore >= 30
  )

#remove unnecessary columns
data_passed1 <- data_passed %>%
  filter(!PROLIFIC_PID %in% data_fraud$PROLIFIC_PID) %>% #remove fraud
  select(-c(StartDate,
            EndDate,
            Status,
           IPAddress,
           Progress,
           Finished,
           RecordedDate,
           ResponseId,
           RecipientLastName,
           RecipientFirstName,
           RecipientEmail,
           ExternalReference,
           LocationLatitude,
           LocationLongitude,
           DistributionChannel,
           UserLanguage,
           'Duration (in seconds)',
           Q_BallotBoxStuffing,
           Q_RecaptchaScore,
           Q_RelevantIDDuplicate,
           Q_RelevantIDDuplicateScore,
           Q_RelevantIDFraudScore,
           PROLIFIC_PID,
           consent,
           not_pc_label,
           pc_label
           )) %>%
  mutate(ppt_ID = row_number())
```

```r
# Separate demographics
data_demo <- data_passed1 %>%
  select(ppt_ID, Prolific_ID,
         label,
         starts_with("Demo_"),
         comment
         )

write.csv(data_demo, "demo_12102024.csv", row.names = FALSE)

# Main data
data_main <- data_passed1 %>%
  select(-c(Prolific_ID, # de-identified
            starts_with("Demo_"),
            comment)) %>%
  mutate(across(where(is.character), # Apply to character columns only
                ~ ifelse(!is.na(suppressWarnings(as.numeric(.))),
                         as.numeric(.),
                         .))) %>%
  mutate(
    experience_1 = as.numeric(experience_1),
    experience_2 = as.numeric(experience_2),
    experience_3 = as.numeric(experience_3),
    SliderOverlapValue = as.numeric(SliderOverlapValue)
  )

# Attention check
data_main <- data_main[data_main$check == 2, ]
```

Recode Data

```r
# In primary and early_language, change 4=English and other to 3 = English and other.
data_main <- data_main %>%
  mutate(
    primary = if_else(primary == 4, 3, primary),
    early_language = if_else(early_language == 4, 3, early_language)
  )

# For all stigma_, 5 = the most stigmatized, 1 = the least stigmatized
## Recode stigma_resp_2, stigma_dist_2, stigma_dist_3
data_main_recode <- data_main %>%
  mutate(stigma_resp_2_mut = 6 - stigma_resp_2,
         stigma_dist_2_mut = 6 - stigma_dist_2,
         stigma_dist_3_mut = 6 - stigma_dist_3) %>%
  select(-c(stigma_resp_2, stigma_dist_2, stigma_dist_3))

# For all stertyp_, 5 = the most stereotyped (negatively)
## Reverse all values in stertyp_ columns
data_main_recode <- data_main_recode %>%
  mutate(across(
    starts_with("stertyp_"), # Select columns starting with "stertyp_"
    ~ 6 - ., # Reverse the values (assuming range is 1 to 5)
```

```
    .names = "{.col}_rev" # Add "_rev" to the new column names
  )) %>%
  select(-starts_with("stertyp_")|ends_with("_rev"))

data_main_noFR <- data_main_recode %>%
  select(-c(general_1, general_2, impression_1, impression_2,impression_3_4_TEXT, like_fr, offens_fr))
write.csv(data_main_noFR, "data_main_noFR.csv", row.names = FALSE)
```

Analysis - Simple Descriptive

Lable meaning overlap

```
# Overlap compare
data_main_recode %>%
  group_by(label) %>%
  summarise(overlap_avg = mean(SliderOverlapValue))
```

```
## # A tibble: 2 x 2
##   label                                    overlap_avg
##   <chr>                                          <dbl>
## 1 people who experience housing insecurity          NA
## 2 the homeless                                      NA
```

Stigma: PC vs. PIC

```
# average stigma
data_main_recode %>%
  mutate(stigma_avg_ppt = rowSums(select(., starts_with('stigma_')))/6) %>%
  group_by(label) %>%
  summarise(stigma_avg_label = mean(stigma_avg_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stigma_avg_label
##   <chr>                                               <dbl>
## 1 people who experience housing insecurity             2.07
## 2 the homeless                                         2.28
```

```
# stigma_resp
data_main_recode %>%
  mutate(stigma_resp_ppt = rowSums(select(., starts_with('stigma_resp')))/2) %>%
  group_by(label) %>%
  summarise(stigma_resp_label = mean(stigma_resp_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stigma_resp_label
##   <chr>                                                <dbl>
## 1 people who experience housing insecurity              2.22
## 2 the homeless                                          2.16
```

```r
# stigma_dist
data_main_recode %>%
  mutate(stigma_dist_ppt = rowSums(select(., starts_with('stigma_dist')))/3) %>%
  group_by(label) %>%
  summarise(stigma_dist_label = mean(stigma_dist_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stigma_dist_label
##   <chr>                                                <dbl>
## 1 people who experience housing insecurity              1.98
## 2 the homeless                                          2.25
```

```r
# stigma_danger
data_main_recode %>%
  group_by(label) %>%
  summarise(stigma_danger_label = mean(stigma_danger))
```

```
## # A tibble: 2 x 2
##   label                                    stigma_danger_label
##   <chr>                                                  <dbl>
## 1 people who experience housing insecurity                2.04
## 2 the homeless                                            2.61
```

Stereotype: PC vs. PIC

```r
# average stertyp
data_main_recode %>%
  mutate(stertyp_avg_ppt = rowSums(select(., starts_with('stertyp_')))/8) %>%
  group_by(label) %>%
  summarise(stertyp_avg_label = mean(stertyp_avg_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stertyp_avg_label
##   <chr>                                                <dbl>
## 1 people who experience housing insecurity              3.37
## 2 the homeless                                          3.51
```

```r
# stertyp_cmptition
data_main_recode %>%
  mutate(stertyp_cmptition_ppt = rowSums(select(., starts_with('stertyp_cmptition_')))/2) %>%
  group_by(label) %>%
  summarise(stertyp_cmptition_label = mean(stertyp_cmptition_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stertyp_cmptition_label
##   <chr>                                                      <dbl>
## 1 people who experience housing insecurity                    3.90
## 2 the homeless                                                4.10
```

```r
# stertyp_comp
data_main_recode %>%
  mutate(stertyp_comp_ppt = rowSums(select(., starts_with('stertyp_comp_')))/2) %>%
  group_by(label) %>%
  summarise(stertyp_comp_label = mean(stertyp_comp_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stertyp_comp_label
##   <chr>                                                 <dbl>
## 1 people who experience housing insecurity               3.27
## 2 the homeless                                           3.33
```

```r
# stertyp_warm
data_main_recode %>%
  mutate(stertyp_warm_ppt = rowSums(select(., starts_with('stertyp_warm_')))/2) %>%
  group_by(label) %>%
  summarise(stertyp_warm_label = mean(stertyp_warm_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stertyp_warm_label
##   <chr>                                                 <dbl>
## 1 people who experience housing insecurity               2.56
## 2 the homeless                                           2.74
```

```r
# stertyp_status
data_main_recode %>%
  mutate(stertyp_status_ppt = rowSums(select(., starts_with('stertyp_status_')))/2) %>%
  group_by(label) %>%
  summarise(stertyp_status_label = mean(stertyp_status_ppt))
```

```
## # A tibble: 2 x 2
##   label                                    stertyp_status_label
##   <chr>                                                  <dbl>
## 1 people who experience housing insecurity                3.74
## 2 the homeless                                            3.89
```

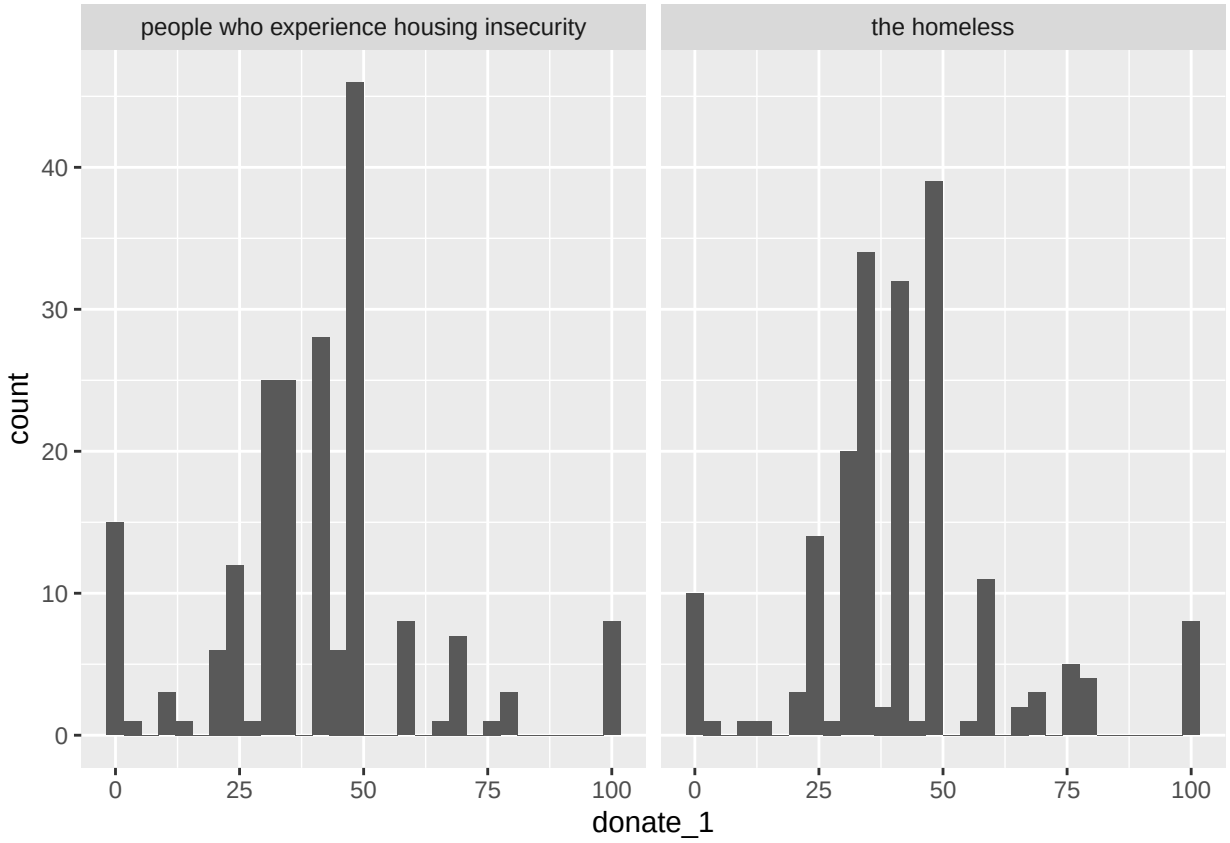
Supportive Behavior: PC vs. PIC

```r
# Average donation: PC vs. PIC
data_main_recode %>%
  group_by(label) %>%
  summarise(donate_homeless_label = mean(donate_1), donate_homeless_sd = sd(donate_1))
```

```
## # A tibble: 2 x 3
##   label                                donate_homeless_label donate_homeless_sd
##   <chr>                                                <dbl>               <dbl>
## 1 people who experience housing insecu~                 40.3                21.1
## 2 the homeless                                          42.1                20.4
```

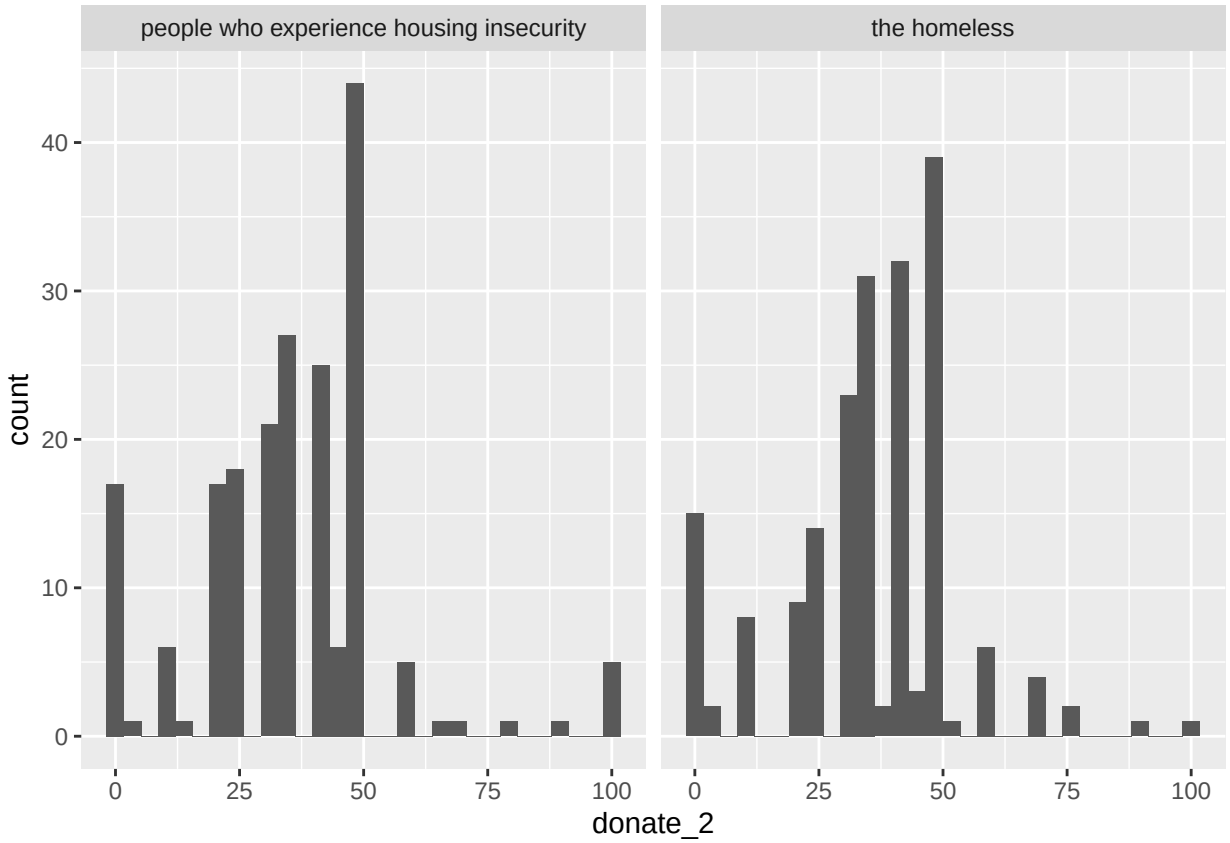
```
ggplot(data_main_recode, aes(x = donate_1)) +
  geom_histogram() +
  facet_wrap(~label)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



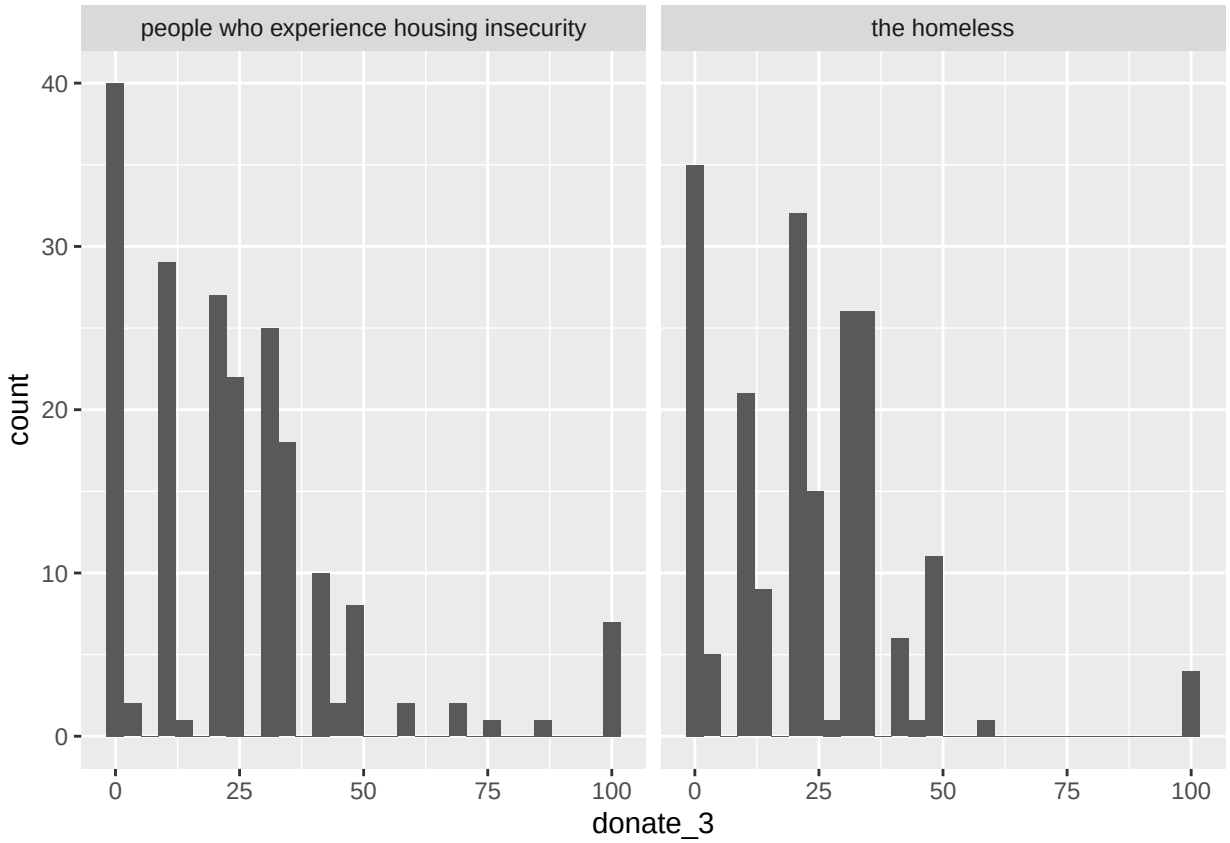

```
ggplot(data_main_recode, aes(x = donate_2)) +
  geom_histogram() +
  facet_wrap(~label)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(data_main_recode, aes(x = donate_3)) +
  geom_histogram() +
  facet_wrap(~label)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Concept: PC vs. PIC

```
# concept_1: access to shelter
data_main_recode %>%
  group_by(label) %>%
  summarise(concept_shelter_lable = mean(concept_1))
```

```
## # A tibble: 2 x 2
##   label                                    concept_shelter_lable
##   <chr>                                                    <dbl>
## 1 people who experience housing insecurity                  3.04
## 2 the homeless                                              2.94
```

```
# concept_2: unemployment
data_main_recode %>%
  group_by(label) %>%
  summarise(concept_unemploy_lable = mean(concept_2))
```

```
## # A tibble: 2 x 2
##   label                                    concept_unemploy_lable
##   <chr>                                                     <dbl>
## 1 people who experience housing insecurity                   2.67
## 2 the homeless                                               3.18
```

```r
# concept_3: duration of homelessness
data_main_recode %>%
  group_by(label) %>%
  summarise(concept_duration_lable = mean(concept_3))
```

```
## # A tibble: 2 x 2
##   label                                      concept_duration_lable
##   <chr>                                                       <dbl>
## 1 people who experience housing insecurity                     3.26
## 2 the homeless                                                 3.32
```

Like better & Less offensive: PC vs. PIC

```r
# Average like, offens: if distinct from 1.5
data_main_recode %>%
  summarise(like_avg_ppt = mean(like),
            offense_avg_ppt = mean(offens))
```

```
## # A tibble: 1 x 2
##   like_avg_ppt offense_avg_ppt
##          <dbl>           <dbl>
## 1          1.5            1.77
```

```r
## like, offens: if correlated with label
data_main_recode %>%
  group_by(label) %>%
  summarise(like_avg_label = mean(like),
            offense_avg_label = mean(offens))
```

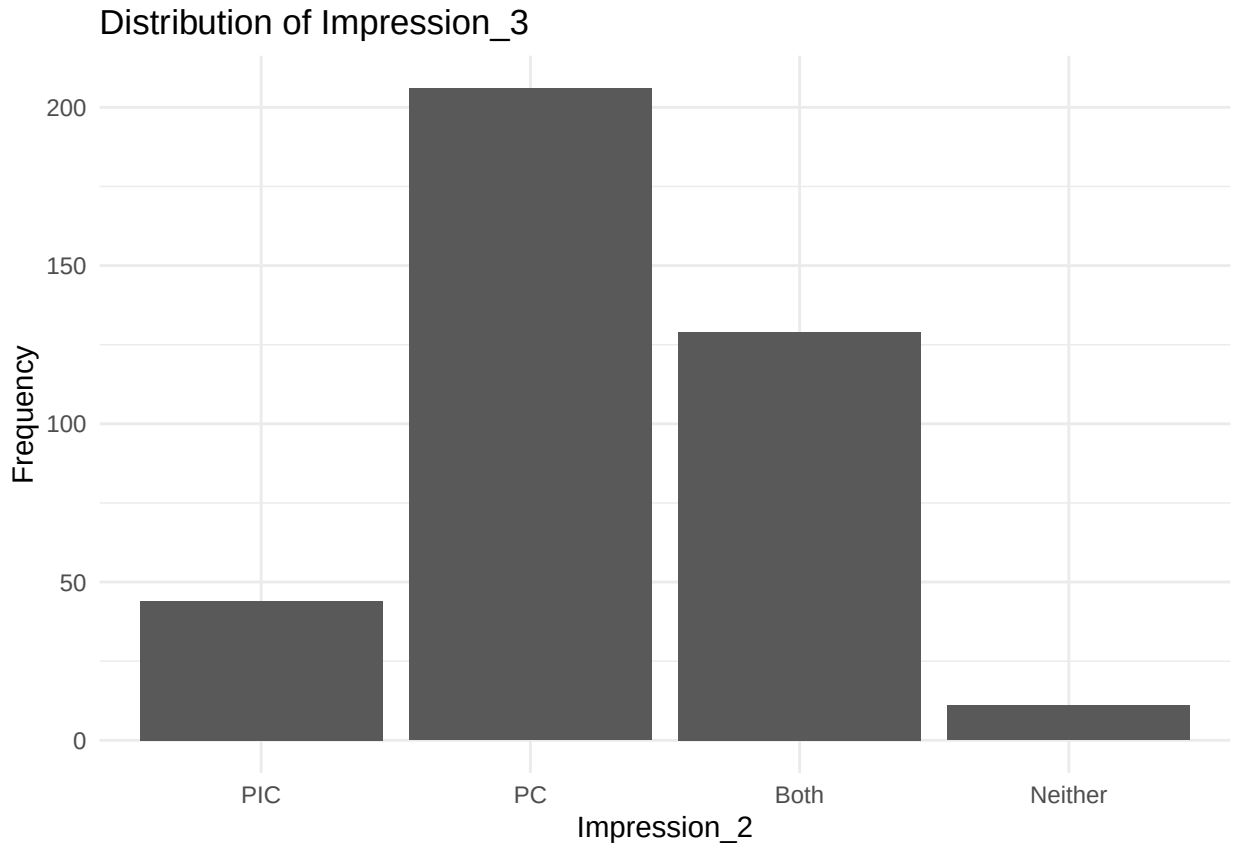
```
## # A tibble: 2 x 3
##   label                                      like_avg_label offense_avg_label
##   <chr>                                               <dbl>             <dbl>
## 1 people who experience housing insecurity             1.59              1.84
## 2 the homeless                                         1.41              1.70
```

Impression - Which one is more politically correct

```r
# Distribution of impression_3
data_main_recode %>%
  summarise(impression_3_avg = mean(impression_3))
```

```
## # A tibble: 1 x 1
##   impression_3_avg
##              <dbl>
## 1             2.27
```

```r
data_main_recode$impression_3 <- factor(data_main$impression_3,
                            levels = c(1, 2, 3, 4),
                            labels = c("PIC", "PC", "Both", "Neither"))

ggplot(data_main_recode, aes(x = impression_3)) +
  geom_bar() +
  labs(title = "Distribution of Impression_3",
       x = "Impression_2",
       y = "Frequency") +
  theme_minimal()
```




```r
data_main_recode %>%
  summarise(familiar_1_avg_ppt = mean(familiar_1),
            familiar_2_avg_ppt = mean(familiar_2))
```

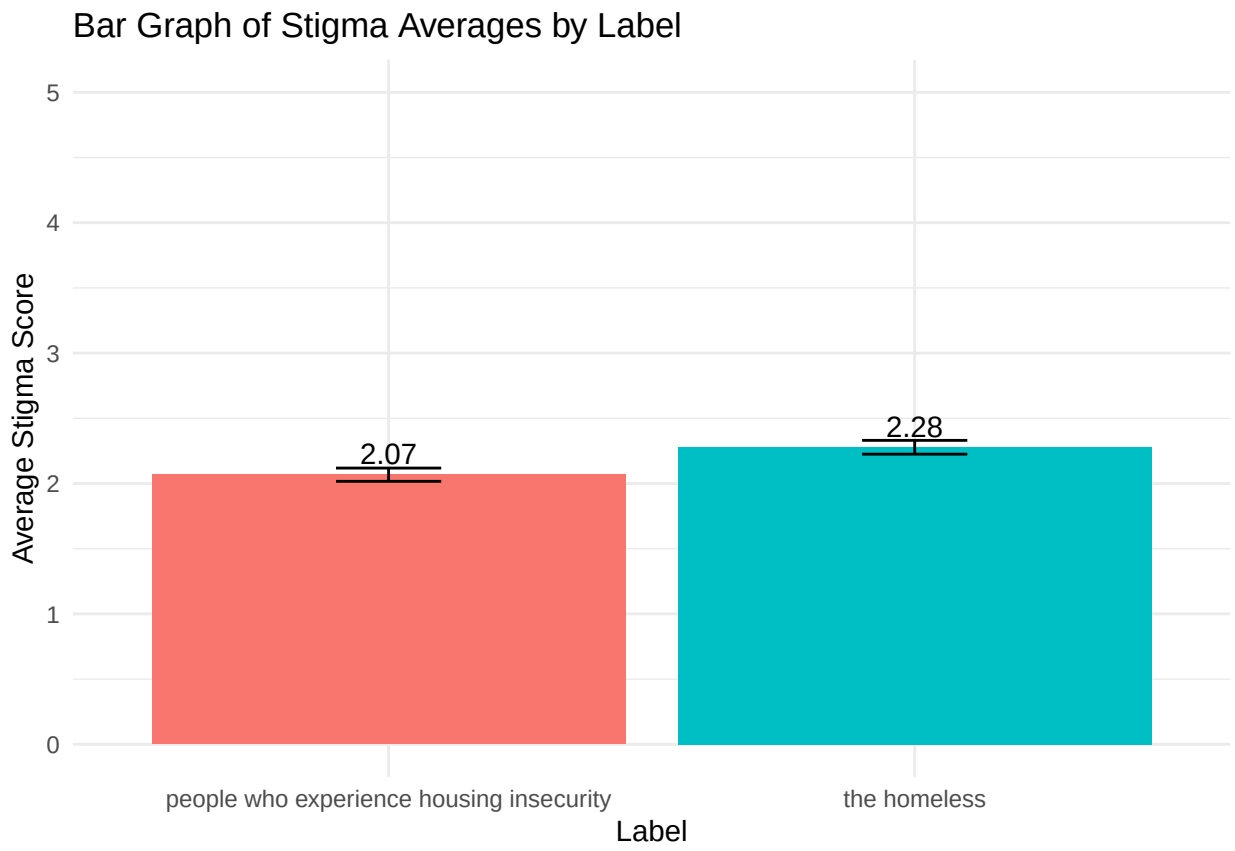
```
## # A tibble: 1 x 2
##   familiar_1_avg_ppt familiar_2_avg_ppt
##                <dbl>              <dbl>
## 1               4.15               3.22
```

Analysis - F Test

```r
# Calculate stigma and stereotype as averages
data_main_recode <- data_main_recode %>%
  mutate(
    stigma_avg = rowMeans(select(., starts_with("stigma_")), na.rm = TRUE),
    stereotype_avg = rowMeans(select(., starts_with("stertyp_")), na.rm = TRUE)
  )

# Perform F-tests for each variable
# 1. Stigma
stigma_aov <- aov(stigma_avg ~ label, data = data_main_recode)
summary(stigma_aov)
```

```
##              Df Sum Sq Mean Sq F value  Pr(>F)
## label         1   4.32   4.315   8.245 0.00431 **
## Residuals   388 203.06   0.523
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
## plot stigma by label
summary_stigma_aov <- data_main_recode %>%
  group_by(label) %>%
  summarise(
    stigma_mean = mean(stigma_avg),
    stigma_se = sd(stigma_avg)/sqrt(n())
  )

ggplot(summary_stigma_aov, aes(x = label, y = stigma_mean, fill = label)) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_errorbar(aes(ymin = stigma_mean - stigma_se,
                    ymax = stigma_mean + stigma_se),
                width = 0.2) +
  geom_text(aes(label = round(stigma_mean, 2), y = stigma_mean), vjust = -0.5) +
  labs(title = "Bar Graph of Stigma Averages by Label",
    x = "Label",
    y = "Average Stigma Score") +
  ylim(0, 5) +
  theme_minimal() +
  theme(legend.position = "none")
```

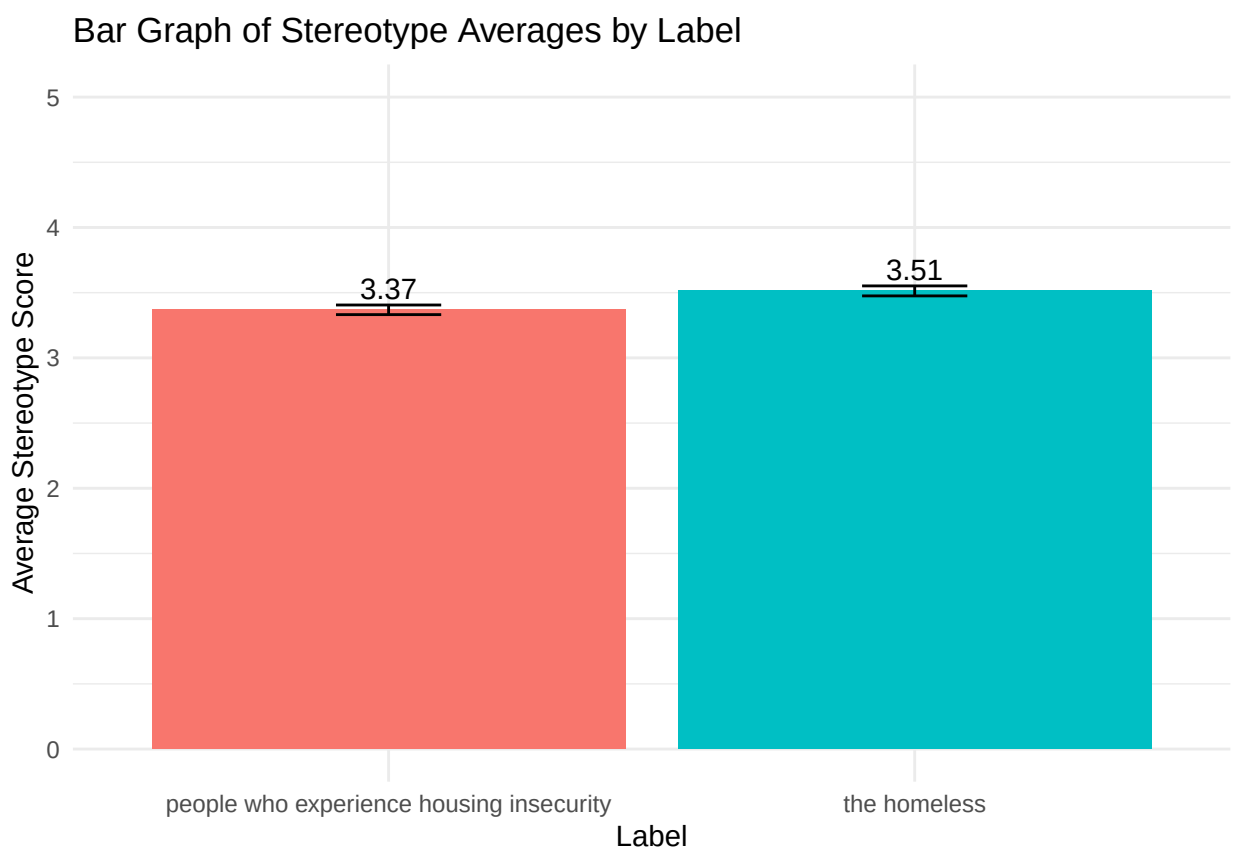



```
# 2. Stereotype
stereotype_aov <- aov(stereotype_avg ~ label, data = data_main_recode)
summary(stereotype_aov)
```

```
##              Df Sum Sq Mean Sq F value  Pr(>F)
## label         1   2.05  2.0482   7.433 0.00669 **
## Residuals   388 106.91  0.2755
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## plot stertyp by label
summary_stereotype_aov <- data_main_recode %>%
  group_by(label) %>%
  summarise(
    stereotype_mean = mean(stereotype_avg),
    stereotype_se = sd(stereotype_avg)/sqrt(n())
  )

ggplot(summary_stereotype_aov, aes(x = label, y = stereotype_mean, fill = label)) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_errorbar(aes(ymin = stereotype_mean - stereotype_se,
                    ymax = stereotype_mean + stereotype_se),
                width = 0.2) +
  geom_text(aes(label = round(stereotype_mean, 2), y = stereotype_mean), vjust = -0.5) +
  labs(title = "Bar Graph of Stereotype Averages by Label",
```

```
    x = "Label",
    y = "Average Stereotype Score") +
  ylim(0, 5) +
  theme_minimal() +
  theme(legend.position = "none")
```

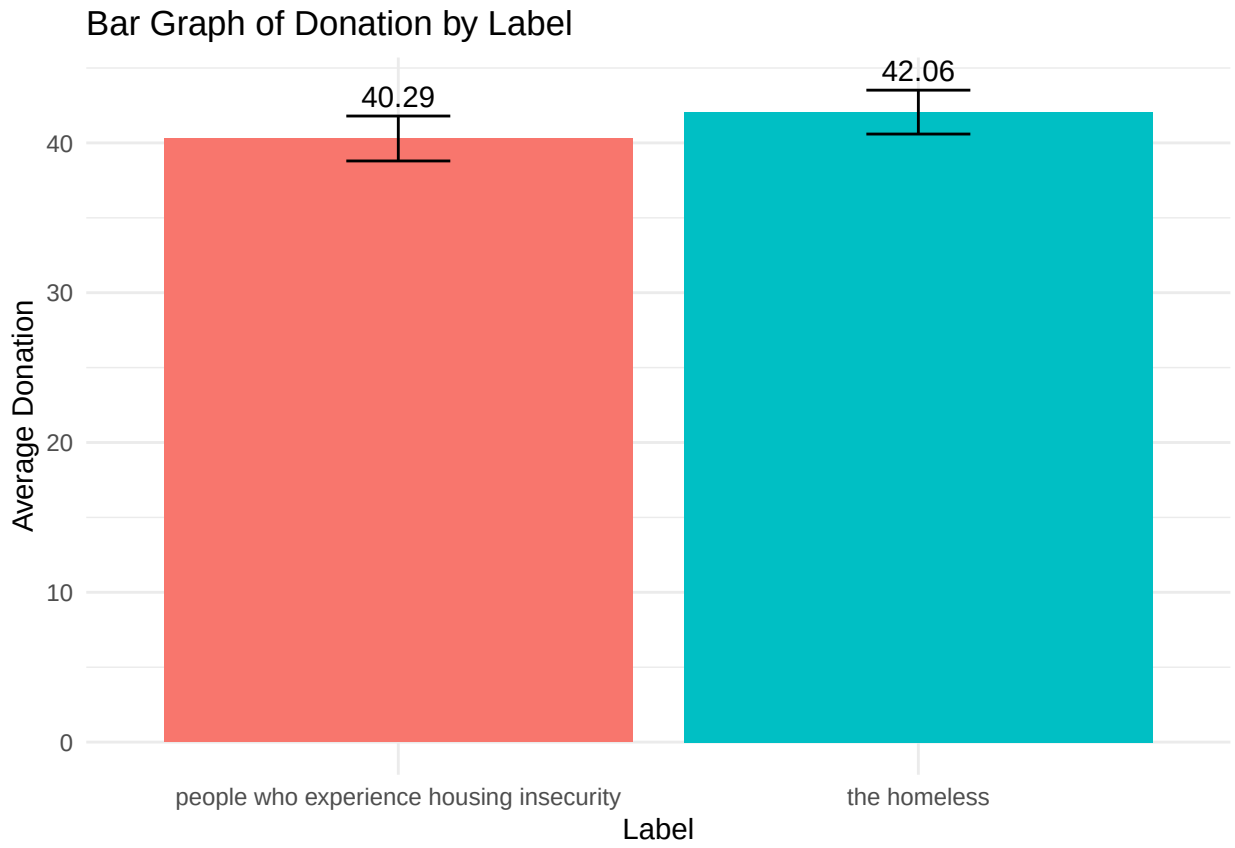



```
# 3. Supportive Behavior
supportive_aov <- aov(donate_1 ~ label, data = data_main_recode)
summary(supportive_aov)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## label         1    303   302.6   0.705  0.402
## Residuals   388 166470   429.0
```

```
## plot donate_1 by label
summary_donate_aov <- data_main_recode %>%
  group_by(label) %>%
  summarise(
    donate_mean = mean(donate_1),
    donate_se = sd(donate_1)/sqrt(n())
  )

ggplot(summary_donate_aov, aes(x = label, y = donate_mean, fill = label)) +
  geom_bar(stat = "identity", position = "dodge") +
```

```
  geom_errorbar(aes(ymin = donate_mean - donate_se,
                    ymax = donate_mean + donate_se),
                width = 0.2) +
  geom_text(aes(label = round(donate_mean, 2), y = donate_mean), vjust = -1.5) +
  labs(title = "Bar Graph of Donation by Label",
    x = "Label",
    y = "Average Donation") +
  theme_minimal() +
  theme(legend.position = "none")
```



Significant difference in stigma (F(1, 389) = 8.245, p = 0.0043) and stereotype (F(1, 389) = 7.433, p = 0.0067) between PC and PIC. No significant difference in donation (F(1, 389) = 0.705, p = 0.402).

Analysis - Correlation (Pearson)

Correlation between stigma and stereotype

```
# Pearson correlation between stigma and stereotype
cor_stigma_stereotype <- cor.test(data_main_recode$stigma_avg, data_main_recode$stereotype_avg, method =

cor_stigma_stereotype
```

```
##
```

```
##  Pearson's product-moment correlation
##
## data:  data_main_recode$stigma_avg and data_main_recode$stereotype_avg
## t = 2.1662, df = 388, p-value = 0.0309
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.01012078 0.20637515
## sample estimates:
##       cor
## 0.1093131
```

Stigma and stereotype are positively correlated ($r = 0.45$, $p < 0.001$).

Correlation between stigma and donate_1

```
# Pearson correlation between stigma and donate_1
cor_stigma_donate <- cor.test(data_main_recode$stigma_avg, data_main_recode$donate_1, method = "pearson

cor_stigma_donate
```

```
##
##  Pearson's product-moment correlation
##
## data:  data_main_recode$stigma_avg and data_main_recode$donate_1
## t = -5.0138, df = 388, p-value = 8.134e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3377003 -0.1510682
## sample estimates:
##      cor
## -0.24667
```

Correlation between stereotype and donate_1

```
# Pearson correlation between stereotype and donate_1
cor_stereotype_donate <- cor.test(data_main_recode$stereotype_avg, data_main_recode$donate_1, method =

cor_stereotype_donate
```

```
##
##  Pearson's product-moment correlation
##
## data:  data_main_recode$stereotype_avg and data_main_recode$donate_1
## t = 0.091396, df = 388, p-value = 0.9272
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.09470599  0.10389421
## sample estimates:
##         cor
## 0.004639861
```

Analysis - Multinomial Logistic Regression

```
# Model 1: Stata

# Model 2: Cleaning
## political orientation
data_main_recode <- data_main_recode %>%
  mutate(political = rowMeans(select(., c(political_social, political_econ)), na.rm = TRUE))

## label preference
### like: 1 = the homeless/PIC, 2 = people who experience housing insecurity/PC

## experience: 1 = have personal or close experience, 0 = don't have experience
data_main_recode <- data_main_recode %>%
  mutate(experience = ifelse(experience_1 == 1 | experience_2 == 1, 1, 0))

# Replace NA with 0 in 'experience'
data_main_recode$experience[is.na(data_main_recode$experience)] <- 0

## empathy
data_main_recode <- data_main_recode %>%
  mutate(empathy = rowMeans(select(., starts_with("empathy_")), na.rm = TRUE))


data_main_noFR <- data_main_recode %>%
  select(-c(general_1, general_2, impression_1, impression_2,impression_3_4_TEXT, like_fr, offens_fr))
write.csv(data_main_noFR, "data_main_noFR.csv", row.names = FALSE)
```

Exploratory Analysis

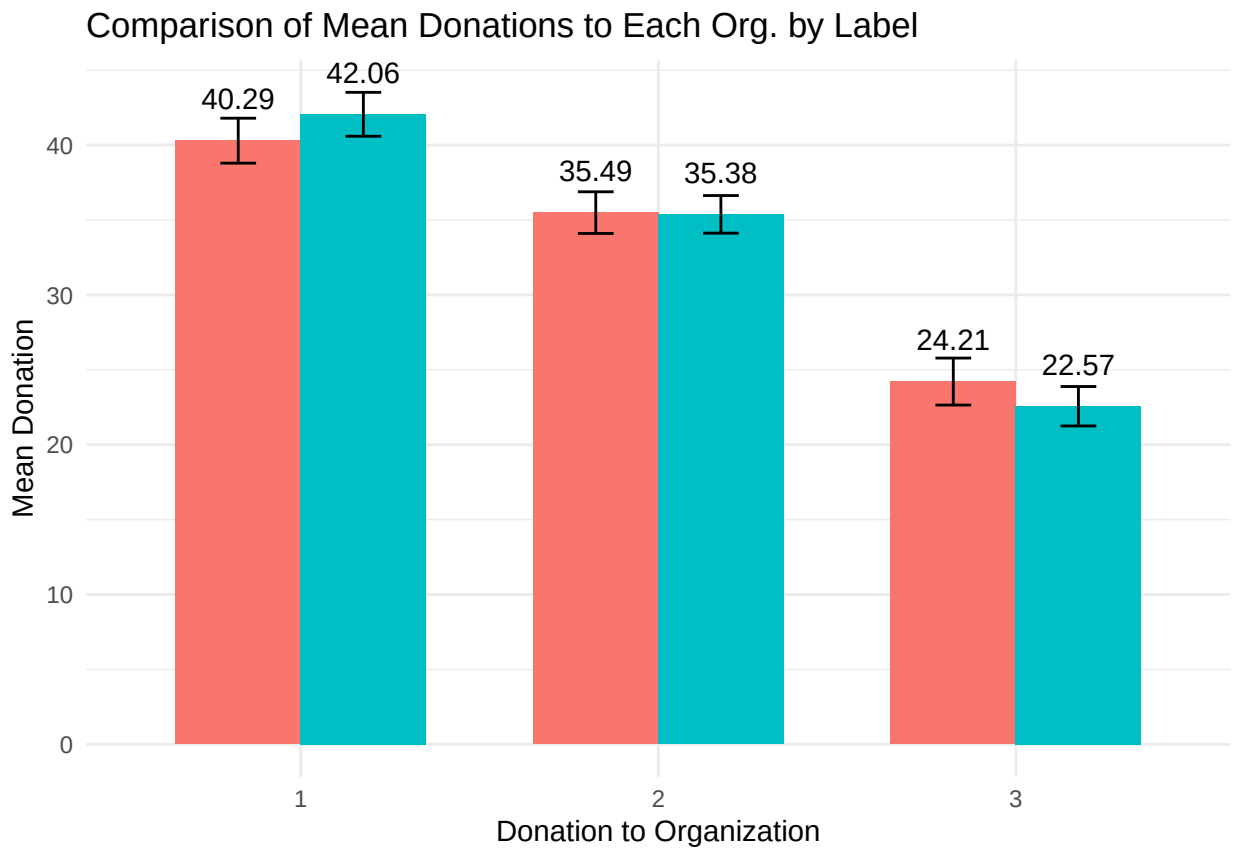
Distribution among three organizations (donate_1, donate_2, donate_3)

```
data_main_recode %>%
  group_by(label) %>%
  summarise(donate_1_mean = mean(donate_1), donate_2_mean = mean(donate_2), donate_3_mean = mean(donate
```

```
## # A tibble: 2 x 4
##   label                                donate_1_mean donate_2_mean donate_3_mean
##   <chr>                                        <dbl>         <dbl>         <dbl>
## 1 people who experience housing insec~          40.3          35.5          24.2
## 2 the homeless                                  42.1          35.4          22.6
```

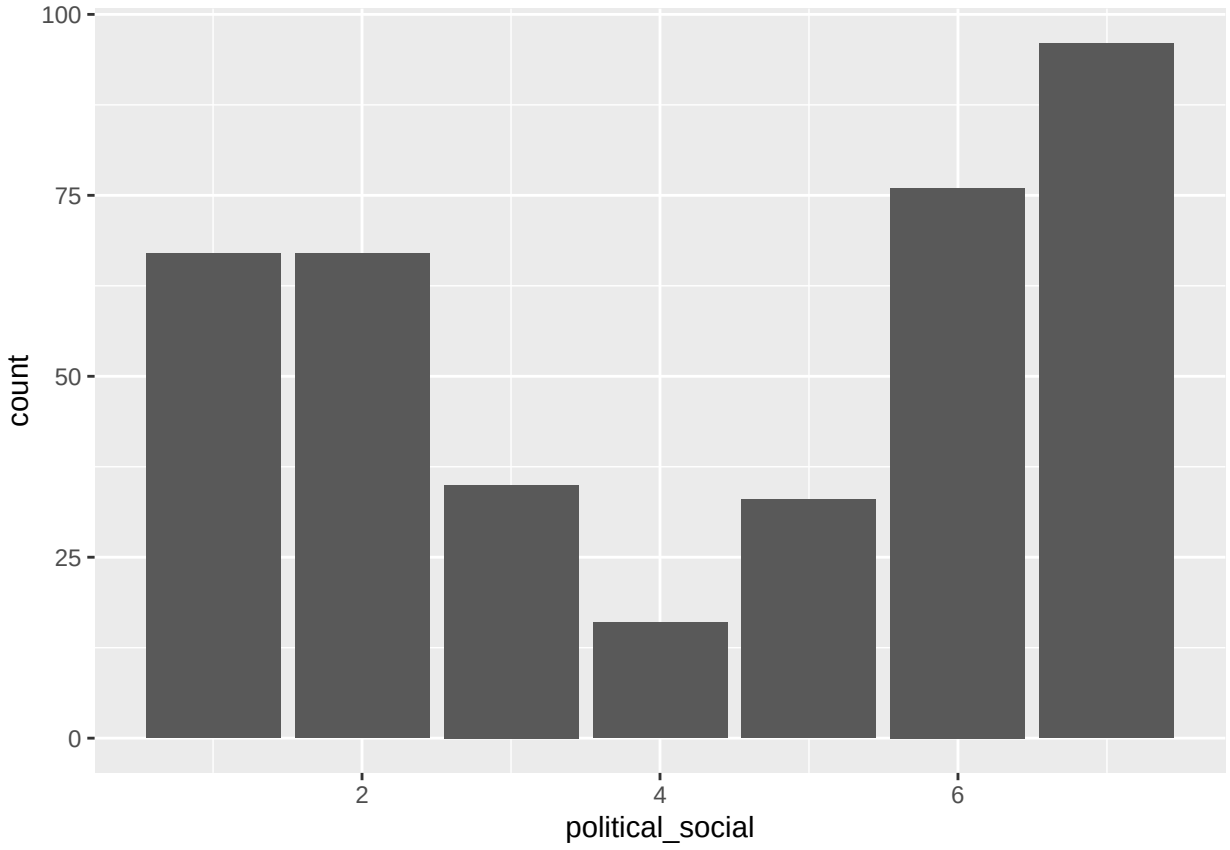
```
# Bar graph of donation by label
summary_donate_all <- data_main_recode %>%
  group_by(label) %>%
  summarise(
    mean_1 = mean(donate_1),
    se_1 = sd(donate_1)/sqrt(n()),
    mean_2 = mean(donate_2),
    se_2 = sd(donate_2)/sqrt(n()),
```

```
    mean_3 = mean(donate_3),
    se_3 = sd(donate_3)/sqrt(n())
  )

summary_donate_all_long <- summary_donate_all %>%
  pivot_longer(
    cols =  mean_1:se_3,  # Columns to reshape
    names_to = c("metric", "donate_type"),  # Split names into two parts
    names_sep = "_",  # Separator between "mean" and type
    values_to = "value"  # New column for the reshaped values
  ) %>%
  pivot_wider(
    names_from = metric,  # Spread 'mean' and 'se' into columns
    values_from = value
  )

ggplot(summary_donate_all_long, aes(x = donate_type, y = mean, fill = label)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.7) +
  geom_errorbar(aes(ymin = mean - se,
                    ymax = mean + se),
                position = position_dodge(0.7),
                width = 0.2) +
  geom_text(aes(label = round(mean, 2), y = mean),
            position = position_dodge(0.7),
            vjust = -1.5) +
  labs(
    title = "Comparison of Mean Donations to Each Org. by Label",
    x = "Donation to Organization",
    y = "Mean Donation",
    fill = "Label"
  ) +
  theme_minimal() +
  theme(legend.position = "none")
```



Political Orientation

```
# political_soical distribution
ggplot(data_main_recode, aes(x = political_social)) +
  geom_bar()
```

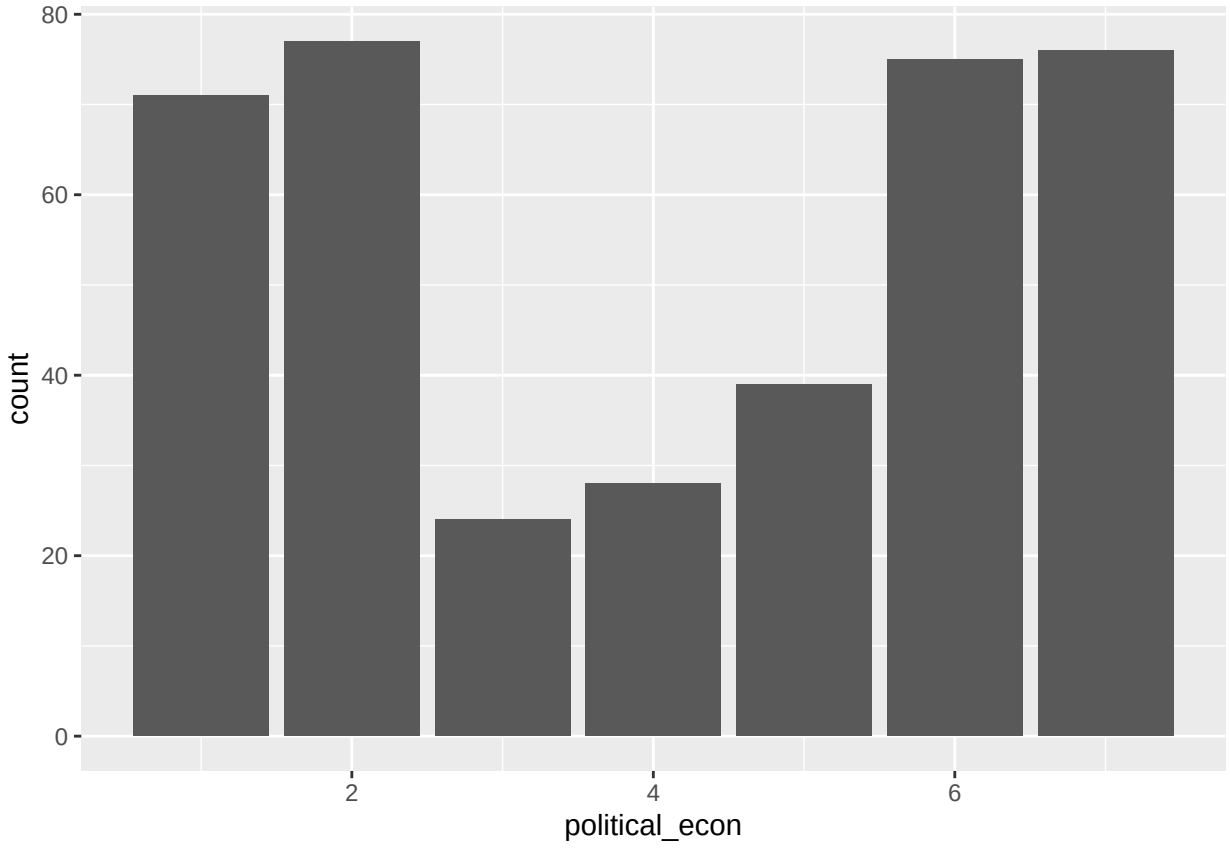



```
# Recode political_social
data_main_politic_re <- data_main_recode %>%
  filter(political_social != 4) %>%
  mutate(political_social_con = if_else(political_social < 4, "conservative", "liberal"))

# Average donation by political_social
data_main_politic_re %>% group_by(label,political_social_con) %>%
  summarise
```

```
## `summarise()` has grouped output by 'label'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 2
## # Groups:   label [2]
##   label                                  political_social_con
##   <chr>                                  <chr>
## 1 people who experience housing insecurity conservative
## 2 people who experience housing insecurity liberal
## 3 the homeless                           conservative
## 4 the homeless                           liberal
```

```
# political_econ distribution
ggplot(data_main_recode, aes(x = political_econ)) +
  geom_bar()
```




```
# Recode political_econ
data_main_politic_re <- data_main_politic_re %>%
  filter(political_econ != 4) %>%
  mutate(political_econ_con = if_else(political_econ < 4, "conservative", "liberal"))

# Average donation by political_econ
data_main_politic_re %>% group_by(label,political_econ_con) %>%
  summarise(donate_1_mean = mean(donate_1))
```

```
## `summarise()` has grouped output by 'label'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 3
## # Groups:   label [2]
##   label                                    political_econ_con donate_1_mean
##   <chr>                                    <chr>                       <dbl>
## 1 people who experience housing insecurity conservative               40.1
## 2 people who experience housing insecurity liberal                    41.3
## 3 the homeless                             conservative               40.1
## 4 the homeless                             liberal                    44.3
```

Between conservative and liberal, is there difference in the effect of label on how much they donate?

```r
# 2(label)x2(political_social_con) ANOVA for donate_1
politic_social_aov <- aov(donate_1 ~ label * political_social_con, data = data_main_politic_re)

summary(politic_social_aov)
```

```
##                              Df Sum Sq Mean Sq F value Pr(>F)
## label                         1    179   178.6   0.413  0.521
## political_social_con          1    342   342.2   0.792  0.374
## label:political_social_con    1     59    59.1   0.137  0.712
## Residuals                   349 150828   432.2
```

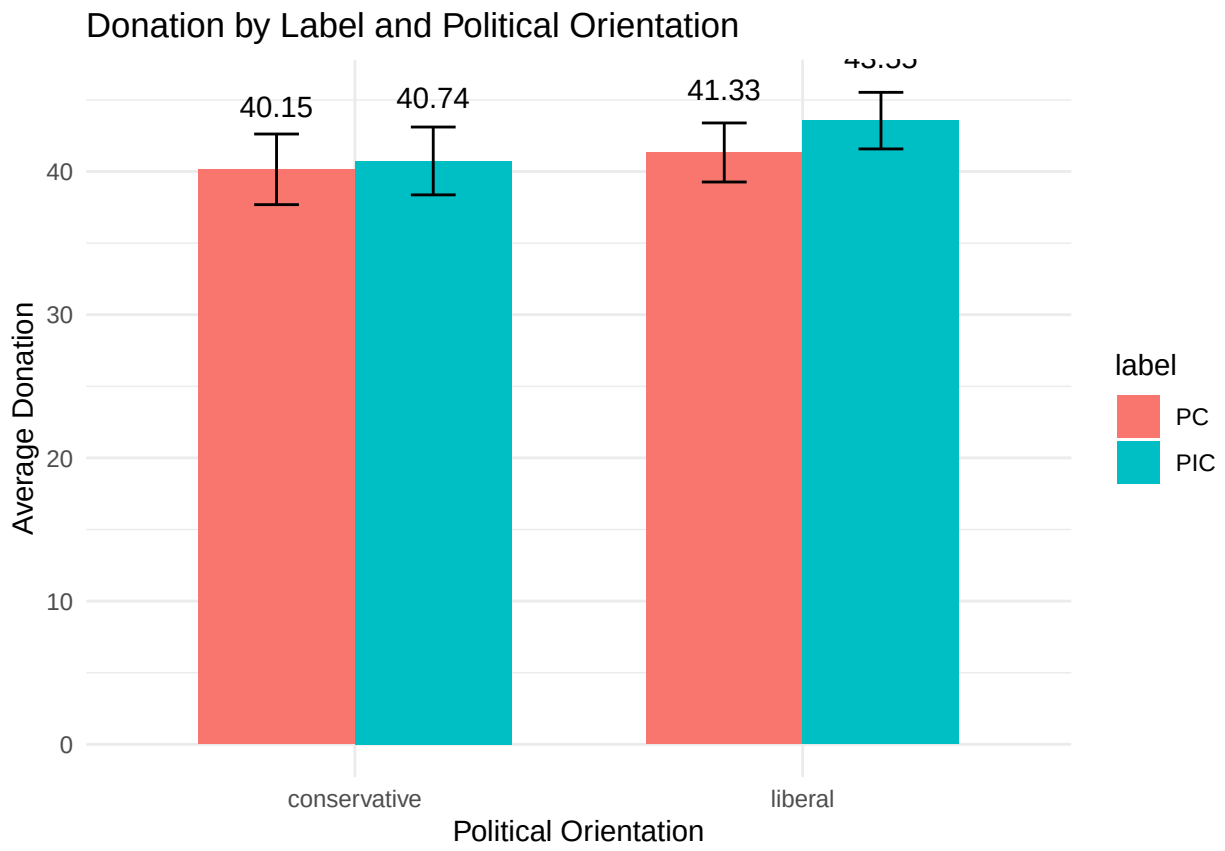
There is no significant interaction effect between label and political_social_con on donate_1 ($F(1, 389) = 0.137$, $p = 0.712$).

Plot the political orientation effect on donation by label

```r
summary_donate_politic_aov <- data_main_politic_re %>%
  group_by(label, political_social_con) %>%
  summarise(
    donate_mean = mean(donate_1),
    donate_se = sd(donate_1)/sqrt(n())
  ) %>%
  mutate(label = ifelse(label == "the homeless", "PIC", "PC"))
```

```
## `summarise()` has grouped output by 'label'. You can override using the
## `.groups` argument.
```

```r
ggplot(summary_donate_politic_aov, aes(x = political_social_con, y = donate_mean, fill = label)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.7) +
  geom_errorbar(aes(ymin = donate_mean - donate_se,
                    ymax = donate_mean + donate_se),
                position = position_dodge(0.7),
                width = 0.2) +
  geom_text(aes(label = round(donate_mean, 2), y = donate_mean),
            vjust = -2.5,
            position = position_dodge(0.7)) +
  labs(title = "Donation by Label and Political Orientation",
    x = "Political Orientation",
    y = "Average Donation") +
  theme_minimal()
```



Political Orientation and Stigma

```
# Stigma by political_social
data_main_politic_re %>% group_by(political_social_con) %>%
  summarise(stigma_avg = mean(stigma_avg))
```

```
## # A tibble: 2 x 2
##   political_social_con stigma_avg
##   <chr>                      <dbl>
## 1 conservative                2.43
## 2 liberal                     1.92
```

```
stigma_politic_aov <- aov(stigma_avg ~ political_social_con, data = data_main_politic_re)
summary(stigma_politic_aov)
```

```
##                       Df Sum Sq Mean Sq F value   Pr(>F)
## political_social_con   1  22.47  22.468   45.27 6.99e-11 ***
## Residuals            351 174.19   0.496
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Stigma by label and political_social
data_main_politic_re %>% group_by(label,political_social_con) %>%
  summarise(stigma_avg = mean(stigma_avg))
```

```
## `summarise()` has grouped output by 'label'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 3
## # Groups:   label [2]
##   label                                  political_social_con stigma_avg
##   <chr>                                  <chr>                       <dbl>
## 1 people who experience housing insecurity conservative              2.29
## 2 people who experience housing insecurity liberal                   1.85
## 3 the homeless                           conservative                2.57
## 4 the homeless                           liberal                     2.01
```

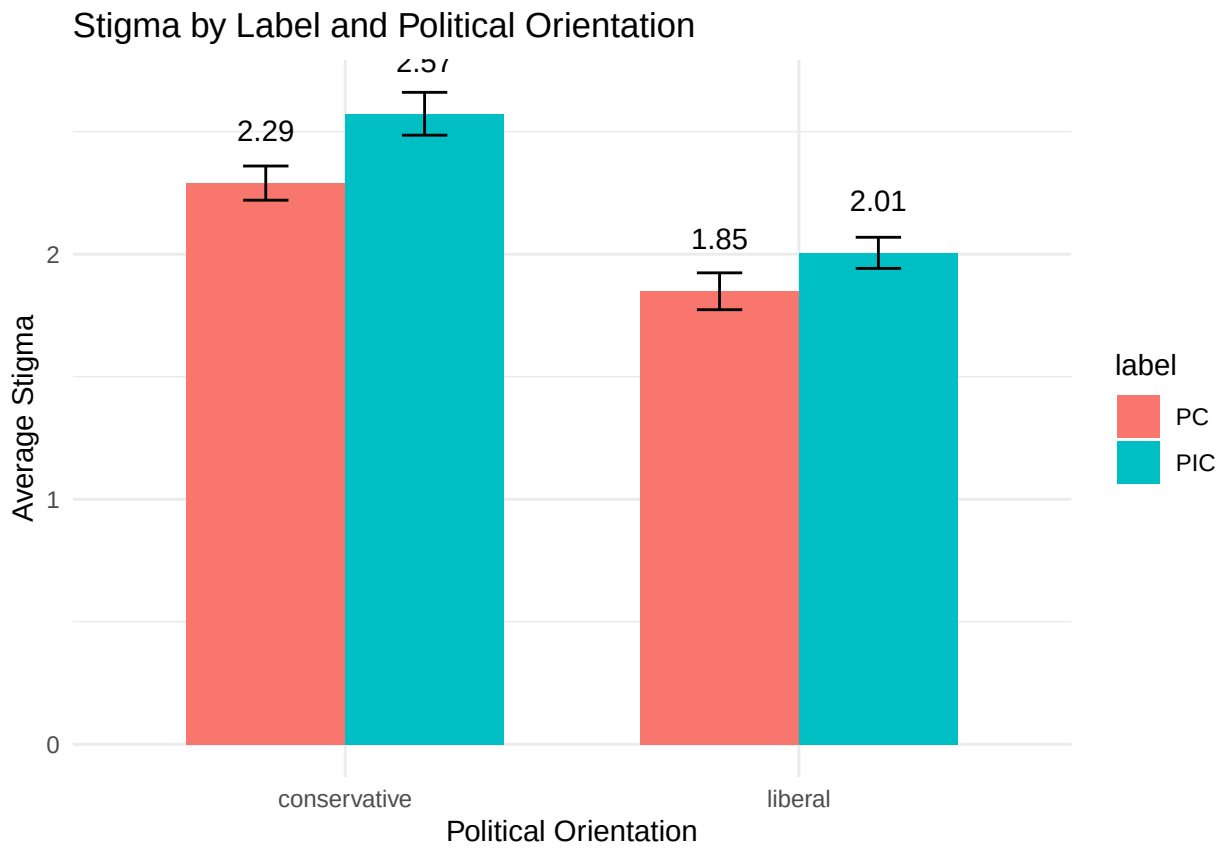
```r
stigma_label_politic_aov <- aov(stigma_avg ~ label * political_social_con, data = data_main_politic_re)

summary(stigma_label_politic_aov)
```

```
##                            Df Sum Sq Mean Sq F value  Pr(>F)
## label                       1   4.38   4.378   9.003 0.00289 **
## political_social_con        1  22.22  22.224  45.703 5.8e-11 ***
## label:political_social_con  1   0.35   0.350   0.719 0.39698
## Residuals                 349 169.71   0.486
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Plot stigma by label and political_social
summary_stigma_label_politic_aov <- data_main_politic_re %>%
  group_by(label, political_social_con) %>%
  summarise(
    stigma_mean = mean(stigma_avg),
    stigma_se = sd(stigma_avg)/sqrt(n())
  ) %>%
  mutate(label = ifelse(label == "the homeless", "PIC", "PC"))
```

```
## `summarise()` has grouped output by 'label'. You can override using the
## `.groups` argument.
```

```r
ggplot(summary_stigma_label_politic_aov, aes(x = political_social_con, y = stigma_mean, fill = label)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.7) +
  geom_errorbar(aes(ymin = stigma_mean - stigma_se,
                    ymax = stigma_mean + stigma_se),
                position = position_dodge(0.7),
                width = 0.2) +
  geom_text(aes(label = round(stigma_mean, 2), y = stigma_mean),
            vjust = -2,
            position = position_dodge(0.7)) +
  labs(title = "Stigma by Label and Political Orientation",
    x = "Political Orientation",
    y = "Average Stigma") +
  theme_minimal()
```



There are main effects of label and political_social on stigma. But there is no significant interaction effect between label and political_social on stigma.

Next Steps

literature: How does **stigma** perception of one group affect the **behavior** towards this group?

example questions (stigma) plot results by label stigma by political description of donation donation by label

Q: what should we do with German sample can be liberal should we change behavioral measure? If so, how about feild experiment? Sign up for newsletter?

Thesis

different labels exists stigma may be influenced by label. stigma may influecne donation behaviors.