University of Toronto
Department of Statistical Sciences

A Replication (and Tribute) of The Log of Gravity

Mauricio “Pacha” Vargas Sepulveda
v. 2021-12-30 14:17

Contents

1 Introduction

1.1 Goal

Silva and Tenreyro (2006) wasn't just an influential article, it defined my interest in gravity models to the point I wrote a master's thesis on it for UN ESCAP. Here I replicate the main results from the original article in R. The original results were obtained in Stata back in 2006. The idea here is to be explicit regarding the conceptual approach to regression in R. For most of the replication I used base R (R Core Team 2021) without external libraries (i.e packages) except when it was absolutely necessary. Much of the methods exposed here lead to the exact same results as using the gravity package (Woelwer et al. 2020) which provides convenient wrappers for gravity estimation.

All questions are welcome to the address `mv.sepulveda@mail.utoronto.ca`.

1.2 Usual disclaimer

The views and opinions expressed in this course are solely those of the author and do not necessarily reflect the official position of any unit of the United Nations, the University of Toronto or the Pontifical Catholic University of Chile.

2 Obtaining the original codes and data

I shall organize the original codes and data from the authors' site to put these on GitHub and therefore ease reproducibility in case of broken links or anything that makes it difficult to obtain the original zip file with the data and codes.

```
url <- "https://personal.lse.ac.uk/tenreyro/regressors.zip"
zip <- gsub(".*/", "", url)
if (!file.exists(zip)) try(download.file(url, zip))

dout <- "regressors"
if (!dir.exists(dout)) unzip(zip, exdir = dout)
```

3 Loading the original data

Thanks to the haven package (Wickham and Miller 2021) we can read Stata datasets directly in R without loss of information about column types and other common problems when reading proprietary formats.

```
log_of_gravity <- haven::read_dta(paste0(dout, "/Log of Gravity.dta"))
```

```
## # A tibble: 18,360 x 63
##    s1_im s2_ex border ex_feenstra im_feenstra landl_im landl_ex  trade  lyim
##    <dbl> <dbl>  <dbl>       <dbl>       <dbl>    <dbl>    <dbl>  <dbl> <dbl>
##  1    88    19      0      530560      443640        0        0 343921  7.16
##  2    72   177      0      367400      162880        0        0      0  5.85
##  3    37   164      0      166940      141400        1        0      0  5.89
##  4   162    20      0      360840      166860        0        0      0  6.34
##  5   117    19      0      530560      454580        0        0 190602  8.04
```

```
##  6   173   194      0      538260      356580        0        0 141604  8.42
##  7   170    90      0      533720      117100        0        0  38457  8.32
##  8    78     8      0      330320      163240        0        0    144  6.28
##  9   150    66      0      532500      336040        0        0  38581  7.55
## 10    53    57      0      138180      162620        0        0    349  7.05
## # ... with 18,350 more rows, and 54 more variables: lyex <dbl>, lpim <dbl>,
## #   lpex <dbl>, laim <dbl>, laex <dbl>, ltrade <dbl>, lremot_im <dbl>,
## #   lremot_ex <dbl>, ldist <dbl>, lypex <dbl>, lypim <dbl>, langclass <dbl>,
## #   comlang <dbl>, colony <dbl>, tariff1 <dbl>, protec <dbl>, comfrt <dbl>,
## #   eec <dbl>, efta <dbl>, sparteca <dbl>, uca <dbl>, aus <dbl>, uis <dbl>,
## #   cacm <dbl>, caricom <dbl>, opeec <dbl>, opefta <dbl>, opsparteca <dbl>,
## #   opuca <dbl>, opaus <dbl>, opuis <dbl>, opcacm <dbl>, opcaricom <dbl>, ...
```

4 Replication attempt

4.1 Poisson Pseudo Maximum Likelihood

Table 3 in Silva and Tenreyro (2006) summarises a large portion of the article and it can be partially replicated with the following Stata code for the Poisson Pseudo Maximum Likelihood.

```
ppml trade lypex lypim lyex lyim ldist border comlang colony landl_ex landl_im
 lremot_ex lremot_im comfrt_wto open_wto
```

In R we would replicate it by fitting two Generalized Linear Models since the article introduces estimates with and without removing zero flows.

```
ppml_formula <- trade ~ lypex + lypim + lyex + lyim + ldist + border + comlang +
  colony + landl_ex + landl_im + lremot_ex + lremot_im + comfrt_wto + open_wto

fit_ppml_1 <- glm(
  ppml_formula,
  data = log_of_gravity,
  subset = trade > 0,
  family = quasipoisson()
)

fit_ppml_2 <- glm(
  ppml_formula,
  data = log_of_gravity,
  family = quasipoisson()
)

coef(fit_ppml_1)
```

```
##  (Intercept)        lypex        lypim         lyex         lyim        ldist
## -31.52955155   0.72132758   0.73187622   0.15443181   0.13268570  -0.77631585
##       border      comlang       colony     landl_ex     landl_im    lremot_ex
##   0.20236913   0.75128105   0.01997206  -0.87241826  -0.70346126   0.64716030
##    lremot_im   comfrt_wto     open_wto
##   0.54925665   0.17944309  -0.13942132
```

```
coef(fit_ppml_2)
```

```
##  (Intercept)        lypex        lypim         lyex         lyim        ldist
## -32.32610286   0.73248076   0.74107804   0.15671171   0.13501848  -0.78380057
##       border      comlang       colony     landl_ex     landl_im    lremot_ex
##   0.19291082   0.74598398   0.02500648  -0.86347371  -0.69642042   0.65984005
##    lremot_im   comfrt_wto     open_wto
##   0.56150020   0.18110716  -0.10681871
```

The replication effort here is null, it just sufficed to look at the summary table in the article and subset the data to drop zero flows. Therefore, it makes sense to proceed with the other models.

4.2 Ordinary Least Squares

The only consideration here is to drop zero flows for some of the models with log in the dependent variable even when Table 3 is not explicit about this, otherwise we break the fitting algorithm.

For example, for estimations of the type $\log(\text{trade}) = \beta_0 + \beta_1 \text{lypex} + \cdots + \varepsilon$, we need to drop zero flows to replicate the result. On the other hand, for estimations of the type $\log(1 + \text{trade}) = \beta_0 + \beta_1 \text{lypex} + \cdots + \varepsilon$, we don't need to drop zero flows.

```
fit_ols_1 <- lm(
  update.formula(ppml_formula, log(.) ~ .),
  data = log_of_gravity,
  subset = trade > 0
)

fit_ols_2 <- lm(
  update.formula(ppml_formula, log(1 + .) ~ .),
  data = log_of_gravity
)
```

4.3 Tobit

The Tobit estimation is similar but requires the use of the censReg package (Henningsen 2020). The complicated part of the estimation here is to extract the right hand side of the model formula to define a vector of zeroes of the length of this right hand side plus two as starting point for the Maximum Likelihood estimation (i.e including the depending variable and intercept besides the estimating slopes).

In order to obtain the a value that matches the results in the article I proceeded with an iteration loop until achieving convergence with respect to one of the estimated slopes. The initial value of $a = 200$ was arbitrary and set after trying reasonable guesses that converge to the slopes in the original article after 9 iterations for a final value of $a = 159$.

```
a <- 200
lypex_ref <- 1.058
tol <- 0.001
lypex_estimate <- 2 * lypex_ref
iter <- 0
```

```
while (abs(lypex_estimate - lypex_ref) > tol) {
  log_of_gravity$log_trade_cens <- log(a + log_of_gravity$trade)
  log_trade_cens_min <- min(log_of_gravity$log_trade_cens, na.rm = TRUE)

  fit_tobit <- censReg::censReg(
    formula = update.formula(ppml_formula, log_trade_cens ~ .),
    left = log_trade_cens_min,
    right = Inf,
    data = log_of_gravity,
    start = rep(0, 2 + length(attr(terms(ppml_formula), "term.labels"))),
    method = "BHHH"
  )

  lypex_estimate <- coef(fit_tobit)[2]
  if (abs(lypex_estimate - lypex_ref) > 2 * tol) {
    a <- a - 5
  } else {
    a <- a - 1
  }
  iter <- iter + 1
}
```

4.4 Non-Linear Least Squares

For this type of estimation the starting values are retrieved from the results of the PPML model with zero flows and then pass these values to a Generalized Linear Model using the Gaussian distribution and a log-link.

```
fit_ppml_eta <- fit_ppml_2$linear.predictors
fit_ppml_mu <- fit_ppml_2$fitted.values
fit_ppml_start <- fit_ppml_2$coefficients

fit_nls <- glm(
  ppml_formula,
  data = log_of_gravity,
  family = gaussian(link = "log"),
  etastart = fit_ppml_eta,
  mustart = fit_ppml_mu,
  start = fit_ppml_start,
  control = list(maxit = 200, trace = FALSE)
)
```

5 Replication results

There wasn’t much effort involved in the replication, which is something desirable. I didn’t even have to email the authors with questions whereas the data was filtered or transformed in ways not mentioned in the article, which is something that we often see. Unlike many articles, this is one of the very few articles that I’ve found which passes the reproducibility review and is very close to full replication according to the criteria from Peng (2011).

Table 1. Replication results for OLS (1-2), Tobit (3), NLS (4) and PPML (5-6).

	Dependent variable:					
	(1)	(2)	(3)	(4)	(5)	(6)
lypex	0.938***	1.128***	1.059***	0.738***	0.721***	0.732***
	(0.012)	(0.011)	(0.011)	(0.004)	(0.008)	(0.006)
lypim	0.798***	0.866***	0.848***	0.862***	0.732***	0.741***
	(0.011)	(0.011)	(0.010)	(0.005)	(0.008)	(0.006)
lyex	0.207***	0.277***	0.228***	0.396***	0.154***	0.157***
	(0.017)	(0.017)	(0.014)	(0.010)	(0.013)	(0.010)
lyim	0.106***	0.217***	0.178***	−0.033***	0.133***	0.135***
	(0.017)	(0.017)	(0.014)	(0.007)	(0.013)	(0.010)
ldist	−1.166***	−1.151***	−1.160***	−0.924***	−0.776***	−0.784***
	(0.034)	(0.037)	(0.029)	(0.008)	(0.018)	(0.013)
border	0.314**	−0.241	−0.225**	−0.081***	0.202***	0.193***
	(0.143)	(0.164)	(0.109)	(0.010)	(0.034)	(0.026)
comlang	0.678***	0.742***	0.759***	0.689***	0.751***	0.746***
	(0.064)	(0.064)	(0.052)	(0.016)	(0.037)	(0.028)
colony	0.397***	0.392***	0.416***	0.036**	0.020	0.025
	(0.068)	(0.068)	(0.056)	(0.018)	(0.043)	(0.032)
landl_ex	−0.062	0.106*	−0.038	−1.367***	−0.872***	−0.863***
	(0.065)	(0.060)	(0.060)	(0.031)	(0.057)	(0.043)
landl_im	−0.665***	−0.278***	−0.478***	−0.471***	−0.703***	−0.696***
	(0.063)	(0.060)	(0.059)	(0.022)	(0.054)	(0.040)
lremot_ex	0.467***	0.526***	0.563***	1.188***	0.647***	0.660***
	(0.078)	(0.089)	(0.077)	(0.018)	(0.048)	(0.036)
lremot_im	−0.205**	−0.109	−0.032	1.010***	0.549***	0.562***
	(0.081)	(0.089)	(0.074)	(0.018)	(0.048)	(0.036)
comfrt_wto	0.491***	1.289***	0.728***	0.443***	0.179***	0.181***
	(0.105)	(0.143)	(0.113)	(0.014)	(0.036)	(0.027)
open_wto	−0.170***	0.739***	0.310***	0.928***	−0.139***	−0.107***
	(0.049)	(0.048)	(0.040)	(0.024)	(0.039)	(0.029)
logSigma			0.677***			
			(0.007)			
Constant	−28.492***	−39.909***	−36.626***	−45.098***	−31.530***	−32.326***
	(1.088)	(1.221)	(1.059)	(0.239)	(0.596)	(0.444)
Observations	9,613	18,360	18,360	18,360	9,613	18,360

Note: *p<0.1; **p<0.05; ***p<0.01

6 References

Henningsen, Arne. 2020. *censReg: Censored Regression (Tobit) Models*. https://CRAN.R-project.org/package=censReg.

Peng, Roger D. 2011. “Reproducible Research in Computational Science.” *Science* 334 (6060): 1226–27.

R Core Team. 2021. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Silva, JMC Santos, and Silvana Tenreyro. 2006. “The Log of Gravity.” *The Review of Economics and Statistics* 88 (4): 641–58.

Wickham, Hadley, and Evan Miller. 2021. *haven: Import and Export 'SPSS', 'Stata' and 'SAS' Files*. https://CRAN.R-project.org/package=haven.

Woelwer, Anna-Lena, Jan Pablo Burgard, Joshua Kunst, and Mauricio Vargas. 2020. *Gravity: Estimation Methods for Gravity Models*. http://pacha.dev/gravity.