Replication: The Structure of Inequality and the Politics of Redistribution

Filippo Teoldi, Zara Riaz and Julian Gerez

October 23rd, 2018

```
library('DeclareDesign')
```

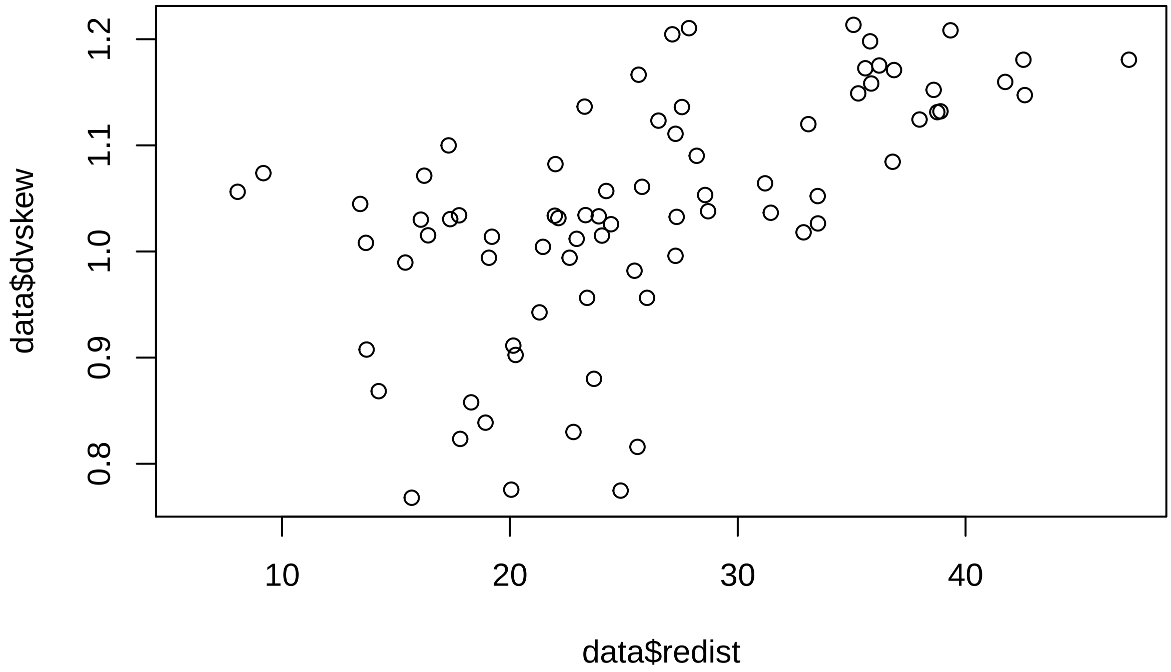
```
## Warning: package 'DeclareDesign' was built under R version 3.5.1

## Loading required package: randomizr

## Warning: package 'randomizr' was built under R version 3.5.1

## Loading required package: fabricatr

## Warning: package 'fabricatr' was built under R version 3.5.1

## Loading required package: estimatr

## Warning: package 'estimatr' was built under R version 3.5.1
```

```
library('knitr')
```

```
## Warning: package 'knitr' was built under R version 3.5.1
```

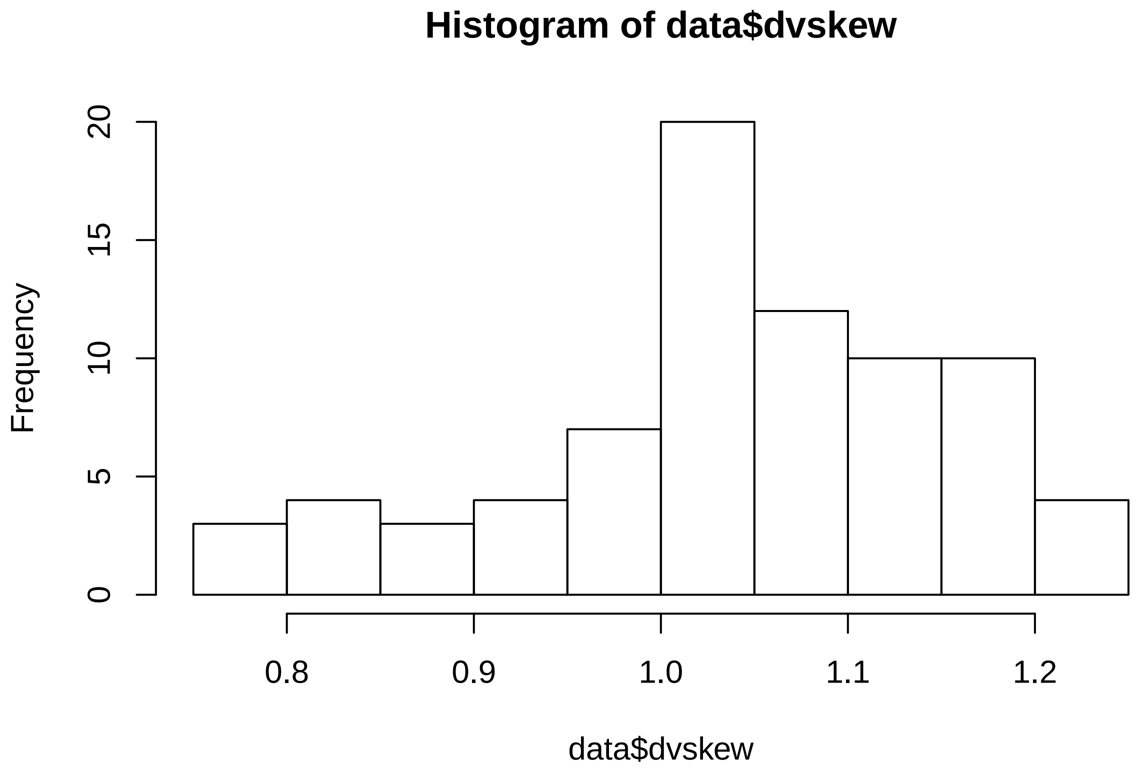
Model

```
load("redistsample.Rdata")
data <- redistsample

# Plot main relationship of interest:
plot(data$redist, data$dvskew)
```

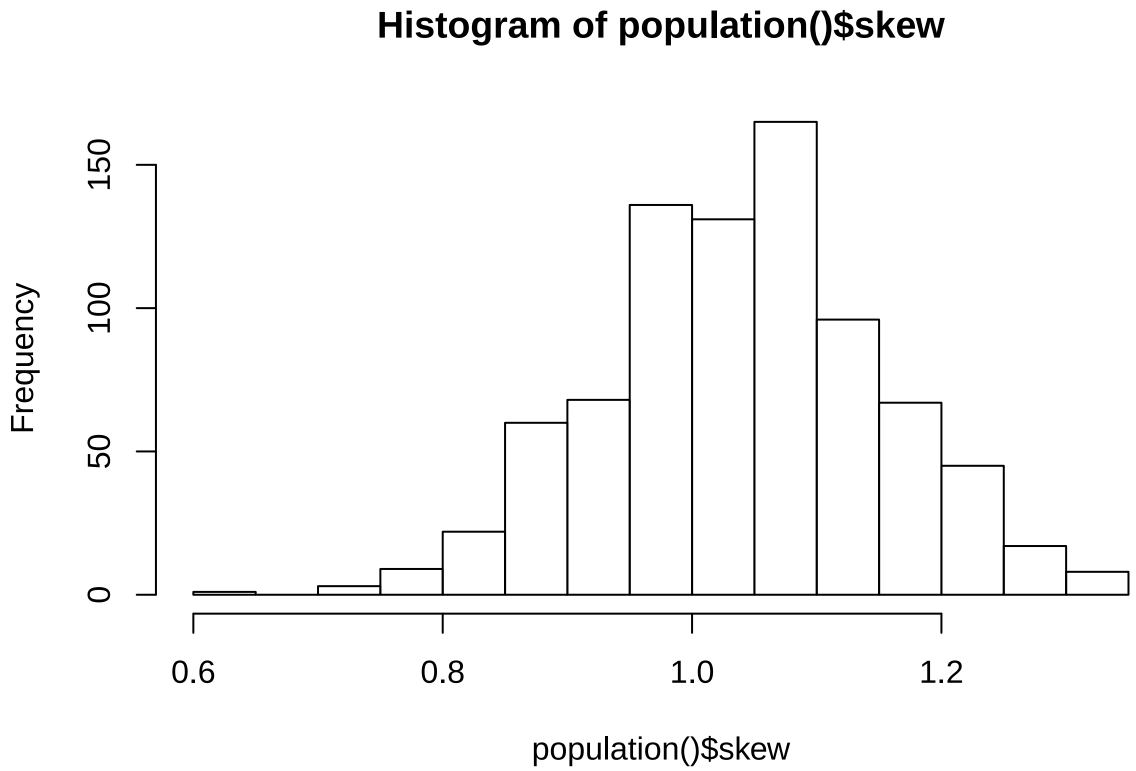


Population

```
population <- declare_population(
  country = add_level(
    N = 18, country_fe = rnorm(N)),
  year = add_level(
    N = 46, t = 1:N, nest = FALSE),
  obs = cross_levels(
    by = fabricatr::join(country, year),
    skew = rnorm(n = N, mean=1.039, sd=0.113),
    error_i = country_fe + rnorm(N)/2
  )
)

hist(data$dvskew)
```



```
hist(population()$skew)
```



```r
library('panelAR')
```

```
## Warning: package 'panelAR' was built under R version 3.5.1
```

```r
# Get parameters

modelX <- panelAR(redist ~ dvratio9010 + dvskew + as.factor(id),
                data=redistsample, panelVar='id', timeVar='time',
                autoCorr='ar1', panelCorrMethod='pcse',rho.na.rm=TRUE,
                panel.weight='t-1', bound.rho=TRUE)
summary(modelX)
```

```
##
## Panel Regression with AR(1) Prais-Winsten correction and panel-corrected standard errors
##
## Unbalanced Panel Design:
##  Total obs.:       77 Avg obs. per panel 5.1333
##  Number of panels: 15 Max obs. per panel 10
##  Number of times:  10 Min obs. per panel 1
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -4.6646     8.6392  -0.540 0.591242
## dvratio9010         1.3439     1.5360   0.875 0.385089
## dvskew             24.4739     7.5166   3.256 0.001860 **
## as.factor(id)3   12.3092     1.3360   9.214 4.32e-13 ***
## as.factor(id)4   -0.0509     2.7927  -0.018 0.985518
## as.factor(id)5   11.0080     2.1338   5.159 2.95e-06 ***
## as.factor(id)6    9.0069     1.9432   4.635 1.97e-05 ***
## as.factor(id)7   -2.6626     1.2938  -2.058 0.043947 *
## as.factor(id)8   -1.6262     0.9011  -1.805 0.076137 .
## as.factor(id)9    0.6049     2.1973   0.275 0.784038
## as.factor(id)12   5.9046     1.6921   3.490 0.000913 ***
## as.factor(id)14   7.9706     1.7490   4.557 2.60e-05 ***
## as.factor(id)15  11.9357     2.3695   5.037 4.62e-06 ***
## as.factor(id)16 -12.8997     1.5345  -8.406 9.96e-12 ***
## as.factor(id)17  -2.1192     1.3775  -1.538 0.129196
## as.factor(id)18  -9.3785     2.2897  -4.096 0.000128 ***
## as.factor(id)20 -13.1480     2.2069  -5.958 1.45e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.8874
## Wald statistic: 63453.0899, Pr(>Chisq(16)): 0
```

```
library(lmtest)
```

```
## Warning: package 'lmtest' was built under R version 3.5.1
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 3.5.1
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
results<- coeftest(modelX) #storing coefficients from model
results
```

```
##
## t test of coefficients:
##
##                   Estimate Std. Error t value  Pr(>|t|)
## (Intercept)      -4.664558   8.639148 -0.5399 0.5912419
## dvratio9010       1.343944   1.536021  0.8750 0.3850891
## dvskew           24.473923   7.516622  3.2560 0.0018604 **
## as.factor(id)3   12.309190   1.335970  9.2137 4.321e-13 ***
## as.factor(id)4   -0.050905   2.792739 -0.0182 0.9855178
## as.factor(id)5   11.007985   2.133845  5.1588 2.950e-06 ***
## as.factor(id)6    9.006874   1.943244  4.6350 1.974e-05 ***
## as.factor(id)7   -2.662586   1.293803 -2.0580 0.0439466 *
## as.factor(id)8   -1.626245   0.901105 -1.8047 0.0761369 .
## as.factor(id)9    0.604899   2.197304  0.2753 0.7840382
## as.factor(id)12   5.904641   1.692072  3.4896 0.0009130 ***
## as.factor(id)14   7.970607   1.749013  4.5572 2.600e-05 ***
## as.factor(id)15  11.935662   2.369536  5.0371 4.616e-06 ***
## as.factor(id)16 -12.899738   1.534515 -8.4064 9.961e-12 ***
## as.factor(id)17  -2.119242   1.377518 -1.5384 0.1291961
## as.factor(id)18  -9.378504   2.289732 -4.0959 0.0001279 ***
## as.factor(id)20 -13.148000   2.206854 -5.9578 1.445e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
a_X <- results["(Intercept)","Estimate"]
b_X <- results["dvskew","Estimate"]

redist_fun <- function(skew, country_fe, error_i, sd = 1) {
  redist <- a_X + b_X*skew + country_fe + error_i
  return(redist)
}
```

Outcomes

```
outcomes <- declare_step(handler = fabricate,
          redist = redist_fun(skew, country_fe, error_i)
)

library('dplyr')
```

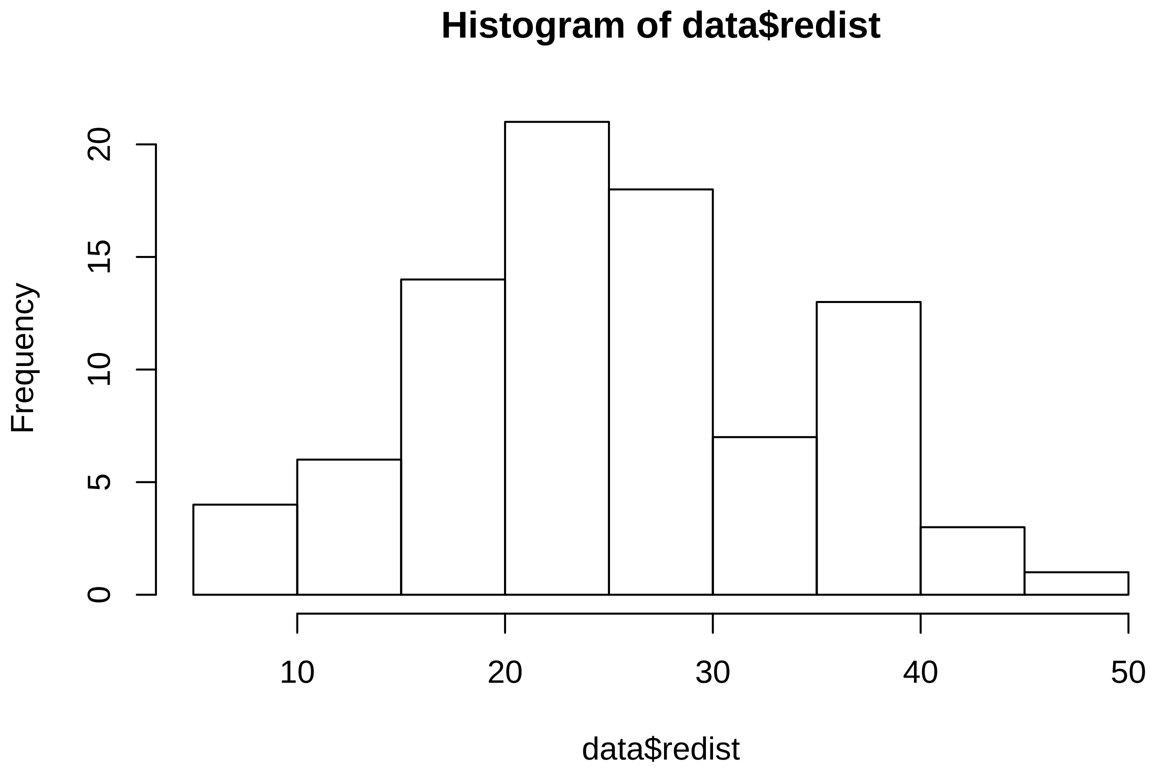
```
## Warning: package 'dplyr' was built under R version 3.5.1
```

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

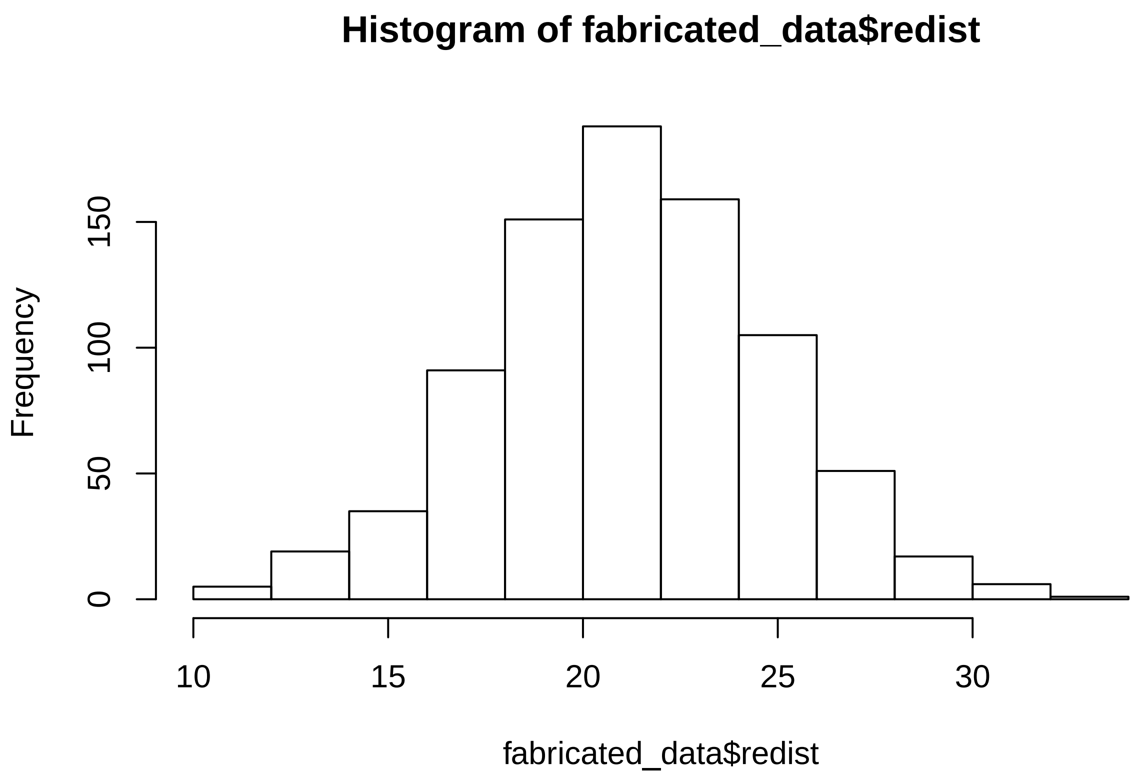
```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

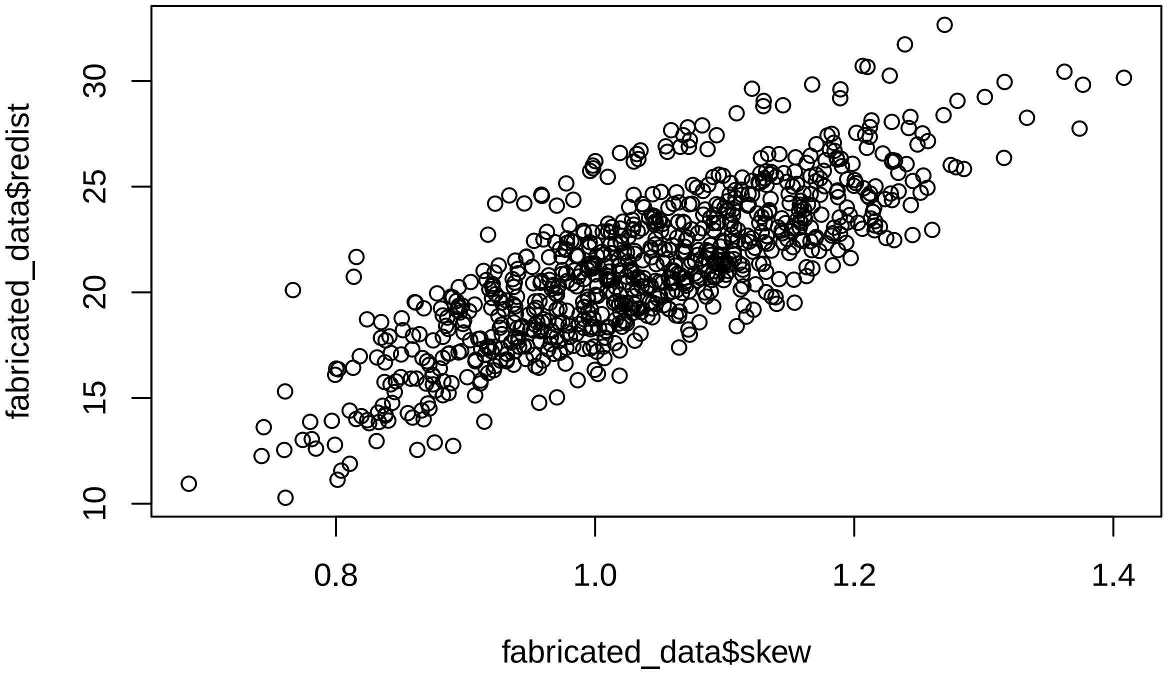
```
fabricated_data <- population() %>% outcomes


hist(data$redist)
```



```r
hist(fabricated_data$redist)
```



```
plot(fabricated_data$skew, fabricated_data$redist)
```



```
lagged_dv <- function(data, lambda = 0.9){ # think about lambda
  within(data, {redist_l1 <- NA
                redist_l1[t == 1] <- 1 + country_fe[t == 1]
                for(j in 2:max(t)){
                  redist_l1[t==j] <- (1-lambda) + lambda*redist_l1[t==(j - 1)] + country_fe[t==j]
                  }
  })
}

add_dv <- declare_step(handler = lagged_dv)
```

Estimand

```
estimand <- declare_estimand(
  SPO = mean((redist_fun(max(skew), country_fe, error_i) - redist_fun(min(skew), country_fe, error_i))/
```

Estimator

```
# Model A
estimator_OLS <- declare_estimator(redist ~ skew, estimand = estimand,
                               model = lm_robust,
                               se_type = "HC1",
                               label = "OLS")


# Model B
```

```
estimator_LDV <- declare_estimator(redist ~ skew + redist_l1,
                                   estimand = estimand,
                                   model = lm_robust,
                                   se_type = "HC1",
                                   label = "LDV")
```

Design

```
temp_design <- population + outcomes + add_dv + estimand + estimator_OLS + estimator_LDV
```

Diagnosis

```
diagnosis <- diagnose_design(
  design = temp_design,
  sims = 500
)

# Normal Design
kable(reshape_diagnosis((diagnosis)))
```

Design Label	Estimand Label	Estimator Label	Term	N Sims	Bias	RMSE	Power	Coverage	Mean Estimat
design	SPO	LDV	skew	500	0.01 (0.01)	0.23 (0.01)	1.00 (0.00)	0.96 (0.01)	24.48 (0.01)
design	SPO	OLS	skew	500	0.03 (0.03)	0.64 (0.02)	1.00 (0.00)	0.94 (0.01)	24.51 (0.03)