Moral Judgment: TAPS Wave 10

Jennifer Lin

1/6/2020

Contents

Introduction

The American Panel Study (data accessible here: https://wc.wustl.edu/taps-data-archive) contains the Moral Foundations Questionnaire in its 20-item version. This was administered to the panel in March of 2012, during Wave 10 of the study. Here, I will use the 20-item questionnaire to create an analysis that replicates the work of Graham, Haidt and Nosek for the Moral Judgment subscale.

Before I begin, I load in the data (available for downlaod here) and relevant packages.

```
# Load packages
library(tidyverse)
library(psych)
library(ggplot2)
library(GGally)
library("ggpubr")
library("reshape2")
library(scales)
library(lsr)
```

Moral Foundations Questionnaire

Clean Data

In this section, I organize the variables that I will need for the graph and the linear model.

First, I remove participants who did not pass the manipulation check items.

```
taps = read.csv("~/Desktop/Working/Moral-Psychology/TAPS/TAPS10/taps10MFQ.csv",
    header = TRUE)
```

```
taps <- taps[!(taps$attchecka == "4"), ]
taps <- taps[!(taps$attchecka == "5"), ]

taps <- taps[!(taps$attcheckb == "1"), ]
taps <- taps[!(taps$attcheckb == "2"), ]
taps <- taps[!(taps$attcheckb == "3"), ]
```

Next, I create variables that represent aggregate scores on each of the moral foundations based on the Moral Judgment subscale.

```
### Harm ###
taps$compat <- taps$compat - 1
taps$hurt <- taps$hurt - 1
taps$Harm <- rowMeans(taps[, c("compat", "hurt")], na.rm = TRUE)

### Fairness ###
taps$treatf <- taps$treatf - 1
taps$justice <- taps$justice - 1
taps$Fairness <- rowMeans(taps[, c("treatf", "justice")], na.rm = TRUE)

### Ingroup ###
taps$proudc <- taps$proudc - 1
taps$loyalf <- taps$loyalf - 1
taps$Ingroup <- rowMeans(taps[, c("proudc", "loyalf")], na.rm = TRUE)

### Authority ###
taps$authc <- taps$authc - 1
taps$roles <- taps$roles - 1
taps$Authority <- rowMeans(taps[, c("authc", "roles")], na.rm = TRUE)

### Purity ###
taps$nodisgust <- taps$nodisgust - 1
taps$notnat <- taps$notnat - 1
taps$Purity <- rowMeans(taps[, c("disgust", "notnat")], na.rm = TRUE)
```

For the descriptive statistics line plot that I will create later, I recode a political ideology variable to reflect factor labels rather than numeric.

```
taps$ideology <- as.character(as.integer(taps$ideo7))



taps$ideology <- recode(taps$ideology, `1` = "Very Liberal")
```

```
taps$ideology <- recode(taps$ideology, `2` = "Liberal")
taps$ideology <- recode(taps$ideology, `3` = "Slightly Liberal")
taps$ideology <- recode(taps$ideology, `4` = "Moderate")
taps$ideology <- recode(taps$ideology, `5` = "Slightly Conservative")
taps$ideology <- recode(taps$ideology, `6` = "Conservative")
taps$ideology <- recode(taps$ideology, `7` = "Very Conservative")


# Rid implicit NAs for the ideology variable
library(forcats)
taps$ideology <- fct_explicit_na(taps$ideology, na_level = "NA")

# Establish factor order for graphing
taps$ideology <- as.factor(as.character(taps$ideology))
taps$ideology <- factor(taps$ideology, levels = c("Very Liberal",
    "Liberal", "Slightly Liberal", "Moderate", "Slightly Conservative",
    "Conservative", "Very Conservative"))

table(taps$ideology)
```

```
##
##          Very Liberal               Liberal      Slightly Liberal
##                    88                   208                   176
##              Moderate Slightly Conservative          Conservative
##                   274                   173                   253
##     Very Conservative
##                    79
```

Descriptive Statistics Plot

In this section, I will create a linegraph that displays the average score on each foundation as a function of the respondent's political ideology. To do this, I generate average scores by moral foundation.

```
Harm <- aggregate(Harm ~ ideology, taps, mean, na.rm = TRUE)
Fairness <- aggregate(Fairness ~ ideology, taps, mean, na.rm = TRUE)
Ingroup <- aggregate(Ingroup ~ ideology, taps, mean, na.rm = TRUE)
Authority <- aggregate(Authority ~ ideology, taps, mean, na.rm = TRUE)
Purity <- aggregate(Purity ~ ideology, taps, mean, na.rm = TRUE)
```

In order for `ggplot` to graph the data, the points need to be merged into one large data frame and reshaped into the proper data frame formation. I do this with the code below.

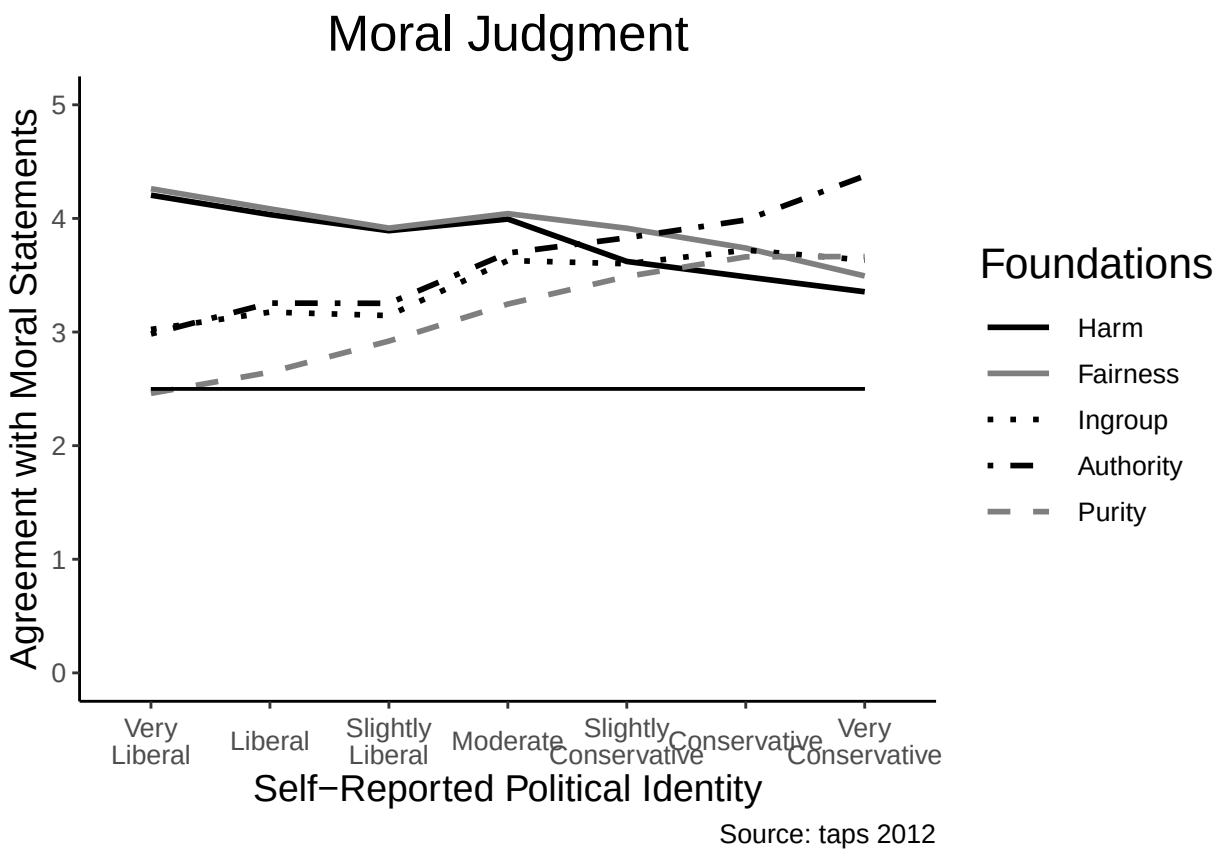
```
moral <- merge(Harm, Fairness, by.x = "ideology", by.y = "ideology",
    all.x = TRUE, all.y = TRUE)
```

```
moral <- merge(moral, Ingroup, by.x = "ideology", by.y = "ideology",
    all.x = TRUE, all.y = TRUE)
moral <- merge(moral, Authority, by.x = "ideology", by.y = "ideology",
    all.x = TRUE, all.y = TRUE)
moral <- merge(moral, Purity, by.x = "ideology", by.y = "ideology",
    all.x = TRUE, all.y = TRUE)

mfq <- reshape2::melt(moral, id.var = "ideology")
```

Now, I create the graph

```
ggplot(mfq, aes(x = ideology, y = value, group = variable)) +
    geom_line(aes(linetype = variable, color = variable), size = 1) +
    theme_classic() + geom_line(aes(y = 2.5)) + scale_linetype_manual("Foundations",
    breaks = c("Harm", "Fairness", "Ingroup", "Authority", "Purity"),
    values = c(Harm = "solid", Fairness = "solid", Ingroup = "dotted",
        Authority = "dotdash", Purity = "dashed")) + scale_color_manual("Foundations",
    breaks = c("Harm", "Fairness", "Ingroup", "Authority", "Purity"),
    values = c(Harm = "black", Fairness = "grey50", Ingroup = "black",
        Authority = "black", Purity = "grey50")) + ggtitle("Moral Judgment") +
    xlab("Self-Reported Political Identity") + ylab("Agreement with Moral Statements") +
    ylim(0, 5) + labs(caption = "Source: taps 2012") + theme(text = element_text(size =
    colour = "black"), axis.title = element_text(size = 14, colour = "black"),
    title = element_text(size = 16, colour = "black"), plot.caption = element_text(size
        color = "black"), axis.text.x = element_text(angle = 0,
        hjust = 0.5, vjust = 0.5), plot.title = element_text(hjust = 0.5),
    legend.key.width = unit(2, "line")) + scale_x_discrete(labels = wrap_format(10))
```



Cronbach's Alpha

I calculate the Cronbach's Alpha for each foundation on the Moral Judgment subscale

```
# Harm
Harm2 <- taps %>% select(c("compat", "hurt"))
psych::alpha(Harm2)
```

```
## Warning in matrix(unlist(drop.item), ncol = 10, byrow = TRUE): data length
## [16] is not a sub-multiple or multiple of the number of columns [10]
```

```
##
## Reliability analysis
## Call: psych::alpha(x = Harm2)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N   ase mean  sd median_r
##       0.35      0.35    0.21      0.21 0.54 0.027  3.8 1.1     0.21
##
##  lower alpha upper     95% confidence boundaries
## 0.3 0.35 0.4
##
##  Reliability if an item is dropped:
```

```
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## compat     0.211      0.21   0.045      0.21  NA       NA 0.211  0.21
## hurt       0.045      0.21      NA        NA  NA       NA 0.045  0.21
##
##  Item statistics
##           n raw.r std.r r.cor r.drop mean  sd
## compat 1319  0.75  0.78  0.36   0.21  3.6 1.3
## hurt   1327  0.80  0.78  0.36   0.21  4.0 1.4
##
## Non missing response frequency for each item
##           0    1    2    3    4    5 miss
## compat 0.04 0.04 0.06 0.24 0.33 0.29 0.45
## hurt   0.05 0.04 0.04 0.15 0.20 0.52 0.45
```

```
# Fairness
Fairness2 <- taps %>% select(c("treatf", "justice"))
psych::alpha(Fairness2)
```

```
## Warning in matrix(unlist(drop.item), ncol = 10, byrow = TRUE): data length
## [16] is not a sub-multiple or multiple of the number of columns [10]
```

```
##
## Reliability analysis
## Call: psych::alpha(x = Fairness2)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N   ase mean sd median_r
##       0.46      0.48    0.31      0.31 0.92 0.021  3.9  1     0.31
##
##  lower alpha upper     95% confidence boundaries
## 0.41 0.46 0.5
##
##  Reliability if an item is dropped:
##         raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## treatf      0.315      0.31   0.099      0.31  NA       NA 0.315  0.31
## justice     0.099      0.31      NA        NA  NA       NA 0.099  0.31
##
##  Item statistics
##            n raw.r std.r r.cor r.drop mean  sd
## treatf  1319  0.88  0.81  0.45   0.31  3.8 1.5
## justice 1326  0.73  0.81  0.45   0.31  4.1 1.0
##
## Non missing response frequency for each item
##            0    1    2    3    4    5 miss
## treatf  0.06 0.04 0.07 0.16 0.23 0.43 0.45
## justice 0.01 0.01 0.04 0.19 0.32 0.43 0.45
```

```r
# Ingroup
Ingroup2 <- taps %>% select(c("proudc", "loyalf"))
psych::alpha(Ingroup2)
```

```
## Warning in matrix(unlist(drop.item), ncol = 10, byrow = TRUE): data length
## [16] is not a sub-multiple or multiple of the number of columns [10]
```

```
##
## Reliability analysis
## Call: psych::alpha(x = Ingroup2)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean sd median_r
##       0.17      0.17   0.091     0.091 0.2 0.034  3.5  1    0.091
##
##  lower alpha upper     95% confidence boundaries
## 0.1 0.17 0.23
##
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## proudc    0.0911     0.091  0.0083     0.091  NA       NA 0.0911 0.091
## loyalf    0.0083     0.091      NA        NA  NA       NA 0.0083 0.091
##
##  Item statistics
##           n raw.r std.r r.cor r.drop mean  sd
## proudc 1320  0.70  0.74  0.22  0.091  3.9 1.3
## loyalf 1324  0.78  0.74  0.22  0.091  3.0 1.4
##
## Non missing response frequency for each item
##           0    1    2    3    4    5 miss
## proudc 0.03 0.04 0.06 0.16 0.33 0.38 0.45
## loyalf 0.06 0.11 0.14 0.26 0.26 0.17 0.45
```

```r
# Authority
Authority2 <- taps %>% select(c("authc", "roles"))
psych::alpha(Authority2)
```

```
## Warning in matrix(unlist(drop.item), ncol = 10, byrow = TRUE): data length
## [16] is not a sub-multiple or multiple of the number of columns [10]
```

```
##
## Reliability analysis
## Call: psych::alpha(x = Authority2)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N   ase mean sd median_r
##       0.29      0.31    0.18      0.18 0.45 0.026  3.6  1     0.18
##
```

```
##  lower alpha upper     95% confidence boundaries
## 0.23 0.29 0.34
##
##  Reliability if an item is dropped:
##       raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## authc     0.184      0.18   0.034      0.18  NA       NA 0.184  0.18
## roles     0.034      0.18      NA        NA  NA       NA 0.034  0.18
##
##  Item statistics
##          n raw.r std.r r.cor r.drop mean  sd
## authc 1318  0.64  0.77  0.33   0.18  4.3 1.0
## roles 1322  0.87  0.77  0.33   0.18  2.9 1.6
##
## Non missing response frequency for each item
##          0    1    2    3    4    5 miss
## authc 0.01 0.02 0.02 0.13 0.24 0.58 0.45
## roles 0.11 0.10 0.12 0.24 0.24 0.18 0.45
```

```r
# Purity
Purity2 <- taps %>% select(c("disgust", "notnat"))
psych::alpha(Purity2)
```

```
## Warning in matrix(unlist(drop.item), ncol = 10, byrow = TRUE): data length
## [16] is not a sub-multiple or multiple of the number of columns [10]
```

```
##
## Reliability analysis
## Call: psych::alpha(x = Purity2)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N  ase mean  sd median_r
##       0.49       0.5    0.33      0.33 0.99 0.02  3.2 1.2     0.33
##
##  lower alpha upper     95% confidence boundaries
## 0.45 0.49 0.53
##
##  Reliability if an item is dropped:
##         raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## disgust      0.33      0.33    0.11      0.33  NA       NA  0.33  0.33
## notnat       0.11      0.33      NA        NA  NA       NA  0.11  0.33
##
##  Item statistics
##            n raw.r std.r r.cor r.drop mean  sd
## disgust 1324  0.77  0.82  0.47   0.33  3.6 1.3
## notnat  1317  0.86  0.82  0.47   0.33  2.8 1.6
##
## Non missing response frequency for each item
```

```
##            0    1    2    3    4    5 miss
## disgust 0.00 0.08 0.14 0.22 0.27 0.29 0.45
## notnat  0.11 0.10 0.18 0.22 0.20 0.18 0.45
```

Repeated Measures GLM

To see if liberals and conservatives differ significantly on the individualizing versus binding foundations, the authors generated a repeated measures GLM to capture the relationship. Additionally, they tested to see if the relationship would be moderated by politics.

Here, I replicate the model using the TAPS dataset.

I generate a composite score for the individualizing and binding foundations.

```
# Individualizing and Binding Scores
taps$indiv <- rowMeans(taps[, c("compat", "hurt", "treatf", "justice")],
    na.rm = TRUE)
taps$bind <- rowMeans(taps[, c("proudc", "loyalf", "authc", "roles",
    "disgust", "notnat")], na.rm = TRUE)

taps$diffscore <- taps$indiv - taps$bind

# The results here generate the same mediation model score
# F(1, 1207) = 224.34 as Study 1
diff.model <- lm(diffscore ~ ideo7, data = taps)
summary(diff.model)
```

```
##
## Call:
## lm(formula = diffscore ~ ideo7, data = taps)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9579 -0.5785  0.0191  0.6254  2.9818
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.69916    0.07147   23.78   <2e-16 ***
## ideo7       -0.31031    0.01625  -19.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9876 on 1249 degrees of freedom
##   (1141 observations deleted due to missingness)
## Multiple R-squared:  0.226,  Adjusted R-squared:  0.2253
```

```
## F-statistic: 364.6 on 1 and 1249 DF,  p-value: < 2.2e-16
```

```
etaSquared(diff.model)
```

```
##          eta.sq eta.sq.part
## ideo7 0.2259563   0.2259563
```

The reported results are as follows

- Aggregate difference between Indivdualizing and binding foundation: $F(1, 1249) = 565.48$, p $< .001$
- Moderation by Politics: $F(1, 1249) = 364.6$, p $< .001$, $\eta^2 = .23$.