3. Analysis Plan

true

2023-06-12

```r
library(data.table)
library(magrittr)
library(kableExtra)
library(downloadthis)
library(ggplot2)
library(patchwork)
library(stargazer)
library(Rmisc)
```

```r
layout <- theme(panel.background = element_rect(fill = "white"),
                panel.grid = element_blank(),
                panel.grid.major.y = element_line(colour="gray", size=0.25),
                legend.key = element_rect(fill = "white"),
                axis.line.x.bottom = element_line(size = 0.25),
                axis.line.y.left = element_line(size = 0.25),
                axis.title.y = element_text(size = rel(0.75)),
                plot.margin = unit(c(0.25,0.25,0.25,0.25), "cm"),
                legend.title = element_blank()
                )
colors <- c("#F3B05C", "#1E4A75", "#65B5C0", "#AD5E21")
```

```r
base::load(file="../../../data/processed/rda/GMTV2017.rda")                       # read GMTV
base::load(file="../../../data/processed/rda/GMTV2017_R1.rda")                    # read GMTVFirstRound
base::load(file="../../../data/processed/rda/GMTV2017_COVS.rda")                  # read GMTVCovariates
base::load(file="../../../data/processed/rda/replication2021.rda")                # read replication
base::load(file="../../../data/processed/rda/replication2021_R1.rda")             # read replicationFirstRo
base::load(file="../../../data/processed/rda/replication2021_COVS.rda")           # read replicationCovaria
base::load(file="../../../data/processed/rda/replication2021_meta.rda")           # read meta data
base::load(file="../../../data/processed/rda/replication2021_timeSpent.rda")  # read time spent
```

```r
main <- rbindlist(list(replication, GMTV),
                  use.names = TRUE)

R1   <- rbindlist(list(replicationFirstRound, GMTVFirstRound),
                  use.names = TRUE)

covs <- rbindlist(list(replicationCovariates, GMTVCovariates),
                  use.names = TRUE,
                  fill = TRUE)
```

Key Takeaways

- We replicate a public goods experiment with dynamic interdependency and find similar results as Gächter et al. (2017).
- Absolute contributions increase over time.
- Just as in *static* public goods experiments, the share of endowments contributed decreases over time.
- The richest groups earn fifteen times more than the poorest groups.
- While there clearly is growth, groups do not realize the maximal potential efficiency and earn just over 8.2% of what is possible.
- Even though we conducted the experiment online and remotely, dropouts (or "attrition") are no concern.
- Relying on an inexperienced non-convenience sample, 95% of all participants stated that they understood the game.
- Participants make relatively fast decisions which makes longer games feasible in the future.[1]

Background

In an attempt to incorporate uncertainty to Gächter et al. (2017)'s dynamic public goods game (DPGG), I plan to run a series of remote online experiments using oTree (Chen, Schonger, and Wickens 2016). This experiment replicated Gächter et al.'s NOPUNISH 10-round treatment arm as close as possible (given the remote circumstances). The current demo version of the experiment can be found here. Click here to visit the corresponding Github repository.

This report is the third in a series of reports covering this project. It reads the data prepared in the previous reports and analyzes them. The whole replication project is registered in the AEA RCT Registry and the unique identifying number is: AEARCTR-0007902 (Berlemann, Roggenkamp, and Traub 2021).

Data

I'll refer to the data Gächter et al. (2017) provided as *GMTV* or *noPunish10* in what follows. GMTV conducted most of their sessions in late 2012. All of these sessions ran in Nottingham using a student sample. collected They 23 observations in their NOPUNISH 10 treatment arm. Our data, refered to as *replication* is more recent, gathered remotely in Hamburg using a different tech stack as well as a different sample.

We conducted a series a 4 sessions in between Thursday, July 01 to Friday, July 23 2021 and collected 29 observations (from 116 participants) in total. 35 additional participants could not be matched with other group members or failed to answer the comprehension questions. These participants are labeled as dropouts. None of them dropped out during the session such that attrition is no problem here.

```
meta %>% kable()
```

session.code

date

time

showups

dropouts

participants

observations

jyf8xd0s

2021-07-01

[1]Taken together with inefficient growth and the lack of dropouts, additional rounds are fairly cheap.

15:00

35

7

28

7

vggk2gh1

2021-07-03

13:00

20

8

12

3

8gi7c8xg

2021-07-09

13:00

21

9

12

3

d6jrsxnr

2021-07-23

14:00

75

11

64

16

Two of these sessions were special: The first (jyf8xd0s) as well as the last one (d6jrsxnr). The first session suffered technical problems such that the risk elicitation task was omitted. The last session (almost exclusively) relied on a student sample as our non-student sample was exhausted after the first three sessions. As a consequence, the last session was conducted with 59 students while all others were conducted without any students. I'll therefore create a boolean `student` variable.

```
main[, student := FALSE]
main[session.code == "d6jrsxnr" | treatment == "noPunish10",
     student := TRUE]

R1[, student := FALSE]
R1[session.code == "d6jrsxnr" | treatment == "noPunish10",
   student := TRUE]
```

All participants were recruited in by the University of Hamburg's WISO Research Lab using HROOT (Bock, Baetge, and Nicklisch 2014).

```
rm(list = c("replication", "GMTV", "replicationFirstRound", "GMTVFirstRound", "GMTVCovariates", "replica
```

Results

First Round

We start by discussing initial contributions which assume the full range between 0 and 20 as Table 1 illustrates. Both the median as well as the mean are about 10 tokens, that is, 50% of the initial endowment in both data sources. This is comparable to initial contributions in the standard game with partner matching.[2]

```
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

replication <- R1[treatment == "replication", ownContribution]
GMTV    <- R1[treatment == "noPunish10", ownContribution]


rows <- sapply(X = list(replication, GMTV), FUN = NROW) %>% max()
temp <- data.frame(replication = c(replication, rep(NA, rows - NROW(replication))),
                   GMTV = c(GMTV, rep(NA, rows - NROW(GMTV))))

stargazer(temp,
          summary.stat = c("mean", "median","sd", "max", "min", "n"),
          type = type,
          flip = TRUE,
          header=FALSE)
```

Table 1:

Statistic	replication	GMTV
Mean	10.017	10.685
Median	10	10
St. Dev.	6.340	5.881
Max	20	20
Min	0	0
N	116	92

```
rs1 <- wilcox.test(replication,
                   GMTV,
                   exact = FALSE)$p.value %>%
  round(digits = 4) %>%
  formatC(format = "f",
          digits = 4)
```

The two-sided rank sum test (comparing differences between data sources) yields a p-Value of 0.3926 for the mean contribution in first round of the game.

[2]See Figure 3B in Fehr and Gächter (2000), for instance.

```
ggplot(R1, aes(y=ownContribution, x=othersContribution/3)) +
  layout +
  geom_abline(intercept = 0, slope = 1, linetype="dashed", alpha = 0.66) +
  geom_point(color = colors) +
  scale_y_continuous(expand = c(0, 0), limits = c(-1, 25)) +
  scale_x_continuous(expand = c(0, 0), limits = c(-1, 25))
```

Provision of the public good

We proceed by further discussing contributions. The left panel in Figure 1 shows the average amount of tokens participants contributed over time. Contributions are clearly non-zero and are increasing over time in the replication treatment.[3] While contributions flatten in the replication, the GMTV data exhibit a drop in the last round. Note that increasing contributions over time imply that participants have increasing endowments over time. Hence, increasing contributions do not necessarily imply that participants contribute increasing shares of their endowments. The right panel in Figure 1 shows the share of overall endowments contributed over time. In the original data, participants contribute around 53% of their endowment in round 1. This amount steadily decreases. The replication exhibits a similar pattern with an initial average contributions of 50%. Both treatments resemble the results from the standard game: Just as in Fehr and Gächter (2000) contributions start at a level of around 50-60% of endowments and decrease to around 10% of endowments in round 10.

```
SUM <- main[,
            lapply(.SD, mean, na.rm = TRUE),
            by = c("round", "treatment"),
            .SDcols = "contribution"]

SUM[,
    sum := round(contribution/4)]

upperLimit <- SUM$contribution %>% max() %>% round() + 10

p1 <- ggplot(data = SUM,
             aes(x = round, y = contribution, fill = treatment, color = treatment, lty = treatment)) +
  layout +
  theme(legend.position="bottom") +
  geom_line(show.legend=FALSE) +
  geom_point() +
  scale_x_continuous(name="",  breaks = 1:15) +
  scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
  labs(y = "Average Amount of Tokens contributed") +
  scale_color_manual(values = colors) +
  theme(plot.margin = margin(0.25,1,0.25,0.25, "cm"))

rm(list = c("SUM"))
```

```
SHARE <- main[,
            lapply(.SD, mean, na.rm = TRUE),
            by = c("round", "treatment"),
            .SDcols = "share"]

# SHARE <- main[,
#             .(share = sum(contribution)/sum(endowment)),
#             by = c("round", "treatment")]
```

[3] Participants contribute about 10 tokens in the first round

```
upperLimit <- 1

p2 <- ggplot(data = SHARE,
             aes(x = round, y = share, fill = treatment, color = treatment, lty = treatment)) +
  layout +
  theme(legend.position="bottom") +
  geom_line(show.legend=FALSE) +
  geom_point() +
  scale_x_continuous(name="",  breaks = 1:15) +
  scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
  labs(y = "Share of Current Endowment contributed") +
  scale_color_manual(values = colors)

p1 + p2 + plot_layout(guides = "collect") & theme(legend.position = "bottom")
```

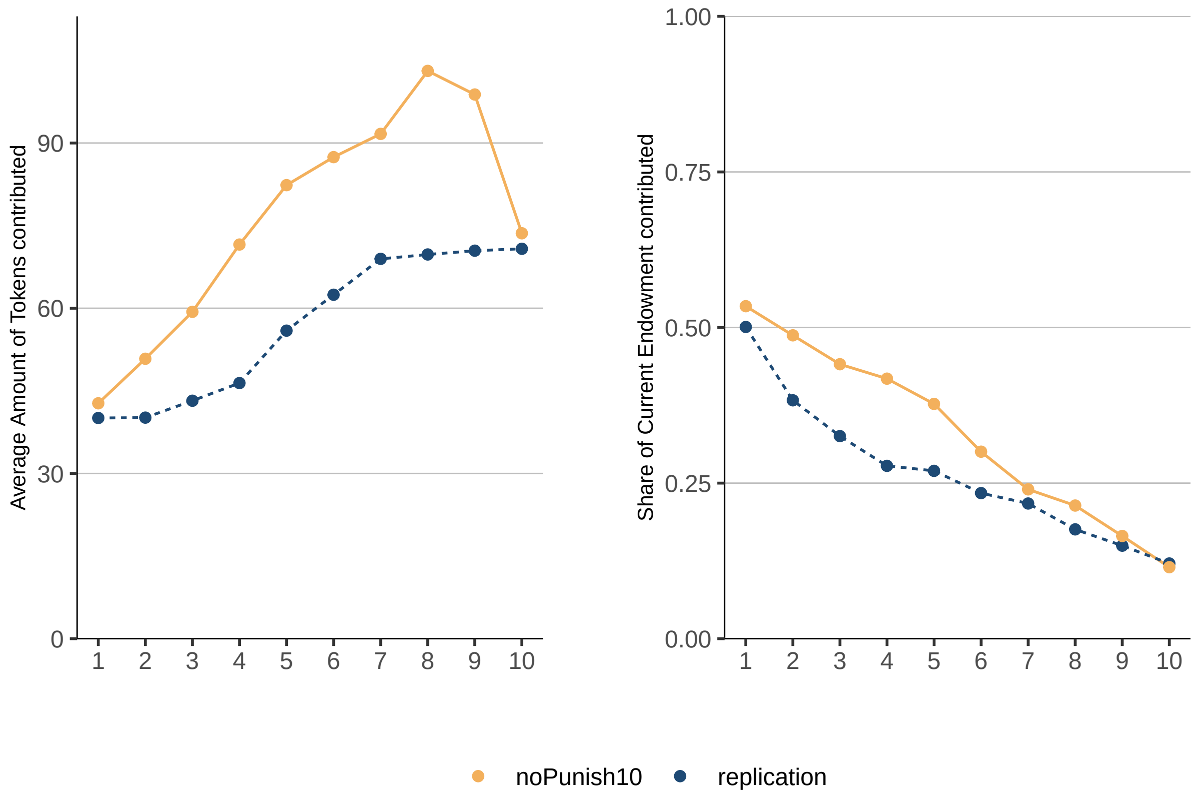

Figure 1: The average amount of tokens contributed over time in treatments.

```
rm(list = c("SHARE", "p1", "p2"))
```

Hm..?

The plot above puzzles me a little as the right panel visualizes the mean `share==contribution/endowment` by treatment and round. This yields a different result than the mean of the sum of contributions divided by the sum of endowments per treatment and round as the following table illustrates.

```
main[round == 10,
     .(contribution = mean(contribution),
       endowment = mean(endowment),
```

```
      # share = mean(share), # same thing as the following line
      share = mean(contribution/endowment),
      sumContBySumEndo = mean(sum(contribution)/sum(endowment))),
    by = c("treatment", "round")] %>%
  kable()
```

treatment

round

contribution

endowment

share

sumContBySumEndo

replication

10

70.7931

342.7586

0.1208084

0.2065392

noPunish10

10

73.6087

439.6957

0.1148712

0.1674083

```
SUM <- main[treatment == "replication",
            lapply(.SD, mean, na.rm = TRUE),
            by = c("round", "student"),
            .SDcols = "contribution"]

SUM[,
    sum := round(contribution/4)]

upperLimit <- SUM$contribution %>% max() %>% round() + 10

p1 <- ggplot(data = SUM,
             aes(x = round, y = contribution, fill = student, color = student, lty = student)) +
  layout +
  theme(legend.position="bottom") +
  geom_line(show.legend=FALSE) +
  geom_point() +
  scale_x_continuous(name="",  breaks = 1:15) +
  scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
  labs(y = "Average Amount of Tokens contributed") +
  scale_color_manual(values = colors) +
  theme(plot.margin = margin(0.25,1,0.25,0.25, "cm"))
```

```r
rm(list = c("SUM"))
```

Wealth Creation

Possibly of more interest are the implications contributions have for wealth generation and growth. To measure growth, we define a variable *stock* which sums the endowments of all participants in a given group at the end of the round (that is, after the contributions have been made, multiplied and redistributed). Gächter et al. (2017) refer to that variable as "wealth" so we will do the same in what follows. Before the start of round 1, wealth will be 80 in all groups by construction. The maximal wealth that can be reached in round 10 (if everyone contributes their entire endowment in each round) is approximately 4613 tokens or 230 Euro per group. Table 3 shows some summary statistics regarding wealth. Groups do achieve growth on average. While there is clearly growth, groups do not realize the maximal potential efficiency as the replication groups reach on average a level of 379 tokens out of 4613 maximally possible or 8.2%. As in the original data, there is large heterogeneity with the richest group reaching 1425 tokens whereas the poorest group ends up with 92 tokens.

```r
STOCK <- main[,
              lapply(.SD, mean, na.rm = TRUE),
              by = c("round", "treatment"),
              .SDcols = "stock"]

STOCKr <- main[rich == TRUE,
               lapply(.SD, mean, na.rm = TRUE),
               by = c("round", "treatment"),
               .SDcols = "stock"]

STOCKp <- main[rich == FALSE,
               lapply(.SD, mean, na.rm = TRUE),
               by = c("round", "treatment"),
               .SDcols = "stock"]

upperLimit <- STOCKr$stock %>% max() %>% round() + 20

p1 <- ggplot(data = STOCK,
       aes(x = round, y = stock, fill = treatment, color = treatment, lty = treatment)) +
          layout +
          theme(legend.position="bottom") +
          # geom_vline(xintercept = 10, alpha = 0.66) +
          geom_line(show.legend=FALSE) +
          geom_point() +
          scale_x_continuous(name="",  breaks = 1:10) +
          scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
          labs(y = "Wealth") +
          scale_color_manual(values = colors) +
          theme(plot.margin = margin(0.25,1,0.25,0.25, "cm"))

p2 <- ggplot(data = STOCKr,
       aes(x = round, y = stock, fill = treatment, color = treatment, lty = treatment)) +
          layout +
          theme(legend.position="bottom") +
          # geom_vline(xintercept = 10, alpha = 0.66) +
          geom_line(show.legend=FALSE) +
          geom_point() +
```

```
          scale_x_continuous(name="",  breaks = 1:10) +
          scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
          labs(y = "Wealth (Rich)") +
          scale_color_manual(values = colors)

p3 <- ggplot(data = STOCKp,
        aes(x = round, y = stock, fill = treatment, color = treatment, lty = treatment)) +
          layout +
          theme(legend.position="bottom") +
          # geom_vline(xintercept = 10, alpha = 0.66) +
          geom_line(show.legend=FALSE) +
          geom_point() +
          scale_x_continuous(name="",  breaks = 1:10) +
          scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
          labs(y = "Wealth (Poor)") +
          scale_color_manual(values = colors)

(p1 | (p2 / p3)) + plot_layout(guides = "collect") & theme(legend.position = "bottom")
```

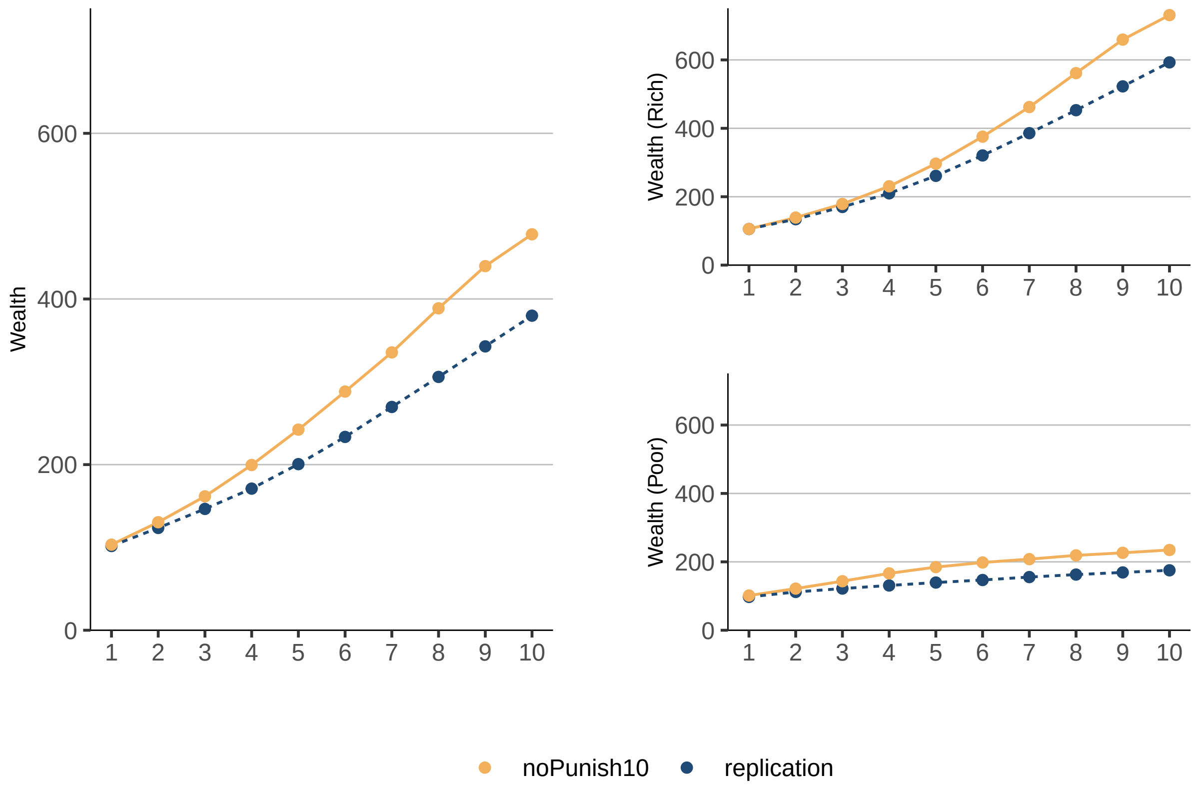

Figure 2: Average wealth over time across treatments.

```
rm(list = c("STOCK", "STOCKr", "STOCKp", "p1", "p2", "p3"))
```

Figure 2 shows the dynamics of wealth over time. The left panel focuses on all groups, the upper right panel on those with above median wealth after round 10 ("successful" groups) and the lower left panel on those with below median wealth after round 10 ("unsuccessful" groups). The average wealth is increasing across rounds and is substantially above 80 once round 10 was played as Table 2 illustrates.

```r
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

replication <- main[treatment == "replication" & round == 10, stock]
GMTV    <- main[treatment == "noPunish10" & round == 10, stock]


rows <- sapply(X = list(replication, GMTV), FUN = NROW) %>% max()
temp <- data.frame(replication = c(replication, rep(NA, rows - NROW(replication))),
                   GMTV = c(GMTV, rep(NA, rows - NROW(GMTV))))

stargazer(temp,
          summary.stat = c("mean", "median","sd", "max", "min", "n"),
          type = type,
          flip = TRUE,
          header=FALSE)
```

Table 2:

Statistic	replication	GMTV
Mean	379.828	478.087
Median	262	304
St. Dev.	336.059	393.575
Max	1,425	1,792
Min	92	161
N	29	23

```r
rs1 <- wilcox.test(replication,
                   GMTV,
                   exact = FALSE)$p.value %>%
  round(digits = 4) %>%
  formatC(format = "f",
          digits = 4)
```

The two-sided rank sum test (comparing differences between data sources) yields a p-Value of 0.1356 for the mean wealth after the last round of the game.

Wealth Differences between Data Sources

We next consider whether we were able to replicate GMTV's results with our data. Absolute contributions tended to be higher in Gächter et al. (2017) but end up at around the same level as in the replication due to a stark decline in contributions on the last round. In terms of shares contributed both data sources exhibit a similar pattern: they decline and do not stabilize. Even though the share of current endowments contributed in the last round is quite similar, the share declined a little faster in our data.

Our groups also tend to be poorer. Median wealth is higher in GMTV. This difference in mean ranks is not significant according to a two-sided ranksum test, however. To assess the statistical significance of differences in means, we run OLS regressions where we regress wealth on a treatment dummy for *Replication* (Table 3). These regressions show that differences in means are only significant for below median groups.

```r
# create subsets
main_all  <- main[round == 10]
main_poor <- main[round == 10 & rich == FALSE]
main_rich <- main[round == 10 & rich == TRUE]

# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

m1 <- lm(formula = stock ~ treatment, data = main_all)
m2 <- lm(formula = stock ~ treatment, data = main_poor)
m3 <- lm(formula = stock ~ treatment, data = main_rich)

stargazer(m1,
          m2,
          m3,

          column.labels = c("All", "Below median", "Above median"),
          model.numbers = FALSE,
          dep.var.labels = "Wealth",
          header=FALSE,
          covariate.labels = c("Replication"),

          type = type, digits = 2, omit.stat = c("adj.rsq", "f"), df = FALSE
          )
```

Table 3:

	Dependent variable:		
	Wealth		
	All	Below median	Above median
Replication	−98.26	−59.41***	−138.21
	(101.21)	(18.32)	(166.67)
Constant	478.09***	234.70***	731.00***
	(75.58)	(13.99)	(124.73)
Observations	52	24	25
R^2	0.02	0.32	0.03
Residual Std. Error	362.49	44.24	413.67
Note:			*p<0.1; **p<0.05; ***p<0.01

Inequality

In this subsection, we focus on the amount of inequality created endogenously in our setting. The smallest possible value the Gini coefficient takes is zero (if all four group members own one fourth of the wealth) and the largest possible value it takes is one (if one group member holds the entire wealth). Table 4 shows some summary statistics regarding the Gini coefficient.

The round 10 Gini coefficient ranges between 0.035 and 0.52 in our data with a median of 0.218.

```
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

replication <- main[treatment == "replication" & round == 10, gini]
GMTV    <- main[treatment == "noPunish10" & round == 10, gini]


rows <- sapply(X = list(replication, GMTV), FUN = NROW) %>% max()
temp <- data.frame(replication = c(replication, rep(NA, rows - NROW(replication))),
                   GMTV = c(GMTV, rep(NA, rows - NROW(GMTV))))

stargazer(temp,
          summary.stat = c("mean", "median","sd", "max", "min", "n"),
          type = type,
          flip = TRUE,
          header=FALSE)
```

Table 4:

Statistic	replication	GMTV
Mean	0.218	0.232
Median	0.218	0.233
St. Dev.	0.123	0.123
Max	0.520	0.423
Min	0.035	0.040
N	29	23

```
rs2 <- wilcox.test(replication,
                   GMTV,
                   exact = FALSE)$p.value %>%
  round(digits = 4) %>%
  formatC(format = "f",
          digits = 4)
```

The two-sided rank sum test yields a p-Value of 0.6059 for the mean `gini` during the last round of the game.

Figure 3 illustrates the dynamics of the Gini coefficient (at the end of each round) over time and shows that inequality increases slightly.

```
GINI <- main[,
             lapply(.SD, mean, na.rm = TRUE),
             by = c("round", "treatment"),
             .SDcols = "gini"]

GINIr <- main[rich == TRUE,
              lapply(.SD, mean, na.rm = TRUE),
              by = c("round", "treatment"),
              .SDcols = "gini"]
```

```r
GINIp <- main[rich == FALSE,
              lapply(.SD, mean, na.rm = TRUE),
              by = c("round", "treatment"),
              .SDcols = "gini"]

upperLimit <- GINI$gini %>% max() %>% round(digits = 1) + 0.15

p1 <- ggplot(data = GINI,
       aes(x = round, y = gini, fill = treatment, color = treatment, lty = treatment)) +
          layout +
          theme(legend.position="bottom") +
          # geom_vline(xintercept = 10, alpha = 0.66) +
          geom_line(show.legend=FALSE) +
          geom_point() +
          scale_x_continuous(name="",  breaks = 1:10) +
          scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
          labs(y = "Gini Coefficient") +
          scale_color_manual(values = colors) +
          theme(plot.margin = margin(0.25,1,0.25,0.25, "cm"))

p2 <- ggplot(data = GINIr,
       aes(x = round, y = gini, fill = treatment, color = treatment, lty = treatment)) +
          layout +
          theme(legend.position="bottom") +
          # geom_vline(xintercept = 10, alpha = 0.66) +
          geom_line(show.legend=FALSE) +
          geom_point() +
          scale_x_continuous(name="",  breaks = 1:10) +
          scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
          labs(y = "Gini (Rich)") +
          scale_color_manual(values = colors)

p3 <- ggplot(data = GINIp,
       aes(x = round, y = gini, fill = treatment, color = treatment, lty = treatment)) +
          layout +
          theme(legend.position="bottom") +
          # geom_vline(xintercept = 10, alpha = 0.66) +
          geom_line(show.legend=FALSE) +
          geom_point() +
          scale_x_continuous(name="",  breaks = 1:10) +
          scale_y_continuous(limits = c(0, upperLimit), expand = c(0, 0)) +
          labs(y = "Gini (Poor)") +
          scale_color_manual(values = colors)

(p1 | (p2 / p3)) + plot_layout(guides = "collect") & theme(legend.position = "bottom")
```

```r
rm(list = c("GINI", "GINIp", "GINIr", "p1", "p2", "p3"))
```

Inequality Differences between Data Sources

Once more, we consider whether we were able to replicate GMTV's results with our data.

The following table shows a simple OLS regression to illustrate differences in the 10th round's Gini coefficient between the replication's data and GMTV's data. Mean Gini coefficients are similar across data sources and

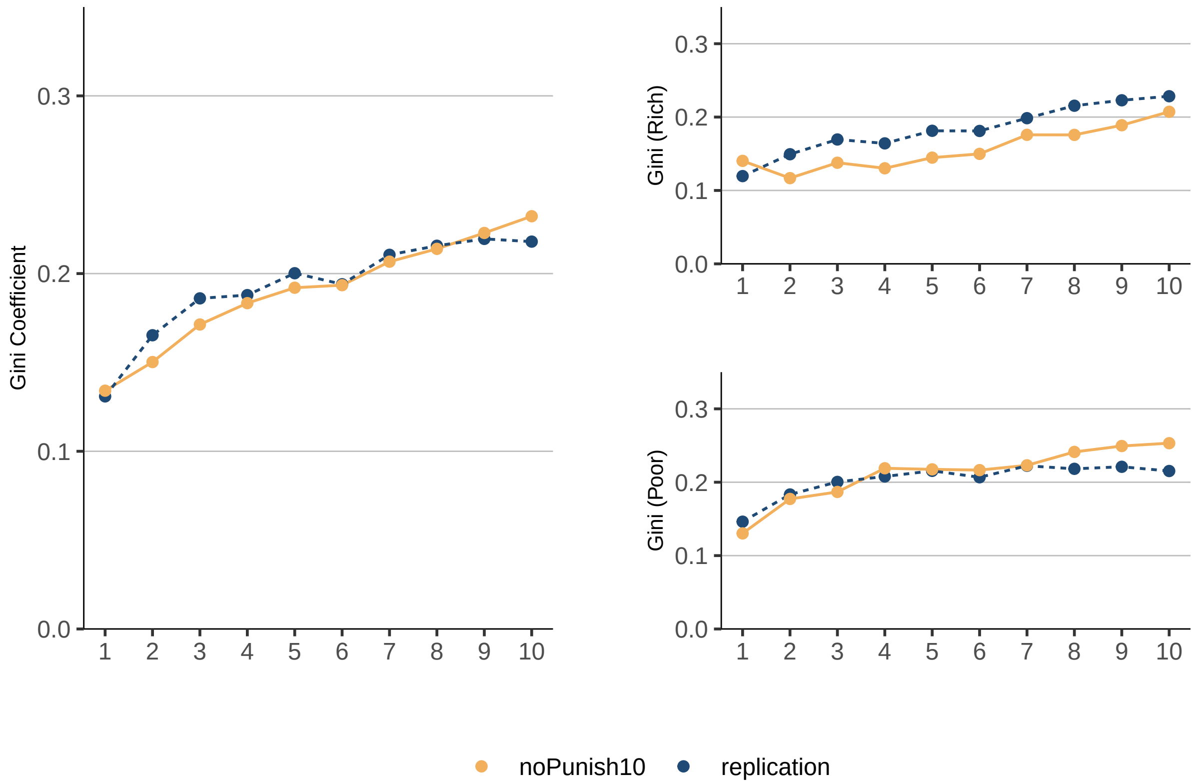

Figure 3: Average Gini coefficient over time across treatments.

there are no statistically significant differences in mean Gini coefficients.

```
# create subsets
main_all  <- main[round == 10]
main_poor <- main[round == 10 & rich == FALSE]
main_rich <- main[round == 10 & rich == TRUE]

# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

m1 <- lm(formula = gini ~ treatment, data = main_all)
m2 <- lm(formula = gini ~ treatment, data = main_poor)
m3 <- lm(formula = gini ~ treatment, data = main_rich)

stargazer(m1,
          m2,
          m3,

          column.labels = c("All", "Below median", "Above median"),
          model.numbers = FALSE,
          dep.var.labels = "Gini",
          header=FALSE,
```

```
         covariate.labels = c("Replication"),

         type = type, digits = 2, omit.stat = c("adj.rsq", "f"), df = FALSE
         )
```

Table 5:

	Dependent variable:		
	Gini		
	All	Below median	Above median
Replication	−0.01	−0.04	0.02
	(0.03)	(0.05)	(0.05)
Constant	0.23***	0.25***	0.21***
	(0.03)	(0.04)	(0.04)
Observations	52	24	25
R^2	0.003	0.02	0.01
Residual Std. Error	0.12	0.13	0.12
Note:			*p<0.1; **p<0.05; ***p<0.01

Time Spent

Participants spent approximately 24 minutes completing the experiment. Reading the instructions and answering the comprehension questions took the most time, that is, 12 minutes. The public goods game required 7 minutes of the participants' time.

```
N <- duration[, participant_code %>% unique() %>% length()]
plotDT <- duration[app_name == "dPGG" & page_name == "dPGG_Decision",
                   .(time_spent = time_spent %>% sum()),
                   by = c("session_code", "participant_code", "page_index", "page_name")]

plotDT[, round := seq(from = 1, to = 10), by = c("participant_code")]

upperLimit <- plotDT[, time_spent %>% mean(), by = c("round")] %>% max()



ggplot(data = summarySE(data = plotDT,
                        measurevar = "time_spent",
                        groupvars=c("round"),
                        na.rm = FALSE,
                        conf.interval = 0.95,
                        .drop = TRUE),
       mapping = aes(x = round, y = time_spent)) +
  layout +
  theme(legend.position="bottom") +
  geom_line(show.legend=FALSE, color = colors[2], lty=2) +
  geom_errorbar(aes(ymin=time_spent-ci, ymax=time_spent+ci), width=.25, alpha = 0.5, color = colors[2])
  geom_point(color = colors[2]) +
  scale_x_continuous(name="",  breaks = 1:10) +
```

```
  scale_y_continuous(limits = c(0, upperLimit + 10), expand = c(0, 0)) +
  labs(y = "Time Passed in Seconds", caption = "Bars indicate 95% confidence intervals.") +
  theme(plot.margin = margin(0.25,1,0.25,0.25, "cm"))
```

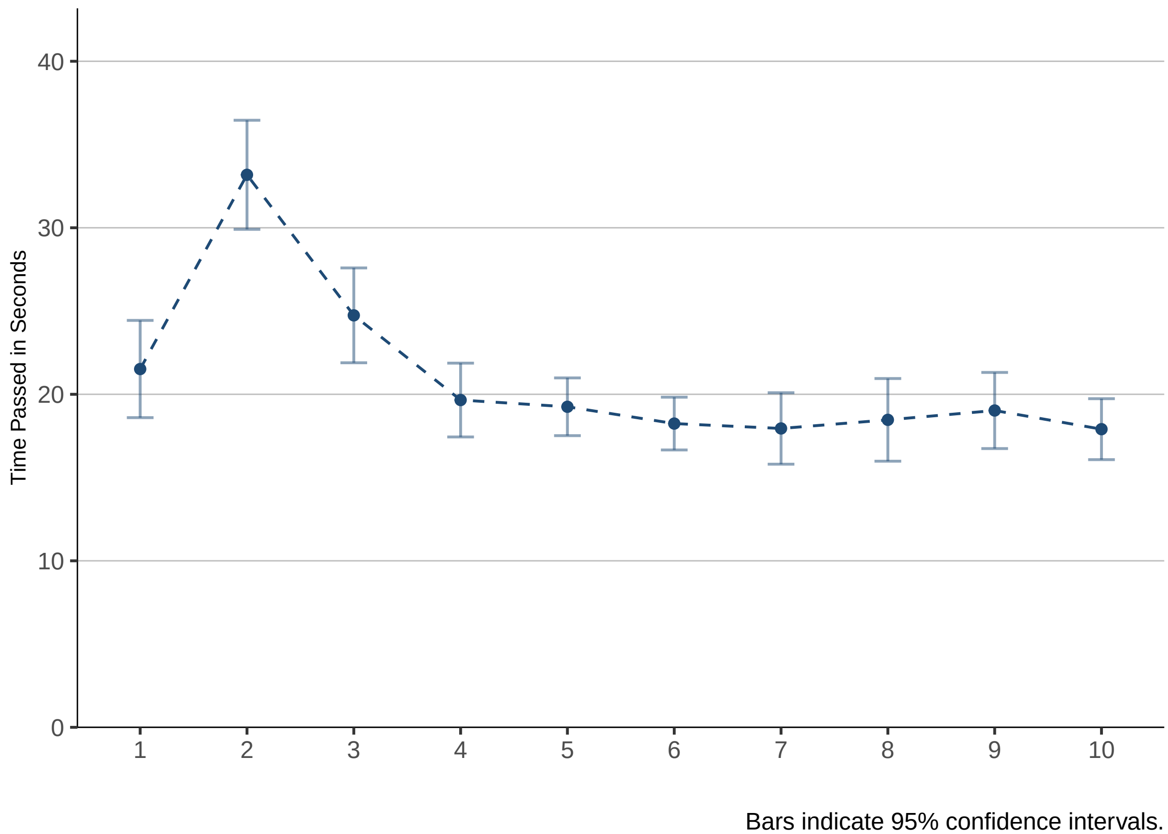

Figure 4: Average Time Spent for each Contribution per Round.

Figure 4 illustrates the time participants needed to make each of their contributions during the replication. One can see that the initial as well as most of the other contributions take about 20 seconds of time. Interestingly, the second contribution takes (on average) about 50% more time than the first one - presumably because this is the first time participants' learn their respective group members' actions. The third contribution is a little faster and all subsequent contributions stabilize at 19 seconds.

Given that no participant dropped out after answering the comprehension questions correctly and given that participants need less than 20 seconds to make their contributions, more than 10 rounds are feasible.

Sample Properties

```
covariates <- data.table::merge.data.table(x  = covs,
                                           y  = main[round == 10, .(groupID, stock, gini)],
                                           by = c("groupID"),
                                           allow.cartesian = TRUE)


covariates[,
           gender := gender %>% as.numeric()]
```

```r
# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

temp <- covariates[treatment == "replication",
                   .(gender, age, switching_row, education, donation, pq01, pq02, pq03, pq04,
                     pq05, pq06, pq07, pq08, pq09, pq10, pq11, pq12, pq13, pq14)]
stargazer(temp, type = type, flip = FALSE)
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Mon, Jun 12, 2023 - 11:18:35

Table 6:

Statistic	N	Mean	St. Dev.	Min	Max
gender	116	0.526	0.501	0	1
age	116	35.767	15.663	9	73
switching_row	78	6.154	2.151	1	12
education	116	6.241	0.966	3	8
donation	116	0.931	1.672	0.000	11.050
pq01	116	4.302	1.144	1	6
pq02	116	2.681	1.702	0	6
pq03	116	3.759	1.184	1	6
pq04	116	1.853	1.551	0	6
pq05	116	4.284	1.207	1	6
pq06	116	3.672	1.525	0	6
pq07	116	4.879	1.463	0	6
pq08	116	4.647	1.385	1	6
pq09	116	1.560	2.006	0	6
pq10	116	3.009	1.639	0	6
pq11	116	3.586	1.358	0	6
pq12	116	3.914	1.564	0	6
pq13	116	3.836	1.480	0	6
pq14	116	4.241	1.618	0	6

```r
# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

m1 <- lm(formula = gender ~ treatment, data = covs)
m2 <- lm(formula = age ~ treatment, data = covs)
m3 <- lm(formula = switching_row ~ treatment, data = covs)
m4 <- lm(formula = donation ~ treatment, data = covs)

stargazer(m1,
          m2,
          m3,
```

```
          m4,

          column.labels = c("female", "age", "risk", "donations"),
          model.numbers = FALSE,
          dep.var.labels.include = FALSE,
          header=FALSE,
          covariate.labels = c("Replication"),

          type = type, digits = 2, omit.stat = c("adj.rsq", "f"), df = FALSE
          )
```

Table 7:

	Dependent variable:			
	female	age	risk	donations
Replication	0.15**	3.04*	−0.28	0.25
	(0.07)	(1.65)	(0.27)	(0.23)
Constant	0.38***	32.73***	6.44***	0.68***
	(0.05)	(1.23)	(0.19)	(0.17)
Observations	208	208	165	208
R^2	0.02	0.02	0.01	0.01
Residual Std. Error	0.50	11.83	1.76	1.66
Note:				*p<0.1; **p<0.05; ***p<0.01

```
# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

m5 <- lm(formula = pq01 ~ treatment, data = covs)
m6 <- lm(formula = pq02 ~ treatment, data = covs)
m7 <- lm(formula = pq03 ~ treatment, data = covs)
m8 <- lm(formula = pq04 ~ treatment, data = covs)
m9 <- lm(formula = pq05 ~ treatment, data = covs)

stargazer(m5,
          m6,
          m7,
          m8,
          m9,

          column.labels = c("quick thinker", "easily offended", "very satisfied", "very dependent", "ge
          model.numbers = FALSE,
          dep.var.labels.include = FALSE,
          header=FALSE,
          covariate.labels = c("Replication"),

          type = type, digits = 2, omit.stat = c("adj.rsq", "f"), df = FALSE
```

```
          )
```

Table 8:

	Dependent variable:				
	quick thinker	easily offended	very satisfied	very dependent	generally happy
Replication	−0.77***	−1.03***	−1.23***	−0.94***	−1.36***
	(0.17)	(0.23)	(0.17)	(0.20)	(0.16)
Constant	5.08***	3.71***	4.99***	2.79***	5.64***
	(0.13)	(0.17)	(0.13)	(0.15)	(0.12)
Observations	208	208	208	208	208
R^2	0.09	0.09	0.21	0.09	0.26
Residual Std. Error	1.20	1.66	1.21	1.46	1.15
Note:					*p<0.1; **p<0.05; ***p<0.01

```
# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

m10 <- lm(formula = pq06 ~ treatment, data = covs)
m11 <- lm(formula = pq07 ~ treatment, data = covs)
m12 <- lm(formula = pq08 ~ treatment, data = covs)
m13 <- lm(formula = pq09 ~ treatment, data = covs)
m14 <- lm(formula = pq10 ~ treatment, data = covs)

stargazer(m10,
          m11,
          m12,
          m13,
          m14,

          column.labels = c("work important", "family important", "friends important", "religion importa
          model.numbers = FALSE,
          dep.var.labels.include = FALSE,
          header=FALSE,
          covariate.labels = c("Replication"),

          type = type, digits = 2, omit.stat = c("adj.rsq", "f"), df = FALSE
          )
```

```
# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}
```

Table 9:

	Dependent variable:				
	work important	family important	friends important	religion important	politics importar
Replication	−1.04***	−0.61***	−1.30***	−0.87***	−0.69***
	(0.20)	(0.21)	(0.18)	(0.26)	(0.22)
Constant	4.72***	5.49***	5.95***	2.43***	3.70***
	(0.15)	(0.16)	(0.13)	(0.20)	(0.16)
Observations	208	208	208	208	208
R^2	0.11	0.04	0.20	0.05	0.05
Residual Std. Error	1.46	1.51	1.28	1.88	1.58
Note:					*p<0.1; **p<0.05; ***p<0.0

```r
m15 <- lm(formula = pq11 ~ treatment, data = covs)
m16 <- lm(formula = pq12 ~ treatment, data = covs)
m17 <- lm(formula = pq13 ~ treatment, data = covs)
m18 <- lm(formula = pq14 ~ treatment, data = covs)
stargazer(m15,
          m16,
          m17,
          m18,

          column.labels = c("most people trusted", "hard work better", "government responsible", "income
          model.numbers = FALSE,
          dep.var.labels.include = FALSE,
          header=FALSE,
          covariate.labels = c("Replication"),

          type = type, digits = 2, omit.stat = c("adj.rsq", "f"), df = FALSE
          )
```

Table 10:

	Dependent variable:			
	most people trusted	hard work better	government responsible	incomes equal
Replication	−0.28	−1.52***	−0.48**	0.42*
	(0.19)	(0.21)	(0.20)	(0.23)
Constant	3.87***	5.43***	4.32***	3.83***
	(0.14)	(0.16)	(0.15)	(0.17)
Observations	208	208	208	208
R^2	0.01	0.20	0.03	0.02
Residual Std. Error	1.36	1.52	1.42	1.63
Note:				*p<0.1; **p<0.05; ***p<0.01

```r
# create table
if(knitr::is_html_output()){
  type <- "html"
} else {
  type = "latex"
}

olsStock1 <- lm(formula = stock ~ gender + age +
                  switching_row +
                  pq01 + pq02 + pq03 + pq04 +
                  pq05 + pq06 + pq07 +pq08 + pq09 + pq10 + pq11 + pq12 + pq13 + pq14,
                data = covariates[treatment == "replication"])
olsStock2  <- lm(formula = stock ~ gender + age + switching_row,
                data = covariates[treatment == "replication"])
olsGINI1  <- lm(formula = gini ~ gender + age +
                  switching_row +
                  pq01 + pq02 + pq03 + pq04 +
                  pq05 + pq06 + pq07 + pq08 + pq09 + pq10 + pq11 + pq12 + pq13 + pq14,
                data = covariates[treatment == "replication"])
olsGINI2  <- lm(formula = gini ~ gender + age + switching_row,
                data = covariates[treatment == "replication"])

stargazer(olsStock1, olsStock2,
          olsGINI1, olsGINI2,

          se = list(coef(summary(olsStock1, cluster = c("groupID")))[, 2],
                    coef(summary(olsStock2, cluster = c("groupID")))[, 2],
                    coef(summary(olsGINI1,  cluster = c("groupID")))[, 2],
                    coef(summary(olsGINI2,  cluster = c("groupID")))[, 2]),

          column.labels   = c("Wealth", "Gini"),
          dep.var.labels.include = FALSE,
          column.separate = c(2, 2),
          model.numbers = FALSE,
          header=FALSE,
          covariate.labels = c("female", "age",
                               "risk",
                               "quick thinker",
                               "easily offended",
                               "very satisfied",
                               "very dependent",
                               "generally happy",
                               "work important",
                               "family important",
                               "friends important",
                               "religion important",
                               "politics important",
                               "most people trusted",
                               "hard work better",
                               "government responsible",
                               "incomes equal"),
          notes = "Robust standard errors in parentheses, clustered by group ID.",
```

```
    type = type, digits = 2, omit.stat = c("adj.rsq", "f"), df = FALSE
    )
```


Berlemann, Michael, Hauke Roggenkamp, and Stefan Traub. 2021. "Replication: Growth and Inequality in Public Good Provision (No-Punish-10) by Gächter Et Al. (2017)." *AEA RCT Registry*. https://doi.org/https://doi.org/10.1257/rct.7902-2.0.

Bock, Olaf, Ingmar Baetge, and Andreas Nicklisch. 2014. "Hroot: Hamburg Registration and Organization Online Tool." *European Economic Review* 71: 117–20. https://doi.org/https://doi.org/10.1016/j.euroecorev.2014.07.003.

Chen, Daniel L., Martin Schonger, and Chris Wickens. 2016. "oTree-an Open-Source Platform for Laboratory, Online, and Field Experiments." *Journal of Behavioral and Experimental Finance* 9: 88–97. https://doi.org/10.1016/j.jbef.2015.12.001.

Fehr, Ernst, and Simon Gächter. 2000. "Cooperation and Punishment in Public Goods Experiments." *American Economic Review* 90 (4): 980–94. https://doi.org/10.1257/aer.90.4.980.

Gächter, Simon, Friederike Mengel, Elias Tsakas, and Alexander Vostroknutov. 2017. "Growth and Inequality in Public Good Provision." *Journal of Public Economics* 150: 1–13. https://doi.org/10.1016/j.jpubeco.2017.03.002.

Table 11:

	Dependent variable:			
	Wealth		Gini	
female	−30.63	−65.02	0.02	0.02
	(88.07)	(83.80)	(0.03)	(0.03)
age	7.05*	9.00**	−0.001	−0.001
	(3.82)	(3.60)	(0.001)	(0.001)
risk	27.21	10.98	−0.01	−0.01
	(22.65)	(19.45)	(0.01)	(0.01)
quick thinker	−55.19		0.02	
	(45.52)		(0.02)	
easily offended	−26.52		0.01	
	(31.48)		(0.01)	
very satisfied	−2.62		0.001	
	(42.60)		(0.02)	
very dependent	−22.41		0.0003	
	(30.47)		(0.01)	
generally happy	70.06		0.004	
	(42.99)		(0.02)	
work important	−64.01		−0.01	
	(39.13)		(0.02)	
family important	11.99		−0.01	
	(38.05)		(0.01)	
friends important	−24.05		−0.004	
	(37.94)		(0.01)	
religion important	−20.48		0.02*	
	(26.15)		(0.01)	
politics important	86.94***		−0.01	
	(30.79)		(0.01)	
most people trusted	−0.49		−0.001	
	(31.94)		(0.01)	
hard work better	35.38		−0.03*	
	(34.46)		(0.01)	
government responsible	−47.38		−0.03*	
	(39.54)		(0.02)	
incomes equal	−9.13		0.01	
	(33.79)		(0.01)	
Constant	278.16	100.12	0.39***	0.28***
	(377.35)	(180.67)	(0.14)	(0.07)
Observations	78	78	78	78