DDuell_followMajority_runFile.R

dominik

2020-08-20

```
################################################################################
#  File-Name:   DDuell_followMajority_runFile.r
#  Date:        20/05/18
#  Author:      Dominik Duell
################################################################################
pkgs <- c('ggplot2','dplyr','tidyverse','broom','magrittr','coin','clusrank',
          'summarytools','estimatr','car','conflicted','rsample','Hmisc','grid',
          'gridExtra','ggpubr','ggforce','xtable','lme4','stargazer')
for (pkg in pkgs) library(pkg,character.only=T)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
## -- Attaching packages --------------------------------------------------------------------- tidyverse 1.3
```

```
## v tibble  3.0.1     v purrr   0.3.4
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------------------------ tidyverse_conflicts
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
## Loading required package: survival
```

```
##
## Attaching package: 'clusrank'
```

```
## The following object is masked from 'package:survival':
```

```
##
##     cluster
```

```
## Registered S3 method overwritten by 'pryr':
##   method      from
##   print.bytes Rcpp
```

```
## For best results, restart R session and update pander using devtools:: or remotes::install_github('ra
```

```
##
## Attaching package: 'summarytools'
```

```
## The following object is masked from 'package:tibble':
##
##     view
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## Loading required package: lattice
```

```
## Loading required package: Formula
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Registered S3 methods overwritten by 'lme4':
##   method                          from
##   cooks.distance.influence.merMod car
##   influence.merMod                car
##   dfbeta.influence.merMod         car
##   dfbetas.influence.merMod        car
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
conflict_prefer("filter","dplyr")
```

```
## [conflicted] Will prefer dplyr::filter over any other package
```

```
conflict_prefer("recode","dplyr")
```

```
## [conflicted] Will prefer dplyr::recode over any other package
```

```r
conflict_prefer("lag","dplyr")
```

```
## [conflicted] Will prefer dplyr::lag over any other package
```

```r
options(warn=-1)
##############################################################################
##############################################################################
# Get and prepare data, reorder factor levels as desired, create subsetted data
# to shorten code further down
##############################################################################
data <- read.csv('DDuell_followMajority_data.csv') %>%
  mutate(voteWinner = recode(voteWinner,`1`='P wins',`2`='R wins'),
    RWins=ifelse(voteWinner=='R wins',1,0),
    treatment=factor(treatment,levels=c('No appeal','Group appeal','Income appeal',
      'Poor MJ - No appeal','Poor MJ - Group appeal','Poor MJ - Income appeal',
      'Rich MI - Group appeal')),
    groupHeterogeneity=factor(groupHeterogeneity,levels=c('Low heterogeneity',
      'Medium heterogeneity','High heterogeneity')),
    groupHeterogeneity.binary=factor(ifelse(groupHeterogeneity=='Low heterogeneity',
      'Low','Medium or high')),
    majorityGroupRich=factor(factor(ifelse(majorityGroup=='MI','MI',
      ifelse(majorityGroup=='MJ'&rich==1,'MJ, rich','MJ, poor')))))

# Main treatments only: No appeal, group appeal, and income appeal treatment
data_richMajority <- data %>%
  filter(treatment=='No appeal'|treatment=='Group appeal'|
           treatment=='Income appeal') %>% droplevels()

# Main treatments and majority group MJ only, adding indicator variables for
# coordination mechanism
data_coordinationMechanism <- data_richMajority %>%
  filter(majorityGroup=='MJ'&groupHeterogeneity!='Low heterogeneity') %>%
  group_by(groupTreatment) %>% summarise(m.voteR=mean(voteR,na.rm=T)) %>%
  left_join(data_richMajority %>% filter(majorityGroup=='MJ'&
    groupHeterogeneity!='Low heterogeneity')) %>%
  mutate(
    m.voteR.noAppeal=ifelse(treatment=='No appeal',
      mean(data$voteR[data$treatment=='No appeal'&data$majorityGroup=='MJ'&
      data$groupHeterogeneity!='Low heterogeneity']),
      ifelse(treatment=='Group appeal',
             mean(data$voteR[data$treatment=='Group appeal'&
                               data$majorityGroup=='MJ'&
                               data$groupHeterogeneity!='Low heterogeneity']),
             mean(data$voteR[data$treatment=='Income appeal'&
                               data$majorityGroup=='MJ'&
                               data$groupHeterogeneity!='Low heterogeneity'])),
    propensity=factor(ifelse(m.voteR>m.voteR.noAppeal,'Propensity to coordinate on R',
      'Propensity to coordinate on P')),
    propensityR=ifelse(propensity=='Propensity to coordinate on R',1,0),
    propensityVsNoAppeal=factor(ifelse(treatment!='No appeal'&
      propensity=='Propensity to coordinate on R',
      'Propensity to coordinate on R',ifelse(treatment!='No appeal'&
      propensity=='Propensity to coordinate on P',
      'Propensity to coordinate on P','No appeal'))))
```

```
## Joining, by = "groupTreatment"
```

```
# Main treatments only, aggregate variables to society-round observations, add
# strategy profile variables
data_by_society <- data_richMajority %>%
  group_by(groupTreatment,majorityGroup,period,voteR) %>%
  summarise(count = n()) %>%
  spread(voteR,count) %>%
  mutate(countVoteR=ifelse(is.na(`1`)==F,`1`,0)) %>%
  select(-c(`0`,`1`)) %>%
  spread(majorityGroup,countVoteR) %>%
  mutate(countVoteRMJ = MJ,
    countVoteRMI = MI,
    countVotePMJ = 3-MJ,
    countVotePMI = 2-MI) %>%
  select(-c(MJ,MI))

strategy_profiles <- c('P wins, all vote P','P wins, MJ or MI split',
                       'R wins, MJ or MI split',
                       'R wins, MJ votes R and MI votes P')

# Main treatments only, merge individual-round and society-round-level data s
# set, add equilibrium indicator variables
data_richMajority_equilibria <- left_join(data_richMajority,data_by_society) %>%
  mutate(
    equilibrium = factor(
      ifelse(countVoteRMI==0&countVoteRMJ==0,'(P,P,P;P,P)',
      ifelse(countVoteRMI==0&countVoteRMJ==3,'(R,R,R;P,P)','No equilibrium')),
      levels=c('No equilibrium','(R,R,R;P,P)','(P,P,P;P,P)')),
    eqVoteWinner = factor(
      ifelse(equilibrium=='(P,P,P;P,P)','P wins, all vote P',
      ifelse(equilibrium=='No equilibrium'&
               voteWinner=='P wins','P wins, MJ or MI split',
      ifelse(equilibrium=='No equilibrium'&
               voteWinner=='R wins','R wins, MJ or MI split',
      ifelse(equilibrium=='(R,R,R;P,P)','R wins, MJ votes R and MI votes P',NA)))),
      levels=strategy_profiles),
    whoGetsI=
      ifelse((voteWinner=='R wins'&countVoteRMJ>countVoteRMI)|(voteWinner=='P wins'&
        countVotePMJ>countVotePMI),'MJ wins I',
      ifelse((voteWinner=='R wins'&countVoteRMJ<countVoteRMI)|(voteWinner=='P wins'&
        countVotePMJ<countVotePMI),'MI wins I',
      'MJ and MI split I')))
```

```
## Joining, by = c("period", "groupTreatment")
```

```
data_exitSurvey <- read.csv('DDuell_followMajority_exitSurvey.csv') %>%
  mutate(
    sid=treatment*1000+session*100+subject,
    treatment=recode(factor(treatment),`1`='No appeal',`2`='Group appeal',
      `3`='Income appeal',`4`='Poor MJ - No appeal',`5`='Poor MJ - Group appeal',
      `6`='Poor MJ - Income appeal',`7`='Rich MI - Group appeal'))

################################################################################
################################################################################
```

```r
# 4 Experimental design
################################################################################
# Table 1:  Number of societies, subjects, and subject-round observations for
# appeal and group heterogeneity treatment conditions. Every session includes
# 8 rounds of the low group heterogeneity, 28 of the medium group heterogeneity,
# and 4 rounds of the high group heterogeneity treatment.
################################################################################
out <- data_richMajority %>%
  group_by(treatment,groupHeterogeneity) %>%
  summarise(societies=length(unique(groupTreatment)),
    subjects=length(unique(sid)),
    rounds=length(sid)) %>%
  spread(groupHeterogeneity,rounds) %>% ungroup() %>%
  add_row(treatment='Total',societies=sum(.$societies),
          subjects=sum(.$subjects),
          `Low heterogeneity`=sum(.$`Low heterogeneity`),
          `Medium heterogeneity`=sum(.$`Medium heterogeneity`),
          `High heterogeneity`=sum(.$`High heterogeneity`))
out
```

```
## # A tibble: 4 x 6
##   treatment societies subjects `Low heterogene~ `Medium heterog~
##   <chr>         <int>    <int>            <int>            <int>
## 1 No appeal        14       70              560             1960
## 2 Group ap~        16       80              640             2240
## 3 Income a~         8       40              320             1120
## 4 Total            38      190             1520             5320
## # ... with 1 more variable: `High heterogeneity` <int>
```

```r
print(xtable(out,type='latex'),file='table1.tex')

################################################################################
# 5 Results
################################################################################
# 5.1 Equilibrium predictions
################################################################################
# Statistics in the text (p.18/19): Relative
# frequency of equilibria by group heterogeneity
# Tests and society-level bootstrap of difference
# in frequency of strategy profiles and equilibria
################################################
results.equ.play <- data_richMajority_equilibria %>%
  group_by(treatment,groupHeterogeneity,eqVoteWinner) %>%
  summarise(n=n()) %>% mutate(N=max(cumsum(n)),prop=n/N)

results.equ.play %>% group_by(groupHeterogeneity,eqVoteWinner) %>%
  summarise(N=sum(N),n=sum(n)) %>%
  mutate(N=ifelse(groupHeterogeneity=='High heterogeneity',760,N),
      # fix at N=760 for High heterogeneity to account in the computation
      # for no observations of any profile with R winning for high heterogeneity
         prop=n/N)
```

```
## # A tibble: 12 x 5
## # Groups:   groupHeterogeneity [3]
##    groupHeterogeneity   eqVoteWinner                          N     n   prop
```

```
##    <fct>                <fct>                              <dbl> <int>  <dbl>
##  1 Low heterogeneity    P wins, all vote P                  1520    35 0.0230
##  2 Low heterogeneity    P wins, MJ or MI split              1520   460 0.303
##  3 Low heterogeneity    R wins, MJ or MI split              1520   340 0.224
##  4 Low heterogeneity    R wins, MJ votes R and MI votes P   1520   685 0.451
##  5 Medium heterogeneity P wins, all vote P                  5320   990 0.186
##  6 Medium heterogeneity P wins, MJ or MI split              5320  2705 0.508
##  7 Medium heterogeneity R wins, MJ or MI split              5320   790 0.148
##  8 Medium heterogeneity R wins, MJ votes R and MI votes P   5320   835 0.157
##  9 High heterogeneity   P wins, all vote P                   760   175 0.230
## 10 High heterogeneity   P wins, MJ or MI split               760   495 0.651
## 11 High heterogeneity   R wins, MJ or MI split               760    45 0.0592
## 12 High heterogeneity   R wins, MJ votes R and MI votes P    760    45 0.0592
```

```
data_richMajority_equilibria %>% group_by(groupHeterogeneity,whoGetsI) %>%
  summarise(n=n()) %>% mutate(N=max(cumsum(n)),prop=n/N)
```

```
## # A tibble: 9 x 5
## # Groups:   groupHeterogeneity [3]
##   groupHeterogeneity   whoGetsI              n     N  prop
##   <fct>                <chr>             <int> <int> <dbl>
## 1 Low heterogeneity    MI wins I           240  1520 0.158
## 2 Low heterogeneity    MJ and MI split I   185  1520 0.122
## 3 Low heterogeneity    MJ wins I          1095  1520 0.720
## 4 Medium heterogeneity MI wins I           995  5320 0.187
## 5 Medium heterogeneity MJ and MI split I  1270  5320 0.239
## 6 Medium heterogeneity MJ wins I          3055  5320 0.574
## 7 High heterogeneity   MI wins I           160   760 0.211
## 8 High heterogeneity   MJ and MI split I   255   760 0.336
## 9 High heterogeneity   MJ wins I           345   760 0.454
```

```
# Hypothesis tests
# Difference in frequency tests (society-level)
out <- data_richMajority_equilibria %>%
  mutate(
    REq=ifelse(equilibrium=='(R,R,R;P,P)',1,0),
    PEq=ifelse(equilibrium=='(P,P,P;P,P)',1,0)
    ) %>%
  group_by(treatment,groupTreatment,groupHeterogeneity) %>%
  summarise(REq=mean(REq),PEq=mean(PEq),RWins=mean(RWins))
out.noHigh <- out %>% filter(groupHeterogeneity!='High heterogeneity')
out.noMedium <- out %>% filter(groupHeterogeneity!='Medium heterogeneity')

# R-equilibrium
out.noHigh %$% wilcox.test(REq~groupHeterogeneity,paired=T,alternative='g')
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  REq by groupHeterogeneity
## V = 543.5, p-value = 1.353e-06
## alternative hypothesis: true location shift is greater than 0
```

```
out.noMedium %$% wilcox.test(REq~groupHeterogeneity,paired=T,alternative='g')
```

```
##
```

```
##  Wilcoxon signed rank test with continuity correction
##
## data:  REq by groupHeterogeneity
## V = 465, p-value = 8.494e-07
## alternative hypothesis: true location shift is greater than 0
```

```
# Dealing with zero differences and ties, two-sided test
out.noHigh %$% wilcoxsign_test(REq~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.5858, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
out.noMedium %$% wilcoxsign_test(REq~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.6541, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
# P-equilibrium
out.noHigh %$% wilcox.test(PEq~groupHeterogeneity,paired=T,alternative='l')
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  PEq by groupHeterogeneity
## V = 5, p-value = 5.111e-06
## alternative hypothesis: true location shift is less than 0
```

```
out.noMedium %$% wilcox.test(PEq~groupHeterogeneity,paired=T,alternative='l')
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  PEq by groupHeterogeneity
## V = 1, p-value = 3.945e-05
## alternative hypothesis: true location shift is less than 0
```

```
out.noHigh %$% wilcoxsign_test(PEq~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.67, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
out.noMedium %$% wilcoxsign_test(PEq~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.6835, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
# R wins election
out.noHigh %$% wilcox.test(RWins~groupHeterogeneity,paired=T,alternative='g')
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  RWins by groupHeterogeneity
## V = 523, p-value = 6.684e-07
## alternative hypothesis: true location shift is greater than 0
```

```
out.noMedium %$% wilcox.test(RWins~groupHeterogeneity,paired=T,alternative='g')
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  RWins by groupHeterogeneity
## V = 561, p-value = 2.595e-07
## alternative hypothesis: true location shift is greater than 0
```

```
out.noHigh %$% wilcoxsign_test(RWins~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.4216, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
out.noMedium %$% wilcoxsign_test(RWins~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.4697, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
# Bootstrapping inference
# Take bootstrap sample from set of societies and re-compute frequency of strategy
# profiles
D <- data_richMajority %>% nest(-groupTreatment)
set.seed(01010)
boots <- bootstraps(D,times=1000)
boot.results.society.level <-
  map(boots$splits, ~as_tibble(.) %>% unnest %>%
    group_by(groupHeterogeneity,groupTreatment,majorityGroup,period,voteR) %>%
    summarise(count = n()) %>% spread(voteR,count) %>%
```

```
      mutate(countVoteR=ifelse(is.na(`1`)==F,`1`,0)) %>%
    select(-c(`0`,`1`)) %>% spread(majorityGroup,countVoteR) %>%
      mutate(countVoteRMJ=MJ,countVoteRMI=MI,
        countVotePMJ=3-MJ,countVotePMI=2-MI,
        RWins=ifelse(countVoteRMJ+countVoteRMI>=3,1,0)) %>%
    select(-c(MJ,MI)) %>%
    mutate(
      REq=ifelse(countVoteRMI==0&countVoteRMJ==3,1,0),
      PEq=ifelse(countVoteRMI==0&countVoteRMJ==0,1,0)
      ) %>% group_by(groupHeterogeneity) %>%
    summarise(REq=mean(REq,na.rm=T),PEq=mean(PEq,na.rm=T),
              RWins=mean(RWins,na.rm=T)) %>%
      mutate(diff.REq.y=ifelse(groupHeterogeneity=='Medium heterogeneity',
                          REq-lag(REq),REq-lag(REq,2)),
        diff.PEq.y=ifelse(groupHeterogeneity=='Medium heterogeneity',
                          PEq-lag(PEq),PEq-lag(PEq,2)),
        diff.RWins.y=ifelse(groupHeterogeneity=='Medium heterogeneity',
                            RWins-lag(RWins),RWins-lag(RWins,2)),)) %>%
  bind_rows(.id = 'boots') %>% filter(!is.na(diff.REq.y)) %>%
  group_by(groupHeterogeneity) %>%
    summarise(
      diff.REq=mean(diff.REq.y),REq.lower=quantile(diff.REq.y,0.025),
      REq.upper=quantile(diff.REq.y,0.975),
      diff.PEq=mean(diff.PEq.y),PEq.lower=quantile(diff.PEq.y,0.025),
      PEq.upper=quantile(diff.PEq.y,0.975),
      diff.RWins=mean(diff.RWins.y),RWins.lower=quantile(diff.RWins.y,0.025),
      RWins.upper=quantile(diff.RWins.y,0.975)
    )
boot.results.society.level
```

```
## # A tibble: 2 x 10
##   groupHeterogene~ diff.REq REq.lower REq.upper diff.PEq PEq.lower PEq.upper
##   <fct>               <dbl>     <dbl>     <dbl>    <dbl>     <dbl>     <dbl>
## 1 Medium heteroge~   -0.158    -0.237   -0.0675    0.162     0.115     0.206
## 2 High heterogene~   -0.204    -0.313   -0.0870    0.206     0.135     0.271
## # ... with 3 more variables: diff.RWins <dbl>, RWins.lower <dbl>,
## #   RWins.upper <dbl>
```

```
# Is the mixed strategy equilibrium being played?
out <- data_richMajority_equilibria %>%
  filter(groupHeterogeneity=='Low heterogeneity'&
           equilibrium=='No equilibrium') %>%
  group_by(groupTreatment,sid,income) %>% summarise(voteR=mean(voteR)) %>%
  mutate(income=factor(income))

out %>% group_by(income) %>% summarise(voteR=mean(voteR))
```

```
## # A tibble: 5 x 2
##   income voteR
##   <chr>  <dbl>
## 1 27     0.319
## 2 38     0.287
## 3 44     0.671
## 4 62     0.748
```

```
## 5 73     0.726
```

```
m <- lm_robust(voteR~income-1,data=out,cluster=groupTreatment)
summary(m)
```

```
##
## Call:
## lm_robust(formula = voteR ~ income - 1, data = out, clusters = groupTreatment)
##
## Standard error type:  CR2
##
## Coefficients:
##          Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper    DF
## income27   0.3193    0.04745   6.728 9.228e-08   0.2229   0.4157 34.50
## income38   0.2866    0.04958   5.781 1.578e-06   0.1859   0.3873 34.50
## income44   0.6707    0.05253  12.769 2.607e-14   0.5639   0.7776 32.96
## income62   0.7479    0.04272  17.508 2.648e-18   0.6610   0.8348 33.11
## income73   0.7260    0.05180  14.016 1.925e-15   0.6206   0.8314 32.96
##
## Multiple R-squared:  0.7096 ,    Adjusted R-squared:  0.7056
## F-statistic:  69.8 on 5 and 37 DF,  p-value: < 2.2e-16
```

```
linearHypothesis(m,diag(5),c(0,0,.38,.62,.74),vcov.m=cluster.vcov(m,out$groupTreatment))
```

```
## Linear hypothesis test
##
## Hypothesis:
## income27 = 0
## income38 = 0
## income44 = 0.38
## income62 = 0.62
## income73 = 0.74
##
## Model 1: restricted model
## Model 2: voteR ~ income - 1
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    373
## 2    368  5 127.28  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linearHypothesis(m,'income27 = 0',vcov.m=cluster.vcov(m,out$groupTreatment))
```

```
## Linear hypothesis test
##
## Hypothesis:
## income27 = 0
##
## Model 1: restricted model
## Model 2: voteR ~ income - 1
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    369
## 2    368  1 45.264  1.722e-11 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linearHypothesis(m,'income38 = 0',vcov.m=cluster.vcov(m,out$groupTreatment))
```

```
## Linear hypothesis test
##
## Hypothesis:
## income38 = 0
##
## Model 1: restricted model
## Model 2: voteR ~ income - 1
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    369
## 2    368  1 33.416  7.441e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linearHypothesis(m,'income44 = .38',vcov.m=cluster.vcov(m,out$groupTreatment))
```

```
## Linear hypothesis test
##
## Hypothesis:
## income44 = 0.38
##
## Model 1: restricted model
## Model 2: voteR ~ income - 1
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    369
## 2    368  1 30.636  3.113e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linearHypothesis(m,'income62 = .62',vcov.m=cluster.vcov(m,out$groupTreatment))
```

```
## Linear hypothesis test
##
## Hypothesis:
## income62 = 0.62
##
## Model 1: restricted model
## Model 2: voteR ~ income - 1
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    369
## 2    368  1 8.9632   0.002755 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linearHypothesis(m,'income73 = .74',vcov.m=cluster.vcov(m,out$groupTreatment))
```

```
## Linear hypothesis test
##
## Hypothesis:
## income73 = 0.74
##
```

```
## Model 1: restricted model
## Model 2: voteR ~ income - 1
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    369
## 2    368  1 0.0729     0.7872
```

```
################################################
# Figure 2: Distribution of relative frequency of
# strategy profiles by group heterogeneity and
# appeal treatments
################################################
pdf('barPlotStacked_shareEquPlay_byTreatmentAndIncDistrAndVoteWinner.pdf',
    height=4)
results.equ.play %>%
  mutate(eqVoteWinner=recode(eqVoteWinner,
    'P wins, all vote P'='Redistributive cand. P wins, all vote P',
    'P wins, MJ or MI split'='Redistributive cand. P wins, MJ or MI split',
    'R wins, MJ or MI split'='Wealth-preserving cand. R wins, MJ or MI split',
    'R wins, MJ votes R and MI votes P'=
      'Wealth-preserving cand. R wins, MJ votes R and MI votes P')) %>%
  ggplot(aes(y=prop,x=treatment,color=eqVoteWinner,fill=eqVoteWinner)) +
  geom_col() +
  facet_grid(~groupHeterogeneity) +
  scale_color_manual(values=c('blue','lightblue','indianred1','red')) +
  scale_fill_manual(values=c('blue','lightblue','indianred1','red')) +
  guides(fill=guide_legend(nrow=2),color=guide_legend(nrow=2)) +
  labs(y='Cumulative relative frequency',x='') +
  theme_bw() +
  theme(
    legend.position='bottom',
    legend.title=element_blank(),
    legend.spacing.x = unit(.25, 'cm'),
    axis.text.x = element_text(angle=45,hjust=1)
  )
dev.off()
```

```
## pdf
##   2
```

```
#############################################################################
# 5.2 Appeal treatment effects
#############################################################################
# Statistics in the text (p.20): Relative frequency
# of equilibria by group heterogeneity and appeal
# treatment. Tests of difference in frequency of
# equilibria and vote share of R as well as
# subject-level bootstrap of difference in vote
# share of R
################################################
results.equ.play %>%
  filter(groupHeterogeneity=='Low heterogeneity'&
           eqVoteWinner=='R wins, MJ votes R and MI votes P')
```

```
## # A tibble: 3 x 6
## # Groups:   treatment, groupHeterogeneity [3]
```

```
##   treatment     groupHeterogenei~ eqVoteWinner                    n     N  prop
##   <fct>         <fct>             <fct>                       <int> <int> <dbl>
## 1 No appeal     Low heterogeneity R wins, MJ votes R and MI v~   235   560 0.420
## 2 Group appeal  Low heterogeneity R wins, MJ votes R and MI v~   330   640 0.516
## 3 Income appeal Low heterogeneity R wins, MJ votes R and MI v~   120   320 0.375
```

```
# Hypothesis tests
# Difference in frequency of equilibria, R winning candidate (society-level)
# Low heterogeneity
out <- data_richMajority_equilibria %>%
  filter(groupHeterogeneity=='Low heterogeneity') %>%
  mutate(REq=ifelse(equilibrium=='(R,R,R;P,P)',1,0),
    PEq=ifelse(equilibrium=='(P,P,P;P,P)',1,0)) %>%
  group_by(groupTreatment,groupHeterogeneity,treatment) %>%
  summarise(REq=mean(REq),PEq=mean(PEq),RWins=mean(RWins))
# R-equilibrium
# No vs group appeal (p-value):
out %>% filter(treatment!='Income appeal') %$%
  wilcox.test(REq~treatment,alternative='l')$p.value
```

```
## [1] 0.1518346
```

```
# No vs income appeal (p-value):
out %>% filter(treatment!='Group appeal') %$%
  wilcox.test(REq~treatment,alternative='l')$p.value
```

```
## [1] 0.6853942
```

```
# P-equilibrium
# No vs group appeal (p-value):
out %>% filter(treatment!='Income appeal') %$%
  wilcox.test(PEq~treatment,alternative='g')$p.value
```

```
## [1] 0.5
```

```
# No vs income appeal (p-value):
out %>% filter(treatment!='Group appeal') %$%
  wilcox.test(PEq~treatment,alternative='l')$p.value
```

```
## [1] 0.3407834
```

```
# R wins
# No vs group appeal (p-value):
out %>% filter(treatment!='Income appeal') %$%
  wilcox.test(RWins~treatment,alternative='l')$p.value
```

```
## [1] 0.2897831
```

```
# No vs income appeal (p-value):
out %>% filter(treatment!='Group appeal') %$%
  wilcox.test(RWins~treatment,alternative='g')$p.value
```

```
## [1] 0.2213477
```

```
# Medium and high heterogeneity
out <- data_richMajority_equilibria %>%
  filter(groupHeterogeneity.binary!='Low') %>%
  mutate(RRRPP=ifelse(equilibrium=='(R,R,R;P,P)',1,0),
    PEq=ifelse(equilibrium=='(P,P,P;P,P)',1,0)) %>%
```

```
  group_by(groupTreatment,groupHeterogeneity.binary,treatment) %>%
  summarise(RRRPP=mean(RRRPP),PEq=mean(PEq),RWins=mean(RWins))

# (R,R,R;P,P)-profile
out %>% group_by(treatment) %>% summarise(RRRPP=mean(RRRPP))
```

```
## # A tibble: 3 x 2
##   treatment      RRRPP
##   <fct>          <dbl>
## 1 No appeal     0.0893
## 2 Group appeal  0.178
## 3 Income appeal 0.176
```

```
# No vs group appeal (p-value):
out %>% filter(treatment!='Income appeal') %$%
  wilcox.test(RRRPP~treatment,alternative='l')$p.value
```

```
## [1] 0.04149996
```

```
# No vs income appeal (p-value):
out %>% filter(treatment!='Group appeal') %$%
  wilcox.test(RRRPP~treatment)$p.value
```

```
## [1] 0.1080308
```

```
# Difference in vote share of R (subject-level)
# Income appeal, poor vs rich for majority group (difference and p-value)
data_richMajority %>%
  filter(treatment=='Income appeal'&majorityGroupRich!='MI') %>%
  group_by(groupHeterogeneity,majorityGroupRich,sid) %>%
  summarise(voteR=mean(voteR)) %>%
  group_by(groupHeterogeneity) %>%
  summarise(poor=t.test(voteR~majorityGroupRich)$estimate[1],
            rich=t.test(voteR~majorityGroupRich)$estimate[2],
            p.value=clusWilcox.test(
              voteR,group=majorityGroupRich,cluster=sid)$p.value,
            paired.p.value=pvalue(wilcoxsign_test(voteR~majorityGroupRich,
                                                  distribution='exact'))) %>%
  mutate(diff=rich-poor)
```

```
## # A tibble: 3 x 6
##   groupHeterogeneity     poor  rich p.value paired.p.value  diff
##   <fct>                 <dbl> <dbl>   <dbl> <pvalue>       <dbl>
## 1 Low heterogeneity    0.642  0.756 0.241   1.136868e-13   0.114
## 2 Medium heterogeneity 0.394  0.631 0.00101 7.105427e-15   0.237
## 3 High heterogeneity   0.0476 0.326 0.00539 5.684342e-14   0.279
```

```
# No vs group appeal and no vs income appeal for minority group (difference
# and p-value)
data_richMajority %>%
  filter(groupHeterogeneity!='Low heterogeneity'&majorityGroupRich=='MI'&
           treatment!='Income appeal') %>%
  group_by(treatment,groupHeterogeneity,sid) %>%
  summarise(voteR=mean(voteR)) %>%
  group_by(groupHeterogeneity) %>%
  summarise(noAppeal=t.test(voteR~treatment)$estimate[1],
```

```
            groupAppeal=t.test(voteR~treatment)$estimate[2],
            p.value=wilcox.test(voteR~treatment,alternative='g')$p.value) %>%
  mutate(diff=groupAppeal-noAppeal)
```

```
## # A tibble: 2 x 5
##   groupHeterogeneity   noAppeal groupAppeal p.value    diff
##   <fct>                   <dbl>       <dbl>   <dbl>   <dbl>
## 1 Medium heterogeneity    0.181       0.102  0.0482  -0.0796
## 2 High heterogeneity      0.196       0.0312 0.00245 -0.165
```

```
data_richMajority %>%
  filter(groupHeterogeneity!='Low heterogeneity'&majorityGroupRich=='MI'&
           treatment!='Group appeal') %>%
  group_by(treatment,groupHeterogeneity,sid) %>%
  summarise(voteR=mean(voteR)) %>%
  group_by(groupHeterogeneity) %>%
  summarise(noAppeal=t.test(voteR~treatment)$estimate[1],
            groupAppeal=t.test(voteR~treatment)$estimate[2],
            p.value=wilcox.test(voteR~treatment)$p.value) %>%
  mutate(diff=groupAppeal-noAppeal)
```

```
## # A tibble: 2 x 5
##   groupHeterogeneity   noAppeal groupAppeal p.value     diff
##   <fct>                   <dbl>       <dbl>   <dbl>    <dbl>
## 1 Medium heterogeneity    0.181       0.176   0.587  -0.00478
## 2 High heterogeneity      0.196       0.0469  0.0833 -0.150
```

```
# Bootstrapping inference
# Take bootstrap sample from set of subjects and re-compute vote share of
# candidate R
D <- data_richMajority %>% nest(-sid)
set.seed(01010)
boots <- bootstraps(D,times=1000)
results.vote.share.subject.clustered <-
  map(boots$splits, ~as_tibble(.) %>% unnest %>%
    group_by(treatment,groupHeterogeneity,majorityGroupRich) %>%
    summarise(mean=mean(voteR,na.rm=T))) %>%
  bind_rows(.id = 'boots') %>%
  group_by(treatment,groupHeterogeneity,majorityGroupRich) %>%
  summarise(y=mean(mean),lower=quantile(mean,0.025),upper=quantile(mean,0.975))
results.vote.share.subject.clustered
```

```
## # A tibble: 27 x 6
## # Groups:   treatment, groupHeterogeneity [9]
##    treatment    groupHeterogeneity   majorityGroupRich     y  lower upper
##    <fct>        <fct>                <fct>             <dbl>  <dbl> <dbl>
##  1 No appeal    Low heterogeneity    MI                0.189 0.107  0.280
##  2 No appeal    Low heterogeneity    MJ, poor          0.813 0.717  0.893
##  3 No appeal    Low heterogeneity    MJ, rich          0.834 0.758  0.906
##  4 No appeal    Medium heterogeneity MI                0.183 0.121  0.249
##  5 No appeal    Medium heterogeneity MJ, poor          0.322 0.211  0.432
##  6 No appeal    Medium heterogeneity MJ, rich          0.498 0.389  0.590
##  7 No appeal    High heterogeneity   MI                0.195 0.0952 0.313
##  8 No appeal    High heterogeneity   MJ, poor          0.217 0.0926 0.385
##  9 No appeal    High heterogeneity   MJ, rich          0.474 0.323  0.621
```

```
## 10 Group appeal Low heterogeneity    MI                0.134 0.0846 0.189
## # ... with 17 more rows
```

```
################################################
# Figure 3: Vote share of candidate R by group
# heterogeneity and appeal treatments for majority
# MJ and minority MI.
################################################
pdf('barPlot_meanAndCIVoteR_byTreatmentAndMajorityGroupAndIncDistr.pdf',
    height=4)
results.vote.share.subject.clustered %>%
  ggplot(aes(y=y,ymin=lower,ymax=upper,x=treatment,color=majorityGroupRich,
             fill=majorityGroupRich)) +
  geom_bar(stat='identity',position=position_dodge(width=.8)) +
  geom_errorbar(position=position_dodge(width=.8),color='lightgray',width=.2) +
  facet_grid(~groupHeterogeneity) +
  scale_color_manual(values=c('gray','black','black')) +
  scale_fill_manual(values=c('gray','black','white')) +
  labs(y='Vote share of candidate R',x='') +
  theme_bw() +
  theme(legend.position=c(.92,.82),
    legend.title=element_blank(),
    legend.spacing.x = unit(.1, 'cm'),
    axis.text.x = element_text(angle=45,hjust=1))
dev.off()
```

```
## pdf
##   2
```

```
# Efficiency (Footnote 27, p.20), difference in payoff no vs group appeal
# (p-values):
data_richMajority_equilibria %>%
  mutate(
    decision.payoff=ifelse(RWins==1,income,income*.5+25),
    identity.payoff=ifelse(majorityGroup=='MJ'&whoGetsI=='MJ wins I',10,
                           ifelse(whoGetsI=='MJ and MI split I',5,0)),
    payoff=decision.payoff+identity.payoff,
    identity.payoff.R = ifelse(RWins==1,identity.payoff,NA),
    identity.payoff.P = ifelse(RWins==0,identity.payoff,NA),
    payoff.R = ifelse(RWins==1,payoff,NA),
    payoff.P = ifelse(RWins==0,payoff,NA)
  ) %>% filter(treatment!='Income appeal') %>% droplevels() %>%
  group_by(groupHeterogeneity) %>%
  summarise_at(vars(payoff,identity.payoff),
               ~t.test.cluster(.,sid,treatment)[20,1])
```

```
## # A tibble: 3 x 3
##   groupHeterogeneity   payoff identity.payoff
##   <fct>                 <dbl>           <dbl>
## 1 Low heterogeneity     0.940           0.765
## 2 Medium heterogeneity  0.651           0.217
## 3 High heterogeneity    0.872           0.380
```

```
#############################################################################
# 5.3 Coordination mechanism
#############################################################################
```

```
# Statistics in the text (p.22-25): Propensity of
# MJ to coordinate on R or P and test for
# coordination mechanism
################################################
# Percentage of societies with a propensity to R/P (p.22):
data_coordinationMechanism %>% group_by(treatment,propensity) %>%
  summarise(n=n()) %>% mutate(N=max(cumsum(n)),prop=n/N)
```

```
## # A tibble: 6 x 5
## # Groups:   treatment [3]
##   treatment     propensity                          n     N  prop
##   <fct>         <fct>                           <int> <int> <dbl>
## 1 No appeal     Propensity to coordinate on P   768  1344 0.571
## 2 No appeal     Propensity to coordinate on R   576  1344 0.429
## 3 Group appeal  Propensity to coordinate on P   672  1536 0.438
## 4 Group appeal  Propensity to coordinate on R   864  1536 0.562
## 5 Income appeal Propensity to coordinate on P   384   768 0.5
## 6 Income appeal Propensity to coordinate on R   384   768 0.5
```

```
# Vote share of R in societies with a propensity to R vs P by treatment (p.22):
data_coordinationMechanism %>% group_by(treatment) %>%
  summarise(
      P=t.test(voteR~propensity)$estimate[1],
      R=t.test(voteR~propensity)$estimate[2],
      p.value=clusWilcox.test(voteR,group=as.numeric(propensity),
                              cluster=groupTreatment)$p.value) %>%
  mutate(diff=P-R)
```

```
## # A tibble: 3 x 5
##   treatment         P     R  p.value   diff
##   <fct>         <dbl> <dbl>    <dbl>  <dbl>
## 1 No appeal     0.232 0.698 0.00211  -0.466
## 2 Group appeal  0.216 0.720 0.000721 -0.504
## 3 Income appeal 0.266 0.766 0.0142   -0.5
```

```
# Vote share of R, no vs group appeal by income for MJ with propensity to
# coordinate on R (p.22):
data_coordinationMechanism %>% group_by(income) %>%
  filter(treatment!='Income appeal'&
           propensityVsNoAppeal!='Propensity to coordinate on P') %>%
  summarise(
    noAppeal=t.test(voteR~treatment)$estimate[1],
    groupAppeal=t.test(voteR~treatment)$estimate[2],
    p.value=clusWilcox.test(voteR,group=as.numeric(treatment),
                            cluster=groupTreatment)$p.value) %>%
  mutate(diff=groupAppeal-noAppeal)
```

```
## # A tibble: 7 x 5
##   income noAppeal groupAppeal p.value  diff
##    <int>    <dbl>       <dbl>   <dbl> <dbl>
## 1     10    0.214       0.444 0.0826  0.230
## 2     22    0.296       0.556 0.0587  0.260
## 3     27    0.342       0.643 0.0113  0.301
## 4     56    0.417       0.704 0.0345  0.287
## 5     62    0.526       0.810 0.0156  0.284
```

```
## 6     73    0.528       0.841 0.00944 0.313
## 7     90    0.5         0.75  0.104   0.25
```

```
# Vote share of R for poorest member if MJ (p.23):
data_richMajority %>% filter(treatment=='Group appeal'&
                               majorityGroup=='MJ'&rich==0) %>%
  group_by(groupHeterogeneity) %>% summarise(voteR=mean(voteR))
```

```
## # A tibble: 3 x 2
##   groupHeterogeneity   voteR
##   <fct>                <dbl>
## 1 Low heterogeneity    0.805
## 2 Medium heterogeneity 0.402
## 3 High heterogeneity   0.266
```

```
# Low vs medium group heterogeneity (p-value):
data_richMajority %>% filter(treatment=='Group appeal'&
                               majorityGroup=='MJ'&rich==0) %>%
  group_by(groupHeterogeneity,sid) %>% summarise(voteR=mean(voteR)) %>%
  filter(groupHeterogeneity!='High heterogeneity') %$%
  wilcoxsign_test(voteR~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.4339, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
# Low vs high group heterogeneity (p-value):
data_richMajority %>% filter(treatment=='Group appeal'&
                               majorityGroup=='MJ'&rich==0) %>%
  group_by(groupHeterogeneity,sid) %>% summarise(voteR=mean(voteR)) %>%
  filter(groupHeterogeneity!='High heterogeneity') %$%
  wilcoxsign_test(voteR~groupHeterogeneity,distribution='exact')
```

```
##
##  Exact Wilcoxon-Pratt Signed-Rank Test
##
## data:  y by x (pos, neg)
##   stratified by block
## Z = -7.4339, p-value < 2.2e-16
## alternative hypothesis: true mu is not equal to 0
```

```
################################################
# Figure 4: Vote share of candidate $R$ by round
# for the majority $MJ$
################################################
D <- data_coordinationMechanism  %>% nest(-sid)
set.seed(01010)
boots <- bootstraps(D,times=1000)
results <- map(boots$splits, ~as_tibble(.) %>% unnest %>%
                 group_by(treatment,propensity,rich,period) %>%
                 summarise(mean=mean(voteR,na.rm=T))) %>%
  bind_rows(.id = 'boots') %>%
  group_by(treatment,propensity,rich,period) %>%
```

```
  summarise(y=mean(mean),lower=quantile(mean,0.025),upper=quantile(mean,0.975))

pdf('pointrange_voteRVsRound_byTreatmentAndPropensity.pdf',
    height=5)
results %>% ungroup() %>%
  mutate(rich=recode(rich,`0`='Poor in MJ',`1`='Rich in MJ')) %>%
  ggplot(aes(y=y,ymin=lower,ymax=upper,x=period,color=rich,fill=rich,
             shape=rich)) +
  annotate("rect",xmin=0,xmax=41,ymin=-.05,ymax=.05,alpha=.2,
           color='orange',fill='lightgray') +
  geom_line(show.legend=F,position=position_dodge(width=1)) +
  geom_pointrange(position=position_dodge(width=1),size=.05,fatten=20) +
  facet_grid(propensity~treatment) +
  scale_shape_manual(values=c(21,24)) +
  scale_color_manual(values=c('black','black')) +
  scale_fill_manual(values=c('black','white')) +
  scale_x_continuous(limits=c(0,41),breaks=c(1,10,20,30,40),
                     labels=c(1,10,20,30,40)) +
  scale_y_continuous(breaks=seq(0,.8,.2),labels=seq(0,.8,.2)) +
  labs(y='Vote share of candidate R',x='Round') +
  theme_bw() +
  theme(legend.position='bottom',
    legend.justification='left',
    legend.box.margin=margin(-10,0,-0,0),
    legend.title=element_blank(),
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank())
dev.off()
```

```
## pdf
##   2
```

```
# Number of MJ with positive/negative time trend (p.25, footnote 28):
tidy(data_coordinationMechanism %>% group_by(groupTreatment,period) %>%
    summarise(voteR=mean(voteR)) %>% group_by(groupTreatment) %>%
    do(m=lm_robust(voteR~period,data=.)),m) %>% filter(term=='period') %>%
  mutate(time.trend=ifelse(estimate>0&p.value<.05,'Positive, significant',
    ifelse(estimate>0&p.value>=.05,'Positive',
    ifelse(estimate<0&p.value<.05,'Negative, significant',
    ifelse(estimate<0&p.value>=.05,'Negative',NA))))
  ) %>% left_join(data_coordinationMechanism) %>%
  group_by(treatment,propensity) %>%
  mutate(N=n()) %>% group_by(treatment,propensity,time.trend) %>%
  mutate(n=n(),percent=n/N*100) %>%
  summarise(N=mean(N),n=mean(n),percent=mean(percent)) %>%
  filter(treatment=='Group appeal')
```

```
## Joining, by = "groupTreatment"
```

```
## # A tibble: 5 x 6
## # Groups:   treatment, propensity [6]
##   treatment    propensity                time.trend              N     n percent
##   <fct>        <fct>                     <chr>               <dbl> <dbl>   <dbl>
## 1 Group appeal Propensity to coordinate~ Negative              672   288    42.9
## 2 Group appeal Propensity to coordinate~ Negative, signific~   672   384    57.1
```

```
## 3 Group appeal Propensity to coordinate~ Negative              864   384    44.4
## 4 Group appeal Propensity to coordinate~ Negative, signific~   864   192    22.2
## 5 Group appeal Propensity to coordinate~ Positive              864   288    33.3
```

```
# Vote share of R in Rich MJ-Group appeal treatment (p.24):
data %>% filter(treatment=='Rich MI - Group appeal') %>%
  summarise(voteR=mean(voteR))
```

```
##   voteR
## 1  0.92
```

```
# Vote share of R for MJ low vs medium/high heterogeneity (p.24):
data_richMajority %>%
  filter(treatment=='Group appeal'&majorityGroup=='MJ') %>%
  summarise(
    low=t.test(voteR~groupHeterogeneity.binary)$estimate[1],
    mediumHigh=t.test(voteR~groupHeterogeneity.binary)$estimate[2],
    p.value=clusWilcox.test(voteR,group=as.numeric(groupHeterogeneity.binary),
                            cluster=groupTreatment)$p.value) %>%
  mutate(diff=mediumHigh-low)
```

```
##         low mediumHigh      p.value       diff
## 1 0.8385417   0.499349 9.323251e-45 -0.3391927
```

```
# Exitsurvey: How many subjects report appeals did not only matter for them but
# also others (p.25, footnote 29):
data_coordinationMechanism %>%
  left_join(data_exitSurvey %>% filter(treatment=='No appeal'|
  treatment=='Group appeal'|treatment=='Income appeal') %>%
    droplevels(),by=c('sid','treatment')) %>%
  filter(treatment=='Group appeal') %>%
  mutate(appealMatters=
    ifelse(appealMattersToYou==1&appealMattersToOthers==1,
           'Appeal matters to me and others',
    ifelse(appealMattersToYou==1&appealMattersToOthers==0,
           'Appeal matters to me but not others',
    ifelse(appealMattersToYou==0&appealMattersToOthers==1,
           'Appeal matters to others but not me',
    'Appeals do not matter')))) %>% group_by(propensity,appealMatters) %>%
  summarise(n=n()) %>% mutate(N=max(cumsum(n)),prop=n/N) %>%
  filter(appealMatters=='Appeal matters to me and others')
```

```
## # A tibble: 2 x 5
## # Groups:   propensity [2]
##   propensity                   appealMatters                      n     N  prop
##   <fct>                        <chr>                          <int> <int> <dbl>
## 1 Propensity to coordinate on P Appeal matters to me and othe~   288   672 0.429
## 2 Propensity to coordinate on R Appeal matters to me and othe~   288   864 0.333
```

```
# Vote share of R for rich MJ low vs medium/high heterogeneity (p.25):
data_richMajority %>%
  filter(treatment=='Group appeal'&majorityGroup=='MJ'&rich==1) %>%
  summarise(
    low=t.test(voteR~groupHeterogeneity.binary)$estimate[1],
    mediumHigh=t.test(voteR~groupHeterogeneity.binary)$estimate[2],
    p.value=clusWilcox.test(voteR,group=as.numeric(groupHeterogeneity.binary),
                            cluster=groupTreatment)$p.value) %>%
```

```
  mutate(diff=low-mediumHigh)
```

```
##         low mediumHigh      p.value      diff
## 1 0.8554688  0.5566406 4.586391e-26 0.2988281
```

```
################################################################################
# 5.4 Robustness of experimental results
################################################################################
out <- data_richMajority_equilibria %>%
  filter(groupHeterogeneity!='High heterogeneity') %>%
  mutate(thirds=factor(ifelse(period<=12,'Round 1-12',
      ifelse(period>12&period<=24,'Round 13-24',
      ifelse(period>24&period<=36,'Round 25-36',NA)))),
    REq=ifelse(equilibrium=='(R,R,R;P,P)',1,0)) %>%
  filter(groupHeterogeneity=='Low heterogeneity')

# Relative frequency of (R,R,R;P,P) strategy profile by thirds of rounds 1-36 (p.25):
out %>% group_by(thirds,groupHeterogeneity,eqVoteWinner) %>% summarise(n=n()) %>%
  mutate(N=max(cumsum(n)),prop=n/N) %>%
  filter(eqVoteWinner=='R wins, MJ votes R and MI votes P')
```

```
## # A tibble: 3 x 6
## # Groups:   thirds, groupHeterogeneity [3]
##   thirds      groupHeterogenei~ eqVoteWinner                        n     N  prop
##   <fct>       <fct>             <fct>                           <int> <int> <dbl>
## 1 Round 1-12  Low heterogeneity R wins, MJ votes R and MI vot~   125   380 0.329
## 2 Round 13-24 Low heterogeneity R wins, MJ votes R and MI vot~    55   190 0.289
## 3 Round 25-36 Low heterogeneity R wins, MJ votes R and MI vot~   505   950 0.532
```

```
# Difference in relative frequency of of (R,R,R;P,P) strategy profile first/second
# vs last third (p.25, p-value):
out %>% group_by(thirds,groupTreatment) %>% summarise(REq=mean(REq)) %>%
  ungroup() %>% filter(thirds!='Round 13-24') %>%
  summarise(
    first=t.test(REq~thirds)$estimate[1],
    last=t.test(REq~thirds)$estimate[2],
    p.value=
      pvalue(wilcoxsign_test(REq~as.numeric(thirds),distribution='exact'))) %>%
  mutate(diff=last-first)
```

```
## # A tibble: 1 x 4
##   first  last p.value        diff
##   <dbl> <dbl> <pvalue>      <dbl>
## 1 0.329 0.532 <2.220446e-16 0.203
```

```
out %>% group_by(thirds,groupTreatment) %>% summarise(REq=mean(REq)) %>%
  ungroup() %>% filter(thirds!='Round 1-12') %>%
  summarise(
    second=t.test(REq~thirds)$estimate[1],
    last=t.test(REq~thirds)$estimate[2],
    p.value=
      pvalue(wilcoxsign_test(REq~as.numeric(thirds),distribution='exact'))) %>%
  mutate(diff=last-second)
```

```
## # A tibble: 1 x 4
##   second  last p.value        diff
```

```
##    <dbl> <dbl> <pvalue>      <dbl>
## 1  0.289 0.532 <2.220446e-16 0.242
```

```
# Vote share of R in no appeal and group appeal treatments (main treatments) vs
# poor MJ-no appeal and poor MJ-group appeal treatments (supplemental
# treatments, p.26):
data %>% filter((treatment=='No appeal'|treatment=='Group appeal'|
  treatment=='Poor MJ - No appeal'|treatment=='Poor MJ - Group appeal')) %>%
  group_by(treatment) %>% summarise(voteR=mean(voteR))
```

```
## # A tibble: 4 x 2
##   treatment              voteR
##   <fct>                  <dbl>
## 1 No appeal              0.38
## 2 Group appeal           0.381
## 3 Poor MJ - No appeal    0.714
## 4 Poor MJ - Group appeal 0.612
```

```
# Difference in votes share main vs supplemental treatments (no appeal and group
# appeal, p.26, p-value):
data %>% mutate(
  reverseDistribution=ifelse(treatment=='Poor MJ - No appeal'|
                               treatment=='Poor MJ - Group appeal',1,
    ifelse(treatment=='No appeal'|treatment=='Group appeal',0,NA))) %>%
  filter(!is.na(reverseDistribution)) %>% summarise(
    mainTreatments=t.test(voteR~reverseDistribution)$estimate[1],
    supplTreatments=t.test(voteR~reverseDistribution)$estimate[2],
    p.value=
      clusWilcox.test(voteR,group=reverseDistribution,cluster=sid)$p.value) %>%
  mutate(diff=supplTreatments-mainTreatments)
```

```
##   mainTreatments supplTreatments      p.value      diff
## 1      0.3803333           0.658 1.583002e-10 0.2776667
```

```
################################################################################
################################################################################
# Online appendix
################################################################################
# B Experimental design appendix
################################################################################
# B.3 Treatments
################################################################################
# Table B.2: Summary of all treatment conditions and treatment statistics.
################################################################################
out <- data %>% group_by(treatment) %>%
  summarise(
    N.societies=length(unique(groupTreatment)),
    N.subjects=length(unique(sid)),
    N=length(sid),
    N.low.het=length(sid[groupHeterogeneity=='Low heterogeneity']),
    N.medium.het=length(sid[groupHeterogeneity=='Medium heterogeneity']),
    N.high.het=length(sid[groupHeterogeneity=='High heterogeneity']))

out
```

```
## # A tibble: 7 x 7
```

```
##   treatment       N.societies N.subjects     N N.low.het N.medium.het N.high.het
##   <fct>                 <int>      <int> <int>     <int>        <int>      <int>
## 1 No appeal                14         70  2800       560         1960        280
## 2 Group appeal             16         80  3200       640         2240        320
## 3 Income appeal             8         40  1600       320         1120        160
## 4 Poor MJ - No a~           9         45  1800       540         1080        180
## 5 Poor MJ - Grou~          11         55  2200       660         1320        220
## 6 Poor MJ - Inco~           8         40  1600       480          960        160
## 7 Rich MI - Grou~           2         10   300       100          200          0
```

```
print(xtable(out,type='latex'),file='tableB2.tex')

###############################################################################
# Table B.3: Treatment balance: summary statistics of exit-survey responses
###############################################################################
var.list <- c('age','german','welfare','taxForEducation','taxForWelfare',
              'closeToGroup','groupIDKlee','correctGroupID')
out <- data %>% left_join(data_exitSurvey) %>%
  select(treatment,age,ethnicity,welfare,publicSpending,closeToGroup,klee,
         whichGroupID,groupID,sid) %>%
  mutate(
    german = ifelse(ethnicity=="Bayern"|ethnicity=="Bayrisch"|
      ethnicity=="Berlin"|ethnicity=="Berlin (egal ob west oder ost)"|
        ethnicity=="Brandenburg"|
      ethnicity=="Brandenburg-Berlin-Region, aber auch Os"|ethnicity=="Deusch"|
        ethnicity=="Deutsch"|
      ethnicity=="Deutsch Griechisch (Europ„isch)"|ethnicity=="Deutsche"|
        ethnicity=="Deutscher"|
      ethnicity=="Deutschland"|ethnicity=="Deutschland, christlich, weiá"|
      ethnicity=="Geburtsort. In meinem Fall: Leipzig"|ethnicity==" Hessen"|
        ethnicity=="Leverkusen"|
      ethnicity=="Mein Elternhaus, deutsch, Tendenz europ Muddastadt Berlin"|
        ethnicity=="Niedersachsen"|
      ethnicity=="Norddeutschland"|ethnicity=="berlin"|ethnicity=="berlin lichtenrade"|
      ethnicity=="berliner"|ethnicity=="de"|ethnicity=="deutsch"|
      ethnicity=="deutsch und sorbisch als Sprache"|ethnicity=="deutsch, europ„isch"|
      ethnicity=="deutschland"|ethnicity=="europ„isch"|ethnicity=="gesamtdeutsch",1,
      ifelse(ethnicity=='',NA,0)),
    taxForEducation=ifelse(publicSpending=='Bildung',1,ifelse(publicSpending!='',0,NA)),
    taxForWelfare=ifelse(publicSpending=='Soziales',1,ifelse(publicSpending!='',0,NA)),
    groupIDKlee=ifelse(groupID==klee,1,ifelse(!is.na(groupID),0,NA)),
    correctGroupID=ifelse(whichGroupID==groupID,1,ifelse(!is.na(whichGroupID),0,NA))
    ) %>% group_by(treatment,sid) %>% summarise_at(vars(var.list),mean) %>% ungroup()
```

```
## Joining, by = c("subject", "date", "treatment", "session", "klee", "kandinsky", "sid")
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(var.list)` instead of `var.list` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
# Balance variables (online appendix, p.4, p-value):
out %>% filter(treatment=='No appeal'|treatment=='Group appeal') %>%
 summarise_at(vars(var.list[-length(var.list)]),
              funs(wilcox.test(.~treatment)$p.value))
```

```
## # A tibble: 1 x 7
##     age german welfare taxForEducation taxForWelfare closeToGroup groupIDKlee
##   <dbl>  <dbl>   <dbl>           <dbl>         <dbl>        <dbl>       <dbl>
## 1 0.517  0.130   0.105           0.995         0.269        0.654           1
```

```
out %>% filter(treatment=='No appeal'|treatment=='Income appeal') %>%
  summarise_at(vars(var.list[-length(var.list)]),
               funs(wilcox.test(.~treatment)$p.value))
```

```
## # A tibble: 1 x 7
##     age german welfare taxForEducation taxForWelfare closeToGroup groupIDKlee
##   <dbl>  <dbl>   <dbl>           <dbl>         <dbl>        <dbl>       <dbl>
## 1 0.121  0.595   0.915           0.374         0.988        0.584           1
```

```
out %>% filter(treatment=='No appeal'|treatment=='Poor MJ - No appeal') %>%
  summarise_at(vars(var.list[-length(var.list)]),
               funs(wilcox.test(.~treatment)$p.value))
```

```
## # A tibble: 1 x 7
##     age german welfare taxForEducation taxForWelfare closeToGroup groupIDKlee
##   <dbl>  <dbl>   <dbl>           <dbl>         <dbl>        <dbl>       <dbl>
## 1 0.203  0.376   0.250           0.187         0.544        0.223       0.910
```

```
out %>% filter(treatment=='No appeal'|treatment=='Poor MJ - Group appeal') %>%
  summarise_at(vars(var.list[-length(var.list)]),
               funs(wilcox.test(.~treatment)$p.value))
```

```
## # A tibble: 1 x 7
##     age german welfare taxForEducation taxForWelfare closeToGroup groupIDKlee
##   <dbl>  <dbl>   <dbl>           <dbl>         <dbl>        <dbl>       <dbl>
## 1 0.220  0.376   0.803           0.898         0.566        0.284       0.922
```

```
out %>% filter(treatment=='No appeal'|treatment=='Poor MJ - Income appeal') %>%
  summarise_at(vars(var.list[-length(var.list)]),
               funs(wilcox.test(.~treatment)$p.value))
```

```
## # A tibble: 1 x 7
##       age german welfare taxForEducation taxForWelfare closeToGroup groupIDKlee
##     <dbl>  <dbl>   <dbl>           <dbl>         <dbl>        <dbl>       <dbl>
## 1 0.00896  0.504   0.129           0.991         0.973        0.522           1
```

```
out %>% filter(treatment=='No appeal'|treatment=='Rich MI - Group appeal') %>%
  summarise_at(vars(var.list[-length(var.list)]),
               funs(wilcox.test(.~treatment)$p.value))
```

```
## # A tibble: 1 x 7
##     age german welfare taxForEducation taxForWelfare closeToGroup groupIDKlee
##   <dbl>  <dbl>   <dbl>           <dbl>         <dbl>        <dbl>       <dbl>
## 1 0.534  0.848   0.125           0.280         0.634       0.0171           1
```

```
##############################################################################
# Table B.3: Treatment balance: summary statistics of exit-survey responses
##############################################################################
count.N <- function(x) sum(!is.na(x))
out.1 <- out %>% group_by(treatment) %>%
  summarise_at(vars(var.list),list(m=mean,sd=sd,min=min,max=max),na.rm=T) %>%
  left_join(out %>% group_by(treatment) %>%
              summarise_at(vars(var.list), funs(N=count.N))) %>%
```

```
  mutate(n=ifelse(treatment!='Rich MI - Group appeal',40,30)) %>%
  mutate_at(
    vars(age_N,german_N,welfare_N,taxForEducation_N,taxForWelfare_N,
         closeToGroup_N,correctGroupID_N,groupIDKlee_N),
    list(~./n)) %>% pivot_longer(-treatment,names_to=c('variable','.value'),
        names_pattern='(.+)_(.+)') %>%
  filter(!is.na(variable)) %>% select(treatment,variable,N,m,sd,min,max) %>%
  group_split(treatment)
```

```
## Joining, by = "treatment"
```

```
out.1
```

```
## [[1]]
## # A tibble: 8 x 7
##   treatment variable            N      m    sd   min   max
##   <fct>     <chr>           <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 No appeal age             1.7   24.5   5.06     18    50
## 2 No appeal german          1.58   0.587 0.496     0     1
## 3 No appeal welfare         1.7    2.26  0.891     1     5
## 4 No appeal taxForEducation 1.7    0.588 0.496     0     1
## 5 No appeal taxForWelfare   1.7    0.176 0.384     0     1
## 6 No appeal closeToGroup    1.7    5.54  2.95      0    10
## 7 No appeal groupIDKlee     1.75   0.5   0.504     0     1
## 8 No appeal correctGroupID  0.725  1     0         1     1
##
## [[2]]
## # A tibble: 8 x 7
##   treatment    variable            N       m     sd   min   max
##   <fct>        <chr>           <dbl>   <dbl>  <dbl> <dbl> <dbl>
## 1 Group appeal age              1.98  24.3    5.47     18    49
## 2 Group appeal german           1.9    0.711  0.457     0     1
## 3 Group appeal welfare          2      2.58   1.13      1     5
## 4 Group appeal taxForEducation  2      0.588  0.495     0     1
## 5 Group appeal taxForWelfare    2      0.112  0.318     0     1
## 6 Group appeal closeToGroup     2      5.41   3.06      0    10
## 7 Group appeal groupIDKlee      2      0.5    0.503     0     1
## 8 Group appeal correctGroupID   0    NaN     NA       Inf  -Inf
##
## [[3]]
## # A tibble: 8 x 7
##   treatment     variable            N      m    sd   min   max
##   <fct>         <chr>           <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Income appeal age             0.925 25.8   5.43     20    45
## 2 Income appeal german          0.775  0.645 0.486     0     1
## 3 Income appeal welfare         0.95   2.29  0.984     1     5
## 4 Income appeal taxForEducation 1      0.675 0.474     0     1
## 5 Income appeal taxForWelfare   1      0.175 0.385     0     1
## 6 Income appeal closeToGroup    1      5.95  2.84      0    10
## 7 Income appeal groupIDKlee     1      0.5   0.506     0     1
## 8 Income appeal correctGroupID  1      1     0         1     1
##
## [[4]]
## # A tibble: 8 x 7
##   treatment           variable            N      m    sd   min   max
```

```
##   <fct>               <chr>           <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Poor MJ - No appeal age             1.02 25.4   4.86     18    43
## 2 Poor MJ - No appeal german          1     0.675 0.474     0     1
## 3 Poor MJ - No appeal welfare         1.12  2.58  1.18      1     5
## 4 Poor MJ - No appeal taxForEducation 1.12  0.711 0.458     0     1
## 5 Poor MJ - No appeal taxForWelfare   1.12  0.133 0.344     0     1
## 6 Poor MJ - No appeal closeToGroup    1.1   4.89  3.20      0    10
## 7 Poor MJ - No appeal groupIDKlee     1.12  0.489 0.506     0     1
## 8 Poor MJ - No appeal correctGroupID  1.12  1     0         1     1
##
## [[5]]
## # A tibble: 8 x 7
##   treatment              variable            N      m    sd   min   max
##   <fct>                  <chr>           <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Poor MJ - Group appeal age             1.3   24.8   3.85     18    39
## 2 Poor MJ - Group appeal german          1      0.675 0.474     0     1
## 3 Poor MJ - Group appeal welfare         1.32   2.32  0.915     1     5
## 4 Poor MJ - Group appeal taxForEducation 1.38   0.6   0.494     0     1
## 5 Poor MJ - Group appeal taxForWelfare   1.38   0.218 0.417     0     1
## 6 Poor MJ - Group appeal closeToGroup    1.35   6.19  2.51      0    10
## 7 Poor MJ - Group appeal groupIDKlee     1.38   0.509 0.505     0     1
## 8 Poor MJ - Group appeal correctGroupID  0.975  1     0         1     1
##
## [[6]]
## # A tibble: 8 x 7
##   treatment               variable            N      m    sd   min   max
##   <fct>                   <chr>           <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Poor MJ - Income appeal age             0.925 26.5   5.27     18    45
## 2 Poor MJ - Income appeal german          0.825  0.515 0.508     0     1
## 3 Poor MJ - Income appeal welfare         0.975  2.54  0.996     1     5
## 4 Poor MJ - Income appeal taxForEducation 0.975  0.590 0.498     0     1
## 5 Poor MJ - Income appeal taxForWelfare   0.975  0.179 0.389     0     1
## 6 Poor MJ - Income appeal closeToGroup    1      5.92  3.08      0    10
## 7 Poor MJ - Income appeal groupIDKlee     1      0.5   0.506     0     1
## 8 Poor MJ - Income appeal correctGroupID  1      1     0         1     1
##
## [[7]]
## # A tibble: 8 x 7
##   treatment              variable            N      m    sd   min   max
##   <fct>                  <chr>           <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Rich MI - Group appeal age             0.267 24.4   2.50     21    28
## 2 Rich MI - Group appeal german          0.267  0.625 0.518     0     1
## 3 Rich MI - Group appeal welfare         0.3    2.89  1.27      1     5
## 4 Rich MI - Group appeal taxForEducation 0.3    0.778 0.441     0     1
## 5 Rich MI - Group appeal taxForWelfare   0.3    0.111 0.333     0     1
## 6 Rich MI - Group appeal closeToGroup    0.333  7.7   2.79      2    10
## 7 Rich MI - Group appeal groupIDKlee     0.333  0.5   0.527     0     1
## 8 Rich MI - Group appeal correctGroupID  0.333  1     0         1     1
##
## attr(,"ptype")
## # A tibble: 0 x 7
## # ... with 7 variables: treatment <fct>, variable <chr>, N <dbl>, m <dbl>,
## #   sd <dbl>, min <dbl>, max <dbl>
```

```
print(xtable(out.1[[1]],type='latex',caption='No appeal'),
      file='tableB3_1.tex')
print(xtable(out.1[[2]],type='latex',caption='Group appeal'),
      file='tableB3_2.tex')
print(xtable(out.1[[3]],type='latex',caption='Income appeal'),
      file='tableB3_3.tex')
print(xtable(out.1[[4]],type='latex',caption='Poor MJ - No appeal'),
      file='tableB3_4.tex')
print(xtable(out.1[[5]],type='latex',caption='Poor MJ - Group appeal'),
      file='tableB3_5.tex')
print(xtable(out.1[[6]],type='latex',caption='Poor MJ - Income appeal'),
      file='tableB3_6.tex')
print(xtable(out.1[[7]],type='latex',caption='Rich MI - Group appeal'),
      file='tableB3_7.tex')

################################################################################
# C Statistical appendix
################################################################################
# C.1 Summary statistics
################################################################################
# Table C.6:  Relative frequency of strategy profiles by group heterogeneity
# and appeal treatments
################################################################################
out <- results.equ.play %>% spread(groupHeterogeneity,prop) %>%
  group_by(eqVoteWinner,treatment) %>%
  summarise(Low=mean(`Low heterogeneity`,na.rm=T),
      Medium=mean(`Medium heterogeneity`,na.rm=T),
      High=mean(`High heterogeneity`,na.rm=T)) %>%
  rename(variable=eqVoteWinner) %>%
  mutate(High=ifelse(is.na(High),0,High)) %>%
  mutate_at(vars(Low:High),funs(round(.,2))) %>%
  arrange(variable,treatment)

out
```

```
## # A tibble: 12 x 5
## # Groups:   variable [4]
##    variable                           treatment       Low Medium  High
##    <fct>                              <fct>         <dbl>  <dbl> <dbl>
##  1 P wins, all vote P                 No appeal     0.01    0.21  0.16
##  2 P wins, all vote P                 Group appeal  0.03    0.19  0.25
##  3 P wins, all vote P                 Income appeal 0.03    0.15  0.31
##  4 P wins, MJ or MI split             No appeal     0.28    0.53  0.68
##  5 P wins, MJ or MI split             Group appeal  0.24    0.51  0.61
##  6 P wins, MJ or MI split             Income appeal 0.47    0.46  0.69
##  7 R wins, MJ or MI split             No appeal     0.290   0.16  0.14
##  8 R wins, MJ or MI split             Group appeal  0.21    0.11  0.02
##  9 R wins, MJ or MI split             Income appeal 0.12    0.19  0
## 10 R wins, MJ votes R and MI votes P No appeal     0.42    0.1   0.02
## 11 R wins, MJ votes R and MI votes P Group appeal  0.52    0.19  0.12
## 12 R wins, MJ votes R and MI votes P Income appeal 0.38    0.2   0
```

```
print(xtable(out,type='latex'),file='tableC6.tex')
```

```
##############################################################################
# Table C.7:  Summary statistics of main variables by income and appeal
# treatments. Statistics are pooled across all levels of group heterogeneity,
# subjects, and rounds within one treatment
##############################################################################
out <- rbind(
  data %>% mutate(incomeLevel='All',variable='Vote R') %>%
    group_by(treatment,variable,incomeLevel) %>%
    summarise_at(vars(voteR),list(m=mean,sd=sd)) %>%
    pivot_wider(names_from=treatment,values_from=c(m,sd)),
  data %>% mutate(
    incomeLevel=factor(ifelse(income<30,'Very poor',
      ifelse(income>30&income<50,'Moderately poor',
      ifelse(income>50&income<70,'Moderately rich','Very rich'))),
      levels=c('Very poor','Moderately poor','Moderately rich','Very rich')),
    variable='Vote R') %>%
    group_by(treatment,variable,incomeLevel) %>%
    summarise_at(vars(voteR),list(m=mean,sd=sd)) %>%
    pivot_wider(names_from=treatment,values_from=c(m,sd)),
  data %>% mutate(incomeLevel='All',variable='R wins election') %>%
    group_by(treatment,variable,incomeLevel) %>%
    summarise_at(vars(RWins),list(m=mean,sd=sd)) %>%
    pivot_wider(names_from=treatment,values_from=c(m,sd)),
  data %>% mutate(incomeLevel='All',variable='income') %>%
    group_by(treatment,variable,incomeLevel) %>%
    summarise_at(vars(income),list(m=mean,sd=sd)) %>%
    pivot_wider(names_from=treatment,values_from=c(m,sd)),
  data %>% mutate(incomeLevel='All',variable='Number of observations') %>%
    group_by(treatment,variable,incomeLevel) %>%
    summarise(m=length(sid),sd=length(sid)) %>%
    pivot_wider(names_from=treatment,values_from=c(m,sd)),
  data %>% mutate(incomeLevel='All',variable='Number of subjects') %>%
    group_by(treatment,variable,incomeLevel) %>%
    summarise(m=length(unique(sid)),sd=length(unique(sid))) %>%
    pivot_wider(names_from=treatment,values_from=c(m,sd))) %>%
select(variable,incomeLevel,`m_No appeal`,`sd_No appeal`,`m_Group appeal`,
       `sd_Group appeal`,`m_Income appeal`,`sd_Income appeal`,
       `m_Poor MJ - No appeal`,`sd_Poor MJ - No appeal`,
       `m_Poor MJ - Group appeal`,`sd_Poor MJ - Group appeal`,
       `m_Poor MJ - Income appeal`,`sd_Poor MJ - Income appeal`,
       `m_Rich MI - Group appeal`,`sd_Rich MI - Group appeal`)

out
```

```
## # A tibble: 9 x 16
## # Groups:   variable [5]
##   variable incomeLevel `m_No appeal` `sd_No appeal` `m_Group appeal`
##   <chr>    <chr>               <dbl>          <dbl>            <dbl>
## 1 Vote R   All                 0.38           0.485            0.381
## 2 Vote R   Very poor           0.231          0.422            0.241
## 3 Vote R   Moderately~         0.299          0.458            0.197
## 4 Vote R   Moderately~         0.537          0.499            0.580
## 5 Vote R   Very rich           0.588          0.493            0.653
## 6 R wins ~ All                 0.343          0.475            0.369
```

```
## 7 income   All                45.3           19.9            45.3
## 8 Number ~ All              2800           2800            3200
## 9 Number ~ All                70             70              80
## # ... with 11 more variables: `sd_Group appeal` <dbl>, `m_Income appeal` <dbl>,
## #   `sd_Income appeal` <dbl>, `m_Poor MJ - No appeal` <dbl>, `sd_Poor MJ - No
## #   appeal` <dbl>, `m_Poor MJ - Group appeal` <dbl>, `sd_Poor MJ - Group
## #   appeal` <dbl>, `m_Poor MJ - Income appeal` <dbl>, `sd_Poor MJ - Income
## #   appeal` <dbl>, `m_Rich MI - Group appeal` <dbl>, `sd_Rich MI - Group
## #   appeal` <dbl>
```

```
print(xtable(out,type='latex'),file='tableC7.tex')

out %>% filter(variable=='R wins election') %>%
  select(variable,`m_No appeal`,`sd_No appeal`)
```

```
## # A tibble: 1 x 3
## # Groups:   variable [1]
##   variable        `m_No appeal` `sd_No appeal`
##   <chr>                   <dbl>          <dbl>
## 1 R wins election         0.343          0.475
```

```
out %>% filter(variable=='Vote R'&incomeLevel=='Moderately poor') %>%
  select(variable,incomeLevel,`m_Poor MJ - Group appeal`,
         `sd_Poor MJ - Group appeal`)
```

```
## # A tibble: 1 x 4
## # Groups:   variable [1]
##   variable incomeLevel     `m_Poor MJ - Group appeal` `sd_Poor MJ - Group appea~
##   <chr>    <chr>                                <dbl>                      <dbl>
## 1 Vote R   Moderately poor                      0.465                      0.499
```

```
################################################################################
# C.2 Additional statistical analysis
################################################################################
# Table C.8: Multi-level random effects regression of indicator for strategy
# profile (R,R,R;P,P) being played and of indicator for strategy profile 3
# (P,P,P;P,P), P-equilibrium, being played on group heterogeneity treatment,
# appeal treatment, interaction of those treatments, and round of play
# including random intercepts for societies.
################################################################################
# Equilibrium play + relative frequency of vote winner over round
out <- data_richMajority_equilibria %>% mutate(
    RRRPPplayed = ifelse(equilibrium=='(R,R,R;P,P)',1,0),
    PPPPPplayed = ifelse(equilibrium=='(P,P,P;P,P)',1,0))
re1 <- lmer(RRRPPplayed~groupHeterogeneity*treatment+period+
              (1|groupTreatment),data=out)
re2 <- lmer(PPPPPplayed~groupHeterogeneity*treatment+period+
              (1|groupTreatment),data=out)
summary(re1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: RRRPPplayed ~ groupHeterogeneity * treatment + period + (1 |
##     groupTreatment)
##    Data: out
##
## REML criterion at convergence: 4899
```

```
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7520 -0.4596 -0.1407  0.2007  3.1343
##
## Random effects:
##  Groups         Name        Variance Std.Dev.
##  groupTreatment (Intercept) 0.03986  0.1997
##  Residual                   0.10843  0.3293
## Number of obs: 7600, groups:  groupTreatment, 38
##
## Fixed effects:
##                                                                  Estimate
## (Intercept)                                                     0.3766873
## groupHeterogeneityMedium heterogeneity                         -0.3105883
## groupHeterogeneityHigh heterogeneity                           -0.4327425
## treatmentGroup appeal                                           0.0959821
## treatmentIncome appeal                                         -0.0446429
## period                                                          0.0019198
## groupHeterogeneityMedium heterogeneity:treatmentGroup appeal   -0.0102041
## groupHeterogeneityHigh heterogeneity:treatmentGroup appeal      0.0111607
## groupHeterogeneityMedium heterogeneity:treatmentIncome appeal   0.1460459
## groupHeterogeneityHigh heterogeneity:treatmentIncome appeal     0.0267857
##                                                                Std. Error
## (Intercept)                                                     0.0558341
## groupHeterogeneityMedium heterogeneity                          0.0158979
## groupHeterogeneityHigh heterogeneity                            0.0249120
## treatmentGroup appeal                                           0.0755103
## treatmentIncome appeal                                          0.0914475
## period                                                          0.0003909
## groupHeterogeneityMedium heterogeneity:treatmentGroup appeal    0.0216051
## groupHeterogeneityHigh heterogeneity:treatmentGroup appeal      0.0330024
## groupHeterogeneityMedium heterogeneity:treatmentIncome appeal   0.0261651
## groupHeterogeneityHigh heterogeneity:treatmentIncome appeal     0.0399679
##                                                                t value
## (Intercept)                                                      6.747
## groupHeterogeneityMedium heterogeneity                         -19.536
## groupHeterogeneityHigh heterogeneity                           -17.371
## treatmentGroup appeal                                            1.271
## treatmentIncome appeal                                          -0.488
## period                                                           4.912
## groupHeterogeneityMedium heterogeneity:treatmentGroup appeal    -0.472
## groupHeterogeneityHigh heterogeneity:treatmentGroup appeal       0.338
## groupHeterogeneityMedium heterogeneity:treatmentIncome appeal    5.582
## groupHeterogeneityHigh heterogeneity:treatmentIncome appeal      0.670
##
## Correlation of Fixed Effects:
##             (Intr) grpHMh grpHHh trtmGa trtmIa period gHMh:Ga gHHh:Ga gHMh:Ia
## grpHtrgntMh -0.237
## grpHtrgntHh -0.100  0.458
## trtmntGrpap -0.721  0.161  0.103
## trtmntIncma -0.596  0.133  0.085  0.440
## period      -0.157  0.122 -0.253  0.000  0.000
## grpHtrMh:Ga  0.161 -0.725 -0.360 -0.223 -0.098  0.000
```

```
## grpHtrHh:Ga  0.105 -0.369 -0.707 -0.146 -0.064  0.000  0.509
## grpHtrMh:Ia  0.133 -0.598 -0.297 -0.098 -0.223  0.000  0.440   0.224
## grpHtrHh:Ia  0.087 -0.305 -0.583 -0.064 -0.146  0.000  0.224   0.440   0.509
```

```
summary(re2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: PPPPPplayed ~ groupHeterogeneity * treatment + period + (1 |
##     groupTreatment)
##    Data: out
##
## REML criterion at convergence: 4066.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.3285 -0.5622 -0.1954  0.1554  3.1543
##
## Random effects:
##  Groups         Name        Variance Std.Dev.
##  groupTreatment (Intercept) 0.03247  0.1802
##  Residual                   0.09721  0.3118
## Number of obs: 7600, groups:  groupTreatment, 38
##
## Fixed effects:
##                                                                  Estimate
## (Intercept)                                                     -0.0462542
## groupHeterogeneityMedium heterogeneity                           0.2099914
## groupHeterogeneityHigh heterogeneity                             0.1120171
## treatmentGroup appeal                                            0.0223214
## treatmentIncome appeal                                           0.0223214
## period                                                           0.0024663
## groupHeterogeneityMedium heterogeneity:treatmentGroup appeal    -0.0414541
## groupHeterogeneityHigh heterogeneity:treatmentGroup appeal       0.0669643
## groupHeterogeneityMedium heterogeneity:treatmentIncome appeal -0.0816327
## groupHeterogeneityHigh heterogeneity:treatmentIncome appeal      0.1294643
##                                                                 Std. Error
## (Intercept)                                                      0.0506087
## groupHeterogeneityMedium heterogeneity                           0.0150528
## groupHeterogeneityHigh heterogeneity                             0.0235878
## treatmentGroup appeal                                            0.0683648
## treatmentIncome appeal                                           0.0827939
## period                                                           0.0003701
## groupHeterogeneityMedium heterogeneity:treatmentGroup appeal     0.0204567
## groupHeterogeneityHigh heterogeneity:treatmentGroup appeal       0.0312482
## groupHeterogeneityMedium heterogeneity:treatmentIncome appeal    0.0247743
## groupHeterogeneityHigh heterogeneity:treatmentIncome appeal      0.0378434
##                                                                 t value
## (Intercept)                                                      -0.914
## groupHeterogeneityMedium heterogeneity                           13.950
## groupHeterogeneityHigh heterogeneity                              4.749
## treatmentGroup appeal                                             0.327
## treatmentIncome appeal                                            0.270
## period                                                            6.664
## groupHeterogeneityMedium heterogeneity:treatmentGroup appeal     -2.026
## groupHeterogeneityHigh heterogeneity:treatmentGroup appeal        2.143
```

```
## groupHeterogeneityMedium heterogeneity:treatmentIncome appeal  -3.295
## groupHeterogeneityHigh heterogeneity:treatmentIncome appeal     3.421
##
## Correlation of Fixed Effects:
##             (Intr) grpHMh grpHHh trtmGa trtmIa period gHMh:Ga gHHh:Ga gHMh:Ia
## grpHtrgntMh -0.248
## grpHtrgntHh -0.104  0.458
## trtmntGrpap -0.720  0.169  0.108
## trtmntIncma -0.595  0.139  0.089  0.440
## period      -0.164  0.122 -0.253  0.000  0.000
## grpHtrMh:Ga  0.168 -0.725 -0.360 -0.233 -0.102  0.000
## grpHtrHh:Ga  0.110 -0.369 -0.707 -0.152 -0.067  0.000  0.509
## grpHtrMh:Ia  0.138 -0.598 -0.297 -0.102 -0.233  0.000  0.440   0.224
## grpHtrHh:Ia  0.091 -0.305 -0.583 -0.067 -0.152  0.000  0.224   0.440   0.509
```

```
notShow <- capture.output(
  stargazer(re1,re2,column.labels=c('(R,R,R;P,P)','(P,P,P;P,P)'),
            order=c(1,2,3,4,6,7,8,9,5,10),
  covariate.labels = c('Medium heterogeneity','High heterogeneity',
  'Group appeal','Income appeal','Medium heterogeneity $\\times$ Group appeal',
  'High heterogeneity $\\times$ Group appeal',
  'Medium heterogeneity $\\times$ Income appeal',
  'High heterogeneity $\\times$ Income appeal','Round','Constant'),
  out='tableC8.tex'))

##############################################################################
# Figure C.6: Distribution of relative frequency of strategy profiles by group
# heterogeneity and appeal treatments in first third (round 1-12), second
# third (round 13-24), and final third (round 25-36) of the experiment.
# Observations on the high group heterogeneity treatment (round 37-40) are
# omitted.
##############################################################################
pdf('barPlotStacked_shareEquPlay_byTreatmentAndIncDistrAndVoteWinnerThirds.pdf',
    height=5)
data_richMajority_equilibria %>%
  filter(groupHeterogeneity!='High heterogeneity') %>%
  mutate(thirds = ifelse(period<=12,'Round 1-12',
    ifelse(period>12&period<=24,'Round 13-24',
    ifelse(period>24&period<=36,'Round 25-36',NA))),
  eqVoteWinner=recode(eqVoteWinner,
    'P wins, all vote P'='Redistributive cand. P wins, all vote P',
    'P wins, MJ or MI split'='Redistributive cand. P wins, MJ or MI split',
    'R wins, MJ or MI split'='Wealth-preserving cand. R wins, MJ or MI split',
    'R wins, MJ votes R and MI votes P'=
      'Wealth-preserving cand. R wins, MJ votes R and MI votes P')) %>%
  group_by(treatment,thirds,groupHeterogeneity,eqVoteWinner) %>%
  summarise(n=n()) %>%
  mutate(N=max(cumsum(n)),prop=n/N) %>%
  ggplot(aes(y=prop,x=treatment,color=eqVoteWinner,fill=eqVoteWinner)) +
    geom_col() +
    facet_grid(groupHeterogeneity~thirds) +
    scale_color_manual(values=c('blue','lightblue','indianred1','red')) +
    scale_fill_manual(values=c('blue','lightblue','indianred1','red')) +
    guides(fill=guide_legend(nrow=2),color=guide_legend(nrow=2)) +
```

```
    labs(y='Cumulative relative frequency',x='') +
    theme_bw() +
    theme(legend.position='bottom',
      legend.title=element_blank(),
      legend.spacing.x = unit(.25, 'cm'),
      axis.text.x = element_text(angle=45,hjust=1))
 dev.off()
```

```
## pdf
##   2
```