# MLE: Lab 1

*Shahryar Minhas*

*January 14, 2014*

## Setting up your workspace

```
# Start with a clean workspace
rm(list=ls())

# Load libraries
library(foreign)
library(xtable)
library(apsrtable)
library(arm)
library(ggplot2)

# Functions that I use frequently
char = function(x){ as.character(x) }
num = function(x){ as.numeric(char(x)) }

# Relevant paths
# Mac
lab1Path='~/Dropbox/Duke/Spring 2015/PS 733/lab1'
lab1Path=file.path('~','Dropbox', 'Duke', 'Spring 2015', 'PS 733', 'lab1')
```

## Choosing a work to replicate

Pick a work to replicate strategically:

- Published within the last five years
- Related to your research interests and that will help you progress in your career
- Aim for one that has been done in at least a second tier journal like JPR or ISQ

## Presenting summary statistics for replication

When I took this class I replicated a paper by Salehyan, Gleditsch, and Cunningham that attempted to provide an explanation for why some rebel groups receive more assistance from foreign governments than others. Data for this paper can be downloaded from here.

```
# Load data
sgcPath=paste0(lab1Path, '/sgc_io_data.txt')
sgc = read.table(sgcPath, sep="\t",header=T,na=".",quote="")

# Preview dataset
dim(sgc)
sgc[1:3,]
```

```r
# Transforming ordinal rebel strength variable into binaries
unique(sgc$rs)
sgc$weak=0; sgc$weak[sgc$rs=='weak']=1
sgc$parity=0; sgc$parity[sgc$rs=='parity']=1
sgc$strong=0; sgc$strong[sgc$rs=='strong']=1
sgc$muchWeaker=0; sgc$muchWeaker[sgc$rs=='much weaker']=1

# descriptive statistics for variables of interest
vars=c( paste0('supp',1:3), # DVs
  'weak', 'parity', 'strong', 'muchWeaker',
  'cl', 'gs', 'tk', 'tc', 'noactors', 'riv' )

summary(sgc[,vars])

# summary may not provide the output you want
# I want following stats: N, mean, sd, min, max for each var
getStats=function(x){
  n=length(x[!is.na(x)])
  mu=mean(x, na.rm=TRUE)
  sd=sd(x, na.rm=TRUE)
  minMax=quantile(x, probs=c(0,1), na.rm=TRUE)
  return( c( n, mu, sd, minMax ) )
}

sgcStats=t( apply(sgc[,vars], 2, getStats) )
colnames(sgcStats)=c('N', 'Mean', 'Std. Dev.', 'Min', 'Max')
xtable(sgcStats)
```

Below is a simple tabular display of the results.

| | N | Mean | Std. Dev. | Min | Max |
|---|---|---|---|---|---|
| supp1 | 370.00 | 0.43 | 0.50 | 0.00 | 1.00 |
| supp2 | 370.00 | 0.35 | 0.48 | 0.00 | 1.00 |
| supp3 | 342.00 | 0.38 | 0.49 | 0.00 | 1.00 |
| weak | 403.00 | 0.46 | 0.50 | 0.00 | 1.00 |
| parity | 403.00 | 0.08 | 0.28 | 0.00 | 1.00 |
| strong | 403.00 | 0.07 | 0.25 | 0.00 | 1.00 |
| muchWeaker | 403.00 | 0.39 | 0.49 | 0.00 | 1.00 |
| cl | 343.00 | 0.26 | 0.44 | 0.00 | 1.00 |
| gs | 366.00 | 0.49 | 0.50 | 0.00 | 1.00 |
| tk | 363.00 | 0.31 | 0.46 | 0.00 | 1.00 |
| tc | 403.00 | 0.29 | 0.45 | 0.00 | 1.00 |
| noactors | 403.00 | 2.52 | 2.54 | 1.00 | 13.00 |
| riv | 403.00 | 0.52 | 0.50 | 0.00 | 1.00 |

Slightly nicer looking table.

| Variable | N | Mean | Std. Dev. | Min | Max |
|---|---|---|---|---|---|
| *Dependent Variables: Types of External Government Support* | | | | | |
| Alleged and Explicit Support | 370 | 43% | 50% | 0 | 1 |
| Explicit Support (Alleged set to 0) | 370 | 35% | 48% | 0 | 1 |
| Explicit Support (Alleged set to missing) | 342 | 38% | 49% | 0 | 1 |
| *Independent Variables: Rebel Characteristics* | | | | | |
| Strong Central Command | 343 | 26% | 44% | 0 | 1 |
| Transnational Constituency | 363 | 31% | 46% | 0 | 1 |
| Territorial Control | 403 | 29% | 45% | 0 | 1 |
| Rebels Much Weaker | 401 | 39% | 49% | 0 | 1 |
| Rebels Weak or at Parity | 401 | 54% | 50% | 0 | 1 |
| Rebels Strong | 401 | 7% | 25% | 0 | 1 |
| No. of Actors | 403 | 2.52 | 2.54 | 1 | 13 |
| *Independent Variables: State Characteristics* | | | | | |
| Govt. Received Support | 366 | 49% | 50% | 0 | 1 |
| Rivalry | 403 | 52% | 50% | 0 | 1 |

Can also use graphics (Mike prefers visualizations where possible).

```
supp1Cnt=table(sgc$supp1)
barplot(supp1Cnt, xlab='Alleged and Explicit Support',
        names.arg=c('No Support', 'Support'))
```

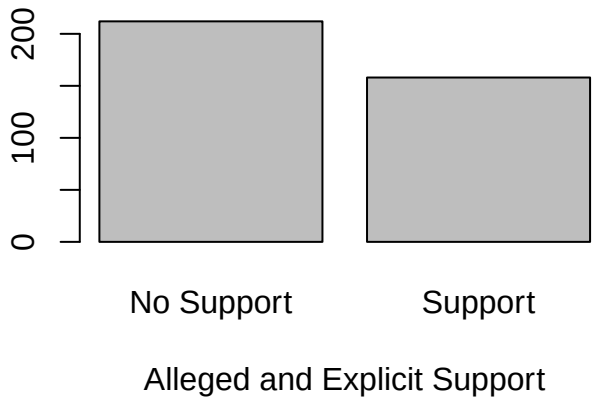



Figure 1: supp1 barplot

```
# Plot efficiently
plotVars=c(vars[1:3], 'rs', vars[8:length(vars)])
```

```
par(mfrow=c(1,3))
dvTitles=c('Alleged and Explicit Support', 'Explicit Support (Alleged=0)',
           'Explicit Support (Alleged=NA)')
for(ii in 1:3){
  cnt=table( sgc[, plotVars[ii] ] )
  barplot(cnt, xlab=dvTitles[ii], names.arg=c('No Support', 'Support'))
}
```

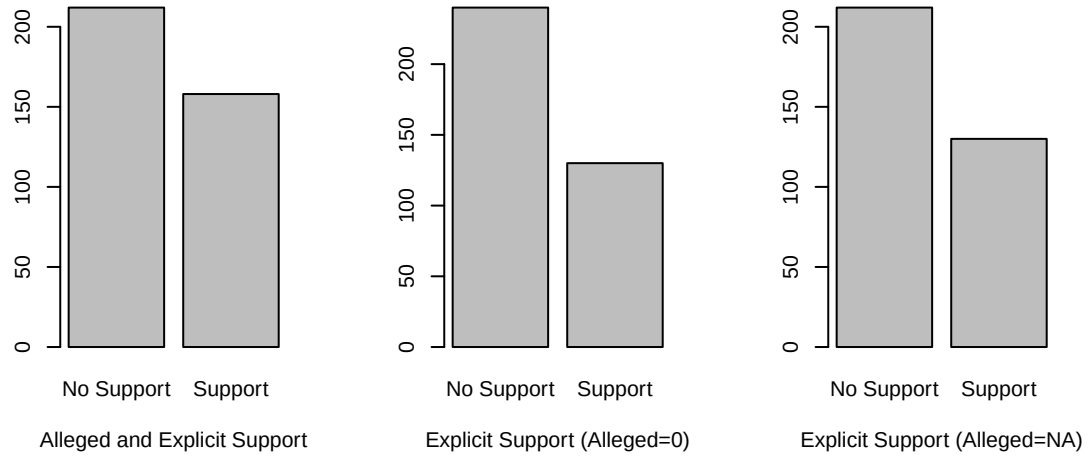



Figure 2: barplot of DVs

One more example...

```
par(mfrow=c(1,2))
hist(sgc$noactors, main='')
plot(density(sgc$noactors), main='')
```

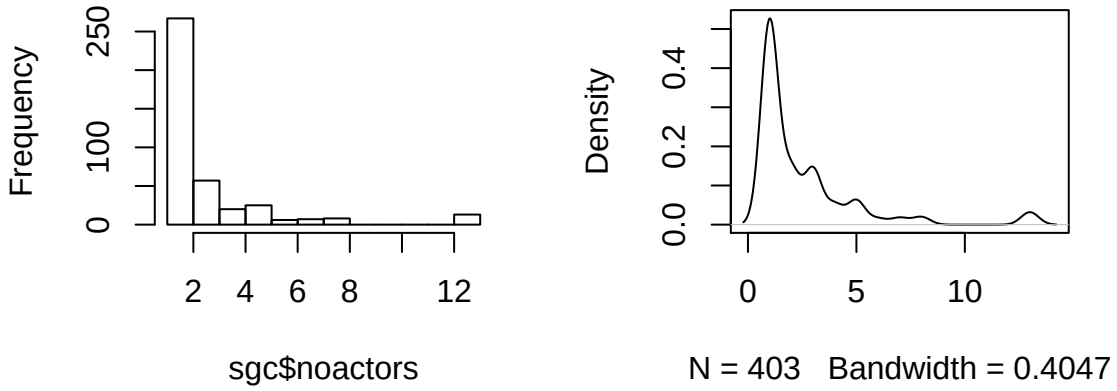
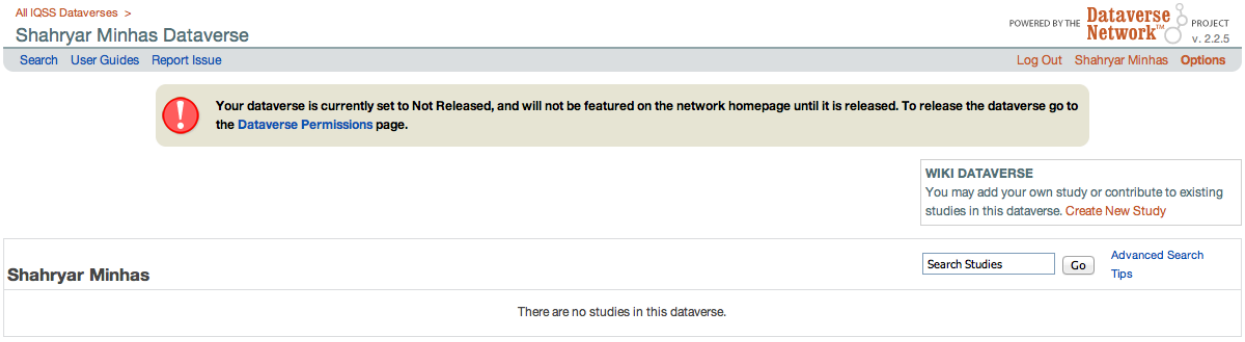



Figure 3: histogram/density of no actors

## Proof that you set up your dataverse



### Muller & Seligson Data

Download `Msrepl87.asc` from Mike's dataverse.

```
# Load Data
msrepPath=paste0(lab1Path, "/Msrepl87.asc")
msrep = read.table(msrepPath, header=TRUE)

# Peak at data
msrep[1:3, ]
```

Some help with replicating figure 2

```
# Var transformation as specified in M & S
# deaths variable
msrep$deaths75_2 <- (msrep$deaths75/msrep$pop75)
msrep$deaths75_2 <- msrep$deaths75_2 + 1
msrep$deaths75_2[msrep$deaths75_2 >= 50] <- 50
msrep$deaths75ln2 <- log(msrep$deaths75_2)

# Note that M & S drop two countries, read their paper and figure out which
# ones they drop

# standardizing measures for chart
msrep$deaths75ln2.std <- scale(msrep$deaths75ln2)
msrep$upper20.std <- scale(msrep$upper20)

#plotting figure
plot(msrep$upper20.std, msrep$deaths75ln2.std,
     xlim = c(-3, 3), ylim = c(-3, 3), axes = FALSE,
     ylab="ln Deaths from Political Violence per 1m, 1973 - 77",
      xlab="Upper 20% Income Share, ca. 1970",
     main = "Figure 2. Plot of Standardized \n Death Rates from Political Violence \n by Standardized In
axis(side = 1, at = c(-3:3), tck = 0.03, las = 1)
axis(side = 2, at = c(-3:3), tck = 0.03)
axis(side = 3, at = c(-3:3), tck = 0.03, labels = FALSE)
axis(side = 4, at = c(-3:3), tck = 0.03, labels = FALSE)
text(-1.2, -2.1, "Y = -1.63 + 0.056(X)")
text(-1.8, -2.6, expression(paste(R[a]^2, " = 0.14", sep = "")))
box()

#finding highlighted values in data
BRZL <- which(msrep$country == "Brazil")

#putting in circles around points
points(msrep$upper20.std[BRZL],msrep$deaths75ln2.std[BRZL], cex = 1.5)

#positioning text lables
text(msrep$upper20.std[BRZL], msrep$deaths75ln2.std[BRZL] +
  .01, "BRZL", cex = .65, pos = 3)
```

Some help with replicating table 1.1

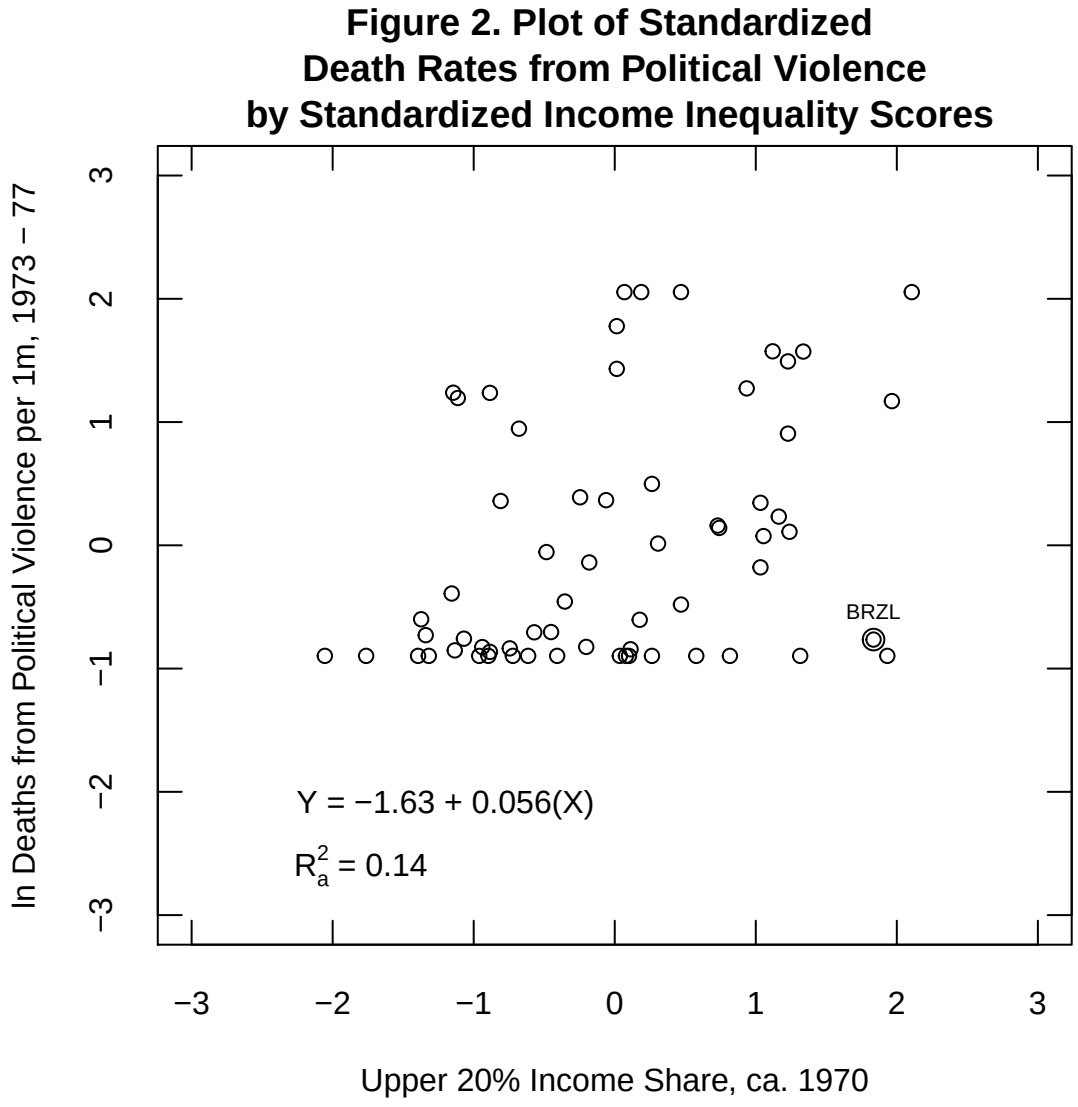



Figure 4: Muller and Seligson Figure 2

```r
# Read through the text carefully for other var transformations before
# running regression.
# A few hints:
# M & S drop another country before moving to the regression
# Read footnote 9 in the paper and make necessary changes

# Running a linear regression
ivs=c('upper20', 'energypc', 'intensep', 'sanctions70', 'sanctions75', 'deaths70')
olsForm=formula(
  paste0('deaths75ln2 ~ ',
         paste(ivs, collapse=' + ') )
  )
mod1 = lm(olsForm, data=msrep)

# View model results
summary(mod1)
```

```
##
## Call:
## lm(formula = olsForm, data = msrep)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2602 -0.6762 -0.0666  0.5449  2.8176
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.01e+00   1.17e+00   -0.86    0.392
## upper20      4.12e-02   2.09e-02    1.97    0.054 .
## energypc    -8.58e-05   8.56e-05   -1.00    0.320
## intensep     1.16e+00   4.63e-01    2.50    0.015 *
## sanctions70  3.32e-03   3.94e-03    0.84    0.403
## sanctions75  3.46e-04   2.06e-03    0.17    0.867
## deaths70     1.12e-06   4.42e-06    0.25    0.802
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.13 on 55 degrees of freedom
##   (33 observations deleted due to missingness)
## Multiple R-squared:  0.336,  Adjusted R-squared:  0.264
## F-statistic: 4.65 on 6 and 55 DF,  p-value: 0.000691
```

```r
apsrtable(mod1)
```

| | Model 1 |
|---|---|
| (Intercept) | −1.01 |
| | (1.17) |
| upper20 | 0.04 |
| | (0.02) |
| energypc | −0.00 |
| | (0.00) |
| intensep | 1.16* |
| | (0.46) |
| sanctions70 | 0.00 |
| | (0.00) |
| sanctions75 | 0.00 |
| | (0.00) |
| deaths70 | 0.00 |
| | (0.00) |
| $N$ | 62 |
| $R^2$ | 0.34 |
| adj. $R^2$ | 0.26 |
| Resid. sd | 1.13 |

Standard errors in parentheses

* indicates significance at $p < 0.05$

## Visualizing regression results

Some consider it easier to quickly look at regression results by using coefficient plots rather than tables.

```
coefplot(mod1)
```

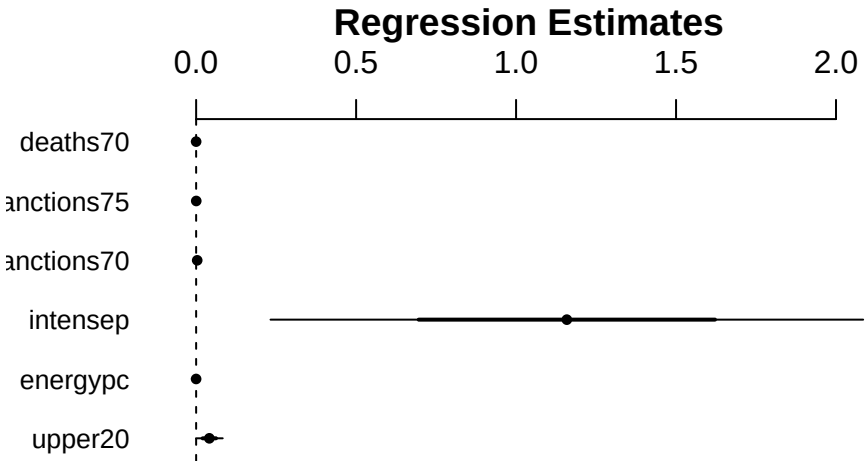



Figure 5: Ugly coefficient plot

## Simulations to construct an experiment

First we set up scenarios

```r
# Scenario 1: High intensity of separatism...other vars at central tendency
# Scenario 2: Low intensity of separatism...other vars at central tendency
means=apply(msrep[,ivs], 2, function(x){ mean(x, na.rm=TRUE) })
minMaxSep=quantile(msrep[,ivs[3]], probs=c(0,1), na.rm=TRUE)

scens=rbind(c(1, means[1:2], minMaxSep[1], means[4:length(means)]),
            c(1, means[1:2], minMaxSep[2], means[4:length(means)]) )

# Simulate additional parameter estimates from multivariate normal
draws = mvrnorm(n=1000, coef(mod1), vcov(mod1))

# Get predicted values using matrix multiplication
preds = draws %*% t(scens)

#plotting sim results
plot(density(preds[,1]), col = "black", bty = "n",
     las = 1, xlim=c(0, 4), lwd = 3, main='',
     xlab = "Logged Average Deaths per Million")
lines(density(preds[,2]), col = "grey", lwd = 3)
legend(x=.65,y=1,legend=c("Low Separatism"),
       text.col=c("black"),bty="n", cex = 0.75)
legend(x=1.7,y=0.8,legend=c("High Separatism"),
       text.col=c("grey"),bty="n", cex = 0.75)
```

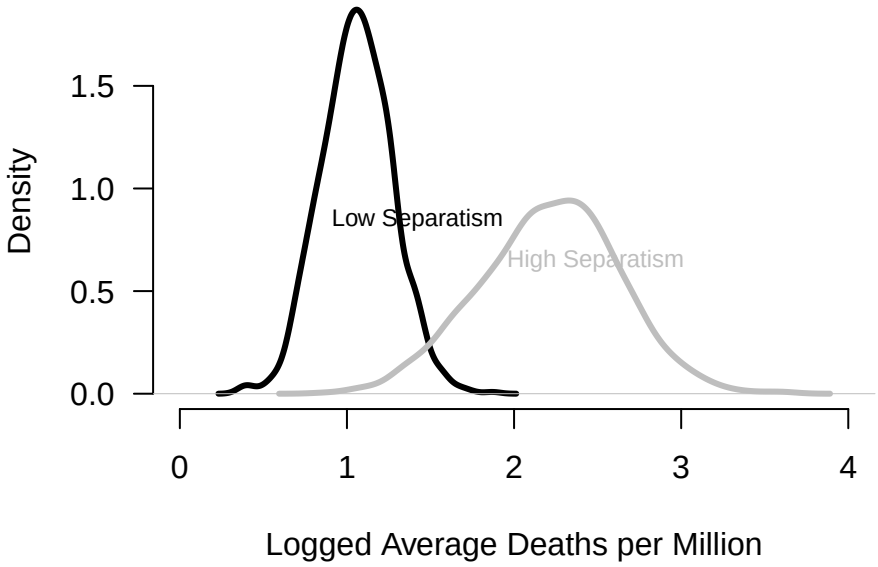



Figure 6: Quick and dirty simulation