Replication Material and Supplementary Material: Citizenship Regimes and the Politicization of Immigrant Groups

Didier Ruedin, didier.ruedin@wolfson.oxon.org

19 January 2017

This document contains all the analyses reported in the paper (replication material), plus supplementary material. Everything is based on claims in the regular (random) sample, and claims about the country under study. Not included are claims in the additional samples and claims that are not about the country under study.

Ruedin, Didier (2017) Citizenship Regimes and the Politicization of Immigrant Groups, *Austrian Journal of Political Science*, 46(1).

Actor Classifications

Anti-Immigrant Parties

Party	Party ID
Austria	
FPÖ	42420
BZÖ	42710
Belgium	
VB	21914
FN	21710
Ireland	
–	–
Netherlands	
PVV	22722
LPF	22720
Spain	
–	–
Switzerland	
SVP	43810
SD	43710
FP/AP	43951
United Kingdom	
UKIP	51951
BNP	51701

Only anti-immigrant parties with claims in the dataset are included; SVP since 1999, according to Ruedin (2013) Swiss party position on immigration: Ruedin, Didier (2013) Obtaining party positions on immigration in Switzerland: Comparing different methods, *Swiss Political Science Review*, 19(1): 84-105.

Left-Wing Parties

Party	Party ID
Austria	
Grüne	42110
SPÖ	42320
Belgium	
Ecolo	21111
Groen (Agalev)	21112
SP/ SP.A	21221
PS	21322
Ireland	
Green Party	53110
Democratic Left	53221
Labour Party	53320
Sinn Fein	53951
Socialist Party	53999
Netherlands	
GL	22110
SP	22220
PvdA	22320
PvdD	22951
Spain	
IU	33220
PSOE	33320
BNG	33908
Switzerland	
Grüne	43110
SP	43320
FDP	43420
United Kingdom	
Labour Party	51320

Data Preparation

```
# load claims data:
  load("C:/Users/ruedind/switchdrive/Politicization/claims.17.RData") # claims data
  # load("/home/didier/ownCloud/Politicization/claims.17.RData") # claims data
# knitr options:
  knitr::opts_chunk$set(cache=TRUE, fig.width=5, fig.height=5, echo=TRUE)
# combine asylum/refugees/illegal as voiceless groups:
  core.claims$objecta <- core.claims$object
  core.claims$objecta[is.element(core.claims$object, 2105:2108)] <- 2700
# voiceless dummy:
  core.claims$voiceless <- core.claims$objecta                # keep as basis
  core.claims$voiceless[core.claims$objecta < 2700]  <- 0     # others
  core.claims$voiceless[core.claims$objecta == 2700] <- 1     # voiceless
# anti-immigrant actors:
  core.claims$antiact <- 0                                    # other actors
  core.claims$antiact[core.claims$actor==1504] <- 1           # racist and extreme right
  core.claims$antiact[core.claims$ACTORGPARTYID==51951] <- 1 # UKIP
```

```
  core.claims$antiact[core.claims$ACTORGPARTYID==51701] <- 1 # BNP
  core.claims$antiact[core.claims$ACTORGPARTYID==22722] <- 1 # PVV
  core.claims$antiact[core.claims$ACTORGPARTYID==22720] <- 1 # LPF
  core.claims$antiact[core.claims$ACTORGPARTYID==43810 & core.claims$year>1998] <- 1
    # SVP since 1999, according to Ruedin (2013) Swiss party position on immigration
  core.claims$antiact[core.claims$ACTORGPARTYID==43710] <- 1 # SD
  core.claims$antiact[core.claims$ACTORGPARTYID==43951] <- 1 # FP
  core.claims$antiact[core.claims$ACTORGPARTYID==21914] <- 1 # VB
  core.claims$antiact[core.claims$ACTORGPARTYID==21710] <- 1 # FN
  core.claims$antiact[core.claims$ACTORGPARTYID==42420] <- 1 # FPÖ
  core.claims$antiact[core.claims$ACTORGPARTYID==42710] <- 1 # BZÖ
# left-wing parties, according to project data, based on party ID and year
  core.claims$lw <- 0                                       # others
  core.claims$lw[core.claims$ACTORGPARTYID==21111] <- 1      # left-wing parties
  core.claims$lw[core.claims$ACTORGPARTYID==21112] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==21221] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==21322] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==22110] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==22220] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==22320] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==22951] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==33220] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==33320] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==33908] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==43110] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==43320] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==51320] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==53110] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==53221] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==53320] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==53951] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==53999] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==42320] <- 1
  core.claims$lw[core.claims$ACTORGPARTYID==42110] <- 1
# CSO1 (restrictive definition): 150 (CSO) w/o 1504 (anti-immigrant), 100 (religious):
  core.claims$cso1 <- 0                     # basis
  core.claims$cso1[core.claims$acts1 %in% c(150, 1501, 1502, 1503, 1505, 1509, 100,
       1001, 1002, 1003, 1004, 1005, 1006, 1007, 1008)] <- 1       # CSO
# CSO2 (inclusive definition): CSO1 plus 160 (minorities):
  core.claims$cso2 <- core.claims$cso1                # basis
  core.claims$cso2[core.claims$acts %in% c(160, 1601, 1602)] <- 1
       # add minority organizations
# gpactor1: Actor variable for this paper: other, CSO1, LW parties, anti-immigrant actors:
  core.claims$gpactor1 <- 0                           # other
  core.claims$gpactor1[core.claims$cso1 == 1] <- 1    # CSO, definition 1 (restrictive)
  core.claims$gpactor1[core.claims$lw == 1] <- 2      # left-wing parties
  core.claims$gpactor1[core.claims$antiact == 1] <- 3 # anti-immigrant actors
# gpactor2: Actor variable for this paper: other, CSO2, LW parties, anti-immigrant actors:
  core.claims$gpactor2 <- 0                           # other
  core.claims$gpactor2[core.claims$cso2 == 1] <- 1    # CSO, definition 2 (including minorities)
  core.claims$gpactor2[core.claims$lw == 1] <- 2      # left-wing parties
  core.claims$gpactor2[core.claims$antiact == 1] <- 3 # anti-immigrant actors
# constants:
```

```
  w <- c(-1, -.5, 0, .5, 1)                          # categories of POSIT
  cy <- c("AT","BE","ES","IE","NL","CH","UK")        # countries
  attach(core.claims)
# libraries:
  library(psych)                                     # interpolated median
  library(car)                                       # enhanced scatterplot
```

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

```
## The following object is masked from 'package:psych':
##
##     logit
```

```
  library(effects)                                   # plot interaction effect
```

```
##
## Attaching package: 'effects'
```

```
## The following object is masked from 'package:car':
##
##     Prestige
```

```
  library(visreg)
```

Positions (Tone of Claims)

```
table(voiceless, posit)
```

```
##          posit
## voiceless  -1 -0.5   0 0.5   1
##         0 457  430 788 831 960
##         1 250  275 378 366 459
```

```
# aggregate(posit, by=list(voice=voiceless), mean, na.rm=TRUE)
aggregate(posit, by=list(voice=voiceless), interp.median, na.rm=TRUE)
```

```
##   voice          x
## 1     0 0.06979543
## 2     1 0.39682540
```

```
t.test(posit, voiceless)
```

```
##
##  Welch Two Sample t-test
##
## data:  posit and voiceless
## t = -13.514, df = 10509, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1649442 -0.1231560
## sample estimates:
## mean of x mean of y
## 0.1788666 0.3229167
```

```
# by country:
  # aggregate(posit, by=list(voice=voiceless, cy=c7), mean, na.rm=TRUE)
  aggregate(posit, by=list(voice=voiceless, cy=c7), interp.median, na.rm=TRUE)
```

```
##    voice cy           x
## 1      0 AT  0.14912281
## 2      1 AT  0.25000000
## 3      0 BE  0.23148148
## 4      1 BE  0.43103448
## 5      0 CH  0.27142857
## 6      1 CH  0.36734694
## 7      0 ES  0.10465116
## 8      1 ES  0.08928571
## 9      0 IE  0.69811321
## 10     1 IE  0.52702703
## 11     0 NL  0.20204082
## 12     1 NL  0.22277228
## 13     0 UK -0.03658537
## 14     1 UK -0.52173913
```

```
# by actor
  table(voiceless, posit, gpactor1)
```

```
## , , gpactor1 = 0
##
##          posit
## voiceless  -1 -0.5   0 0.5   1
##         0 308  330 601 535 509
##         1 199  213 299 228 194
##
## , , gpactor1 = 1
##
##          posit
## voiceless  -1 -0.5   0 0.5   1
##         0  24   15  69 138 319
##         1  11    9  27  60 216
##
## , , gpactor1 = 2
##
##          posit
## voiceless  -1 -0.5   0 0.5   1
##         0  29   51 110 146 131
##         1  22   31  48  77  49
##
## , , gpactor1 = 3
##
##          posit
## voiceless  -1 -0.5   0 0.5   1
##         0  96   34   8  12   1
##         1  18   22   4   1   0
```

```
  t.test(voiceless[gpactor1==1], posit[gpactor1==1])
```

```
##
##  Welch Two Sample t-test
```

```
##
## data:  voiceless[gpactor1 == 1] and posit[gpactor1 == 1]
## t = -14.095, df = 2005.3, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.3554968 -0.2686530
## sample estimates:
## mean of x mean of y
## 0.3478261 0.6599010
```

```
t.test(voiceless[gpactor1==2], posit[gpactor1==2])
```

```
##
##  Welch Two Sample t-test
##
## data:  voiceless[gpactor1 == 2] and posit[gpactor1 == 2]
## t = 1.8438, df = 1497.7, p-value = 0.06541
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.003150426  0.101833224
## sample estimates:
## mean of x mean of y
## 0.3212045 0.2718631
```

```
# only negative claims:
table(voiceless[gpactor1==2 & posit==-1],   posit[gpactor1==2  & posit==-1])
```

```
##
##     -1
##   0 29
##   1 22
```

```
t.test(voiceless[gpactor1==2  & posit==-1], posit[gpactor1==2  & posit==-1])
```

```
##
##  Welch Two Sample t-test
##
## data:  voiceless[gpactor1 == 2 & posit == -1] and posit[gpactor1 == 2 & posit == -1]
## t = 20.436, df = 50, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.290690 1.572055
## sample estimates:
##  mean of x  mean of y
##  0.4313725 -1.0000000
```

```
t.test(voiceless[gpactor1==2  & posit<0],   posit[gpactor1==2  & posit<0])
```

```
##
##  Welch Two Sample t-test
##
## data:  voiceless[gpactor1 == 2 & posit < 0] and posit[gpactor1 == 2 & posit < 0]
## t = 23.049, df = 187.98, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.9915074 1.1771067
## sample estimates:
```

```
##  mean of x  mean of y
##  0.3984962 -0.6858108
```

```
  # positive claims:
  t.test(voiceless[gpactor1==2  & posit==1], posit[gpactor1==2  & posit==1])
```

```
##
##  Welch Two Sample t-test
##
## data:  voiceless[gpactor1 == 2 & posit == 1] and posit[gpactor1 == 2 & posit == 1]
## t = -21.876, df = 179, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7934269 -0.6621286
## sample estimates:
## mean of x mean of y
## 0.2722222 1.0000000
```

```
  t.test(voiceless[gpactor1==2  & posit>0],  posit[gpactor1==2  & posit>0])
```

```
##
##  Welch Two Sample t-test
##
## data:  voiceless[gpactor1 == 2 & posit > 0] and posit[gpactor1 == 2 & posit > 0]
## t = -15.726, df = 603.61, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4598577 -0.3577545
## sample estimates:
## mean of x mean of y
## 0.3126551 0.7214612
```

```
  t.test(voiceless[gpactor1==1  & posit==1], posit[gpactor1==1  & posit==1])
```

```
##
##  Welch Two Sample t-test
##
## data:  voiceless[gpactor1 == 1 & posit == 1] and posit[gpactor1 == 1 & posit == 1]
## t = -28.083, df = 534, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.6379708 -0.5545525
## sample estimates:
## mean of x mean of y
## 0.4037383 1.0000000
```

```
  t.test(voiceless[gpactor1==1  & posit>0],  posit[gpactor1==1  & posit>0])
```

```
##
##  Welch Two Sample t-test
##
## data:  voiceless[gpactor1 == 1 & posit > 0] and posit[gpactor1 == 1 & posit > 0]
## t = -25.463, df = 991.28, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5336127 -0.4572497
## sample estimates:
```

```
## mean of x mean of y
## 0.3765348 0.8719660
```

Anti-Immigrant Actors

```
  round(prop.table(table(antiact, voiceless), margin=1)*100,1)
```

```
##        voiceless
## antiact    0    1
##       0 67.4 32.6
##       1 76.7 23.3
```

Correlations between Group Size and Claims Share

Mentioned in the main text around table 1. The numbers are in table 1.

```
  setwd("C:/Users/ruedind/switchdrive/Politicization/Groups Paper/Analysis")
  # setwd("/home/didier/ownCloud/Politicization/Groups Paper/Anaysis")
  x <- read.csv("table1.csv")
  attach(x)
  # Pearson
    cor(data.frame(Muslims, Muslims.of.Immigrants, Claims))
```

```
##                          Muslims Muslims.of.Immigrants    Claims
## Muslims                1.0000000             0.9092813 0.4296048
## Muslims.of.Immigrants 0.9092813             1.0000000 0.3702721
## Claims                 0.4296048             0.3702721 1.0000000
```

```
    cor(EU.Immigrants, Claims.1)
```

```
## [1] 0.5895471
```

```
    cor(Asylum.Seekers, Claims.2)
```

```
## [1] -0.07732095
```

```
  # Spearman
    cor(data.frame(Muslims, Muslims.of.Immigrants, Claims), method="spearman")
```

```
##                           Muslims Muslims.of.Immigrants     Claims
## Muslims                1.00000000             0.8928571 0.03571429
## Muslims.of.Immigrants 0.89285714             1.0000000 0.14285714
## Claims                 0.03571429             0.1428571 1.00000000
```

```
    cor(EU.Immigrants, Claims.1, method="spearman")
```

```
## [1] 0.3928571
```

```
    cor(Asylum.Seekers, Claims.2, method="spearman")
```

```
## [1] 0.3214286
```

```
  # Kendall
    cor(data.frame(Muslims, Muslims.of.Immigrants, Claims), method="kendall")
```

```
##                          Muslims Muslims.of.Immigrants     Claims
## Muslims                1.00000000            0.80952381 0.04761905
## Muslims.of.Immigrants 0.80952381            1.00000000 0.04761905
## Claims                 0.04761905            0.04761905 1.00000000
```

```
    cor(EU.Immigrants, Claims.1, method="kendall")
```

```
## [1] 0.3333333
```

```
    cor(Asylum.Seekers, Claims.2, method="kendall")
```

```
## [1] 0.2380952
```

```
  # Pearson: Significant?
    cor.test(Muslims, Claims)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Muslims and Claims
## t = 1.0638, df = 5, p-value = 0.3361
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.4781399  0.8935757
## sample estimates:
##       cor
## 0.4296048
```

```
    cor.test(EU.Immigrants, Claims.1)
```

```
##
##  Pearson's product-moment correlation
##
## data:  EU.Immigrants and Claims.1
## t = 1.6321, df = 5, p-value = 0.1636
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2940651  0.9298057
## sample estimates:
##       cor
## 0.5895471
```

```
    cor.test(Asylum.Seekers, Claims.2)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Asylum.Seekers and Claims.2
## t = -0.17341, df = 5, p-value = 0.8691
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7846888  0.7175161
## sample estimates:
##         cor
## -0.07732095
```

```
  # Spearman
    cor.test(Muslims, Claims, method="spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  Muslims and Claims
## S = 54, p-value = 0.9635
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## 0.03571429
```

```
    cor.test(EU.Immigrants, Claims.1, method="spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  EU.Immigrants and Claims.1
## S = 34, p-value = 0.3956
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.3928571
```

```
    cor.test(Asylum.Seekers, Claims.2, method="spearman")
```

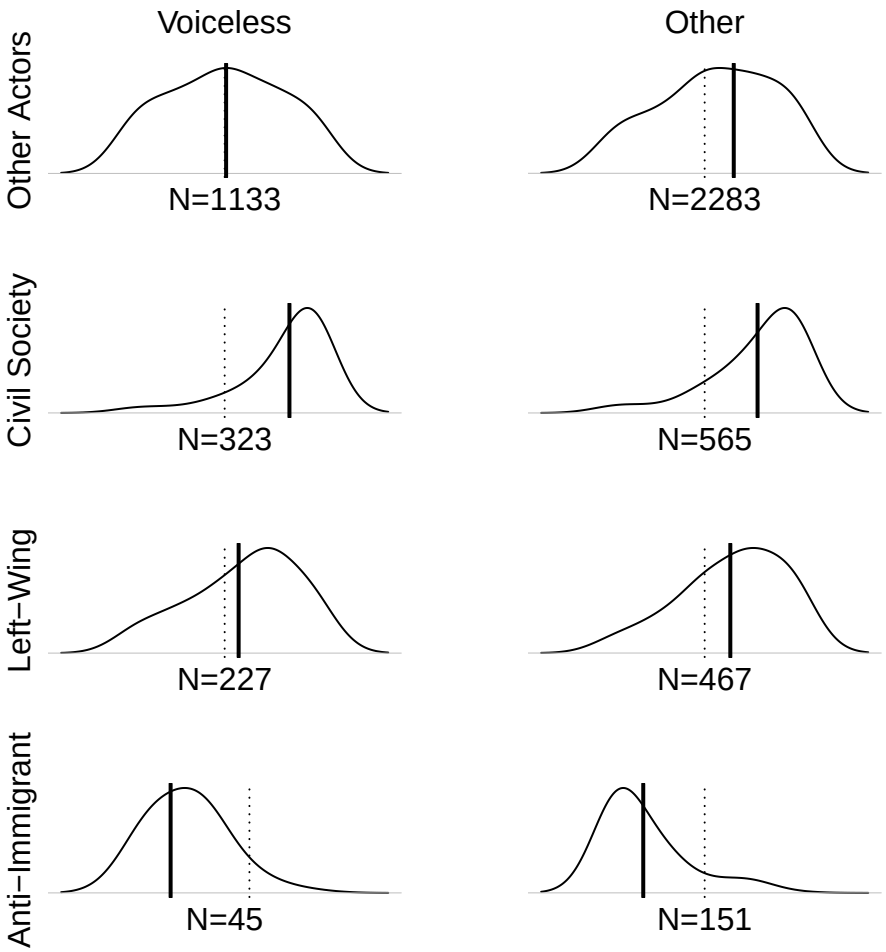
```
##
##  Spearman's rank correlation rho
##
## data:  Asylum.Seekers and Claims.2
## S = 38, p-value = 0.4976
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.3214286
```

Comparative Groups/Actor Plot

Figure 1 in the paper; below two alternatives with object evaluation rather than tone of the claim, and with different definitions of CSO.

```
# tiff(file="Figure1.tif", width = 8, height = 8, units = "in", pointsize = 12, compression = "lzw", bg
  xlabels <- c("Voiceless", "Other")
  xcodes  <- c(1, 0)
  ylabels <- c("Other Actors", "Civil Society", "Left-Wing", "Anti-Immigrant")
  ycodes  <- c(0, 1, 2, 3)
  len <- length(xlabels)
  dwn <- length(ylabels)
par(mfrow = c(4,2))          # 4 by 2 cells, labeling in margins
  par(mar=c(2,3,3,2))        # smaller borders
  for(j in 1:dwn) {          # repeat for each actor
    for(i in 1:len) {        # repeat for each immigrant group
      z <- table(posit[gpactor1==ycodes[j] & voiceless ==xcodes[i]]) # cross tab
      # table() does not give zero counts, so I re-create them here:
      z <- cbind(z["-1"],z["-0.5"],z["0"],z["0.5"],z["1"])  # joining, to get empty cells, too
      z[is.na(z)] <- 0                                      # replace empty cells (NA) with zero
```

```
      plot(density(posit[gpactor1==ycodes[j] & voiceless ==xcodes[i]], bw=0.3,
        na.rm=TRUE), axes=FALSE, ylab="", xlab="", main="")
      abline(v=interp.median(posit[gpactor1==ycodes[j] & voiceless ==xcodes[i]],
        na.rm=TRUE), lwd=2)
      abline(v=0, lty=3)
      mtext(paste("N",sum(z), sep="="),1, 0.5)     # give N
      if(j==1) mtext(xlabels[i],3, 1)              # labels in margins, with offset
      if(i==1) mtext(ylabels[j],2, .5)
    }
  }
```

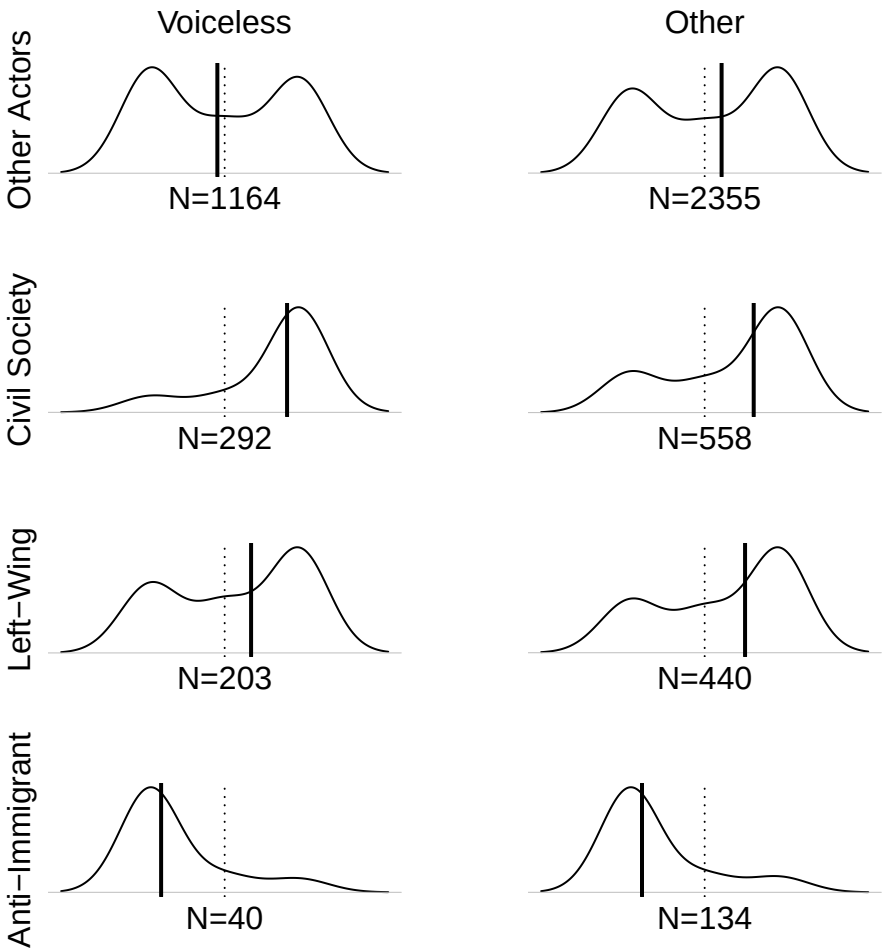



```
# dev.off()
```

Here's an alternative with object evaluation.

```
  xlabels <- c("Voiceless", "Other")
  xcodes  <- c(1, 0)
  ylabels <- c("Other Actors", "Civil Society", "Left-Wing", "Anti-Immigrant")
  ycodes  <- c(0, 1, 2, 3)
  len <- length(xlabels)
  dwn <- length(ylabels)
par(mfrow = c(4,2))          # 4 by 2 cells, labeling in margins
  par(mar=c(2,3,3,2))        # smaller borders
  for(j in 1:dwn) {          # repeat for each actor
```

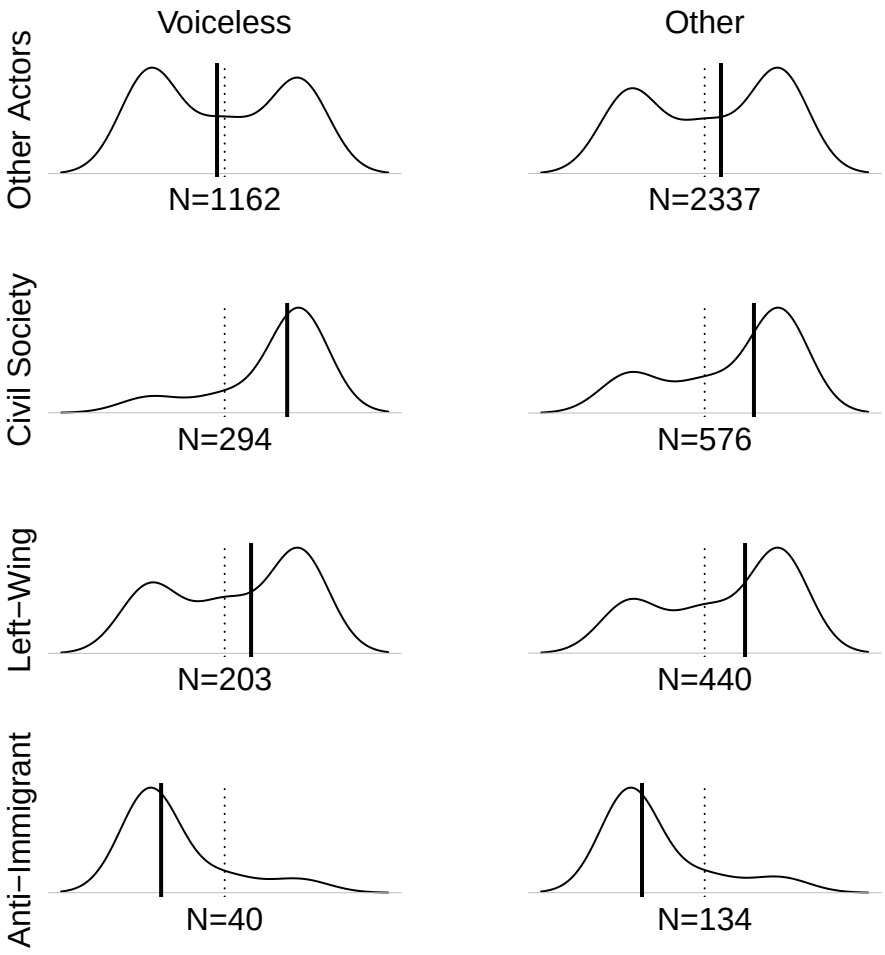
```
  for(i in 1:len) {       # repeat for each immigrant group
    z <- table(objeval1[gpactor1==ycodes[j] & voiceless ==xcodes[i]]) # cross tab
    # table() does not give zero counts, so I re-create them here:
    z <- cbind(z["-1"],z["0"],z["1"])             # joining, to get empty cells, too
    z[is.na(z)] <- 0                              # replace empty cells (NA) with zero
      plot(density(objeval1[gpactor1==ycodes[j] & voiceless ==xcodes[i]],
          bw=0.4, na.rm=TRUE), axes=FALSE, ylab="", xlab="", main="")
          # kernel densities, bw set to smooth out
      abline(v=interp.median(objeval1[gpactor1==ycodes[j] & voiceless ==xcodes[i]],
          na.rm=TRUE), lwd=2)
      abline(v=0, lty=3)                            # interpolated median; zero line
      mtext(paste("N",sum(z), sep="="),1, 0.5)      # give N
      if(j==1) mtext(xlabels[i],3, 1)     # labels in margins (only edge), with offset
      if(i==1) mtext(ylabels[j],2, .5)    # labels in margins (only edge), smaller offset
  }
}
```



Here is an alternative version not in the paper. It uses the alternative definition of CSO specified at the beginning of this document: CSO2 definition includes migrant actors as CSO:

```
xlabels <- c("Voiceless", "Other")
xcodes  <- c(1, 0)
ylabels <- c("Other Actors", "Civil Society", "Left-Wing", "Anti-Immigrant")
```

```
  ycodes  <- c(0, 1, 2, 3)
  len <- length(xlabels)
  dwn <- length(ylabels)
par(mfrow = c(4,2))          # 4 by 2 cells, labeling in margins
  par(mar=c(2,3,3,2))        # smaller borders
  for(j in 1:dwn) {          # repeat for each actor
    for(i in 1:len) {        # repeat for each immigrant group
      z <- table(objeval1[gpactor2==ycodes[j] & voiceless ==xcodes[i]]) # cross tab
      # table() does not give zero counts, so I re-create them here:
      z <- cbind(z["-1"],z["0"],z["1"])              # joining, to get empty cells, too
      z[is.na(z)] <- 0                               # replace empty cells (NA) with zero
        plot(density(objeval1[gpactor2==ycodes[j] & voiceless ==xcodes[i]], bw=0.4,
          na.rm=TRUE), axes=FALSE, ylab="", xlab="", main="")
        abline(v=interp.median(objeval1[gpactor2==ycodes[j] & voiceless ==xcodes[i]],
          na.rm=TRUE), lwd=2)
        abline(v=0, lty=3)
        mtext(paste("N",sum(z), sep="="),1, 0.5)     # give N
        if(j==1) mtext(xlabels[i],3, 1)              # labels in margins, with offset
        if(i==1) mtext(ylabels[j],2, .5)
    }
  }
```



Justifications (Frames)

```
round(prop.table(table(voiceless, frame1d), margin=1)*100,0)
```

```
##          frame1d
## voiceless  1  2  3
##         0 55 16 28
##         1 62  2 35
```

```
chisq.test(table(voiceless, frame1d))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(voiceless, frame1d)
## X-squared = 177, df = 2, p-value < 2.2e-16
```

```
sapply(cy, function(x) round(prop.table(table(voiceless[c7==x], frame1d[c7==x]),
  margin=1)*100,0))
```

```
##      AT BE ES IE NL CH UK
## [1,] 62 40 68 66 62 53 43
## [2,] 54 51 76 58 74 61 66
## [3,] 11 17  9 14 20 13 27
## [4,]  3  3  1  5  2  1  3
## [5,] 27 43 23 20 18 33 29
## [6,] 44 46 23 37 24 39 31
```

```
  # read table as follows: [1,] is other, 1 (=instrumental),
                           # [2,] is voicless, 1;
                           # [3,] is other, 2 (identity),
                           # [4,] is voiceless, 3.;
```

Aggregate Level

```
    detach(core.claims) # manipulating data.frame
  # get combined.X.RData in:
    load("C:/Users/ruedind/switchdrive/Politicization/Aggregated SOM Claims Data/combined.17.RData")
    # load("/home/didier/ownCloud/Politicization/Aggregated SOM Claims Data/combined.17.RData")
  # create case variable
    core.claims$case <- paste(core.claims$c7, core.claims$year, sep="")
  # merge with aggregate data
    #kCivic
    for(i in unique(core.claims$case)){
      core.claims$Civic[core.claims$case == i] <- manual$kCivic[manual$case == i]
    }
    #kPluralism
    for(i in unique(core.claims$case)){
      core.claims$Pluralism[core.claims$case == i] <- manual$kPluralism[manual$case == i]
    }
    # collapse into binary variable at 50 (theoretical midpoint, close to actual mean)
    core.claims$CivicBinary[core.claims$Civic <= 50]         <- "Ethnic"
    core.claims$CivicBinary[core.claims$Civic >  50]         <- "Civic"
```

```
  core.claims$PluralismBinary[core.claims$Pluralism <= 50] <- "Monism"
  core.claims$PluralismBinary[core.claims$Pluralism >  50] <- "Pluralism"
attach(core.claims) # re-attaching data.frame
```

Citizenship Regime Scores by Country

Appendix A1

```
cases <- unique(manual$country)
meanc <- sapply(cases, function(cy) mean(manual$kCivic[manual$country==cy]))
minc <- sapply(cases, function(cy) min(manual$kCivic[manual$country==cy]))
maxc <- sapply(cases, function(cy) max(manual$kCivic[manual$country==cy]))
meanp <- sapply(cases, function(cy) mean(manual$kPluralism[manual$country==cy]))
minp <- sapply(cases, function(cy) min(manual$kPluralism[manual$country==cy]))
maxp <- sapply(cases, function(cy) max(manual$kPluralism[manual$country==cy]))
tab <- rbind(meanc, minc, maxc, meanp, minp, maxp)
colnames(tab) <- cases
rownames(tab) <- c("Civic mean", "min", "max", "Pluralism mean", "min", "max")
round(tab)
```

```
##                AT BE CH ES IE NL UK
## Civic mean     34 61 43 41 59 73 62
## min            29 51 40 26 40 72 56
## max            41 77 47 52 76 76 75
## Pluralism mean 36 63 46 66 43 80 50
## min            19 55 37 57 33 61 41
## max            51 75 49 72 60 85 52
```

Not just country effects

```
table(core.claims$PluralismBinary, core.claims$c7)
```

```
##
##                AT   BE   CH   ES   IE   NL   UK
##   Monism      843    0 1005    0  415    0  516
##   Pluralism   172 1093    0 1058  199 1319  409
```

```
table(core.claims$CivicBinary, core.claims$c7)
```

```
##
##             AT   BE   CH   ES   IE   NL   UK
##   Civic      0 1093    0   25  509 1319  925
##   Ethnic  1015    0 1005 1033  105    0    0
```

Frames

```
# more instrumental claims in civic contexts?
round(prop.table(table(CivicBinary, frame1d), margin=1)*100,1)
```

```
##             frame1d
## CivicBinary    1    2    3
##      Civic  54.9 15.7 29.4
```

```
##      Ethnic 62.9  8.1 29.0
```

```
  chisq.test(table(CivicBinary, frame1d))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(CivicBinary, frame1d)
## X-squared = 68.19, df = 2, p-value = 1.559e-15
```

```
  round(prop.table(table(PluralismBinary, frame1d), margin=1)*100,1)
```

```
##                frame1d
## PluralismBinary    1    2    3
##       Monism    55.6 12.7 31.7
##       Pluralism 59.6 12.6 27.9
```

```
  chisq.test(table(PluralismBinary, frame1d))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(PluralismBinary, frame1d)
## X-squared = 8.9947, df = 2, p-value = 0.01114
```

```
  # comparing only moral claims
  core.claims$frame1dbin <- core.claims$frame1d
  core.claims$frame1dbin[core.claims$frame1d %in% 1:2] <- 1
  chisq.test(table(core.claims$CivicBinary, core.claims$frame1dbin))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(core.claims$CivicBinary, core.claims$frame1dbin)
## X-squared = 0.08043, df = 1, p-value = 0.7767
```

Number of Claims

```
# more claims about voiceless in ethnic contexts?
  round(prop.table(table(CivicBinary, voiceless), margin=1)*100,1)
```

```
##            voiceless
## CivicBinary    0    1
##      Civic  71.1 28.9
##      Ethnic 64.0 36.0
```

```
  chisq.test(table(CivicBinary, voiceless))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(CivicBinary, voiceless)
## X-squared = 34.741, df = 1, p-value = 3.767e-09
```

```
# more claims about voiceless in monistic contexts?
  round(prop.table(table(PluralismBinary, voiceless), margin=1)*100,1)
```

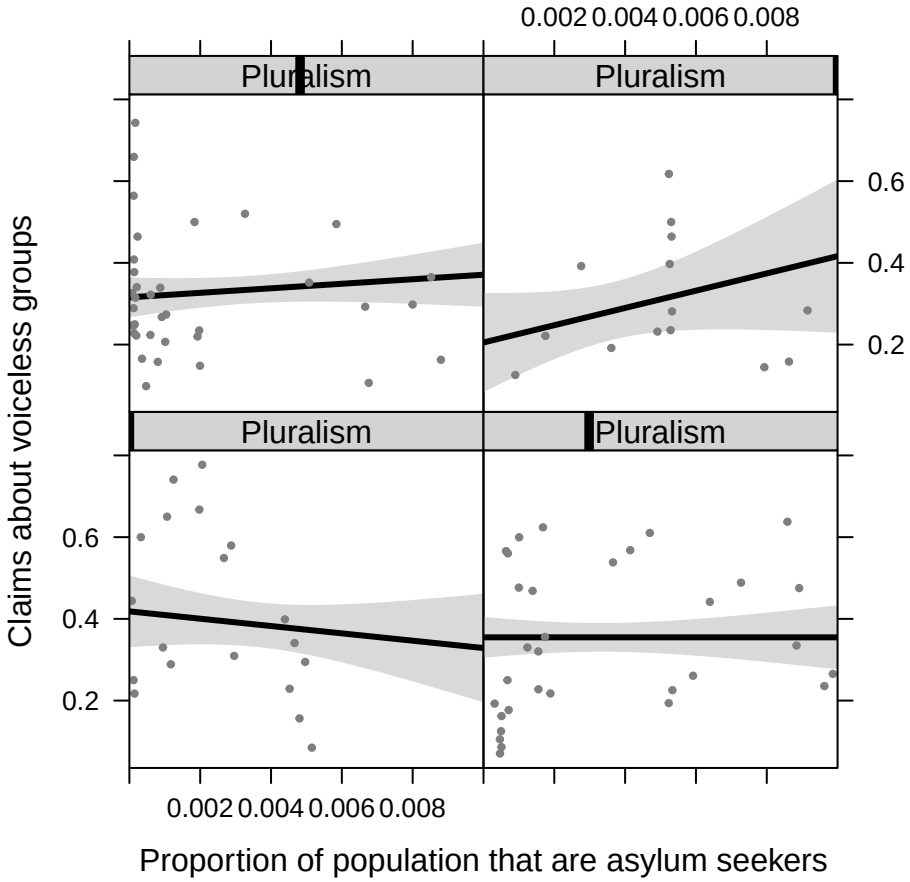
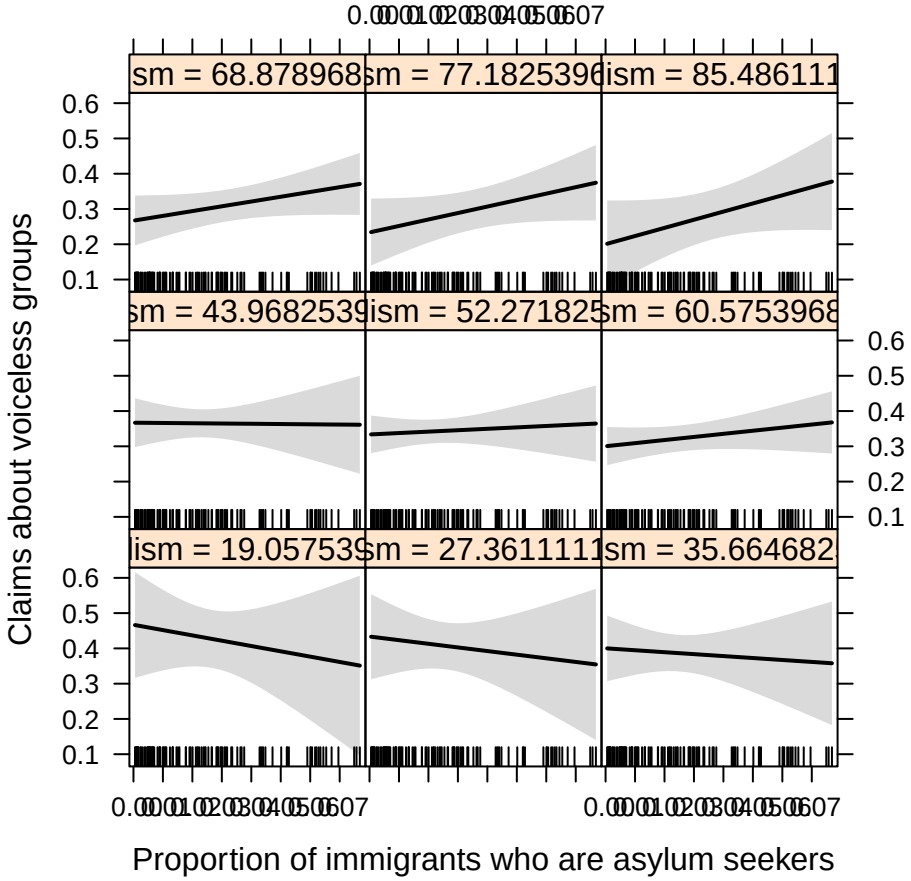
```
##                 voiceless
## PluralismBinary    0    1
##       Monism    65.6 34.4
##       Pluralism 69.3 30.7
```

```
  chisq.test(table(PluralismBinary, voiceless))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(PluralismBinary, voiceless)
## X-squared = 9.2116, df = 1, p-value = 0.002405
```

Regression Model and Interaction

```
  manual$Pluralism <- manual$kPluralism # prettier plot
# asylum seekers as proportion of the population
  a3 <- lm(voiceClaims ~ asylumprop + Pluralism, data=manual)
  a4 <- lm(voiceClaims ~ asylumprop * Pluralism, data=manual)
  # tiff(file="Figure2.tif", width = 8, height = 8, units = "in", pointsize = 12, compression = "lzw",
  visreg(a4, "asylumprop", by="Pluralism", breaks=4, ylab="Claims about voiceless groups", xlab="Propor
```



```
  # dev.off()
# asylum seekers as proportion of immigrants
  a5 <- lm(voiceClaims ~ asylumimmigrants + Pluralism, data=manual)
  a6 <- lm(voiceClaims ~ asylumimmigrants * Pluralism, data=manual)
  plot(allEffects(a6, xlevels=9), main="", ylab="Claims about voiceless groups",
    xlab="Proportion of immigrants who are asylum seekers")
```