Project 2: Global Non-Linear Effect of Temperature on Economic Production

Andy Hultgren, Chester Harvey, and Rafael Grillo Avila

September 10, 2018

Write-Up

Objective

The objective of our assignment was to recreate Figure 2, Panel A of Burke, Hsiang, and Miguel (2015), looking at the global non-linear (quadratic) relationship between annual average temperature andchange in log gross domestic product (GDP) per capita.

Instead of plotting the change in log GDP (roughly measuring the change in growth rates of GDP), we wanted to measure the changes in GDP levels.

Plan of Action and Theoretical Progress & Stumbling Blocks

We were able to recreate the data using the .do files, substituting GDP levels for log-GDP (reflecting growth rate of GDP).

Three models were built, as described below. The issues associated with Model 1 are also described, as well as the attempts to address one of those issues with Models 2 and 3.

Model 1 Format & Issues

Model 1 (extension "_ds421_levels") regresses the level of GDP on a quadratic in country temperature, instead of growth in GDP on a quadratic in country temperature. Country-specific quadratic time controls are still used, just as a default, though it is clear that a country quadratic time control in a levels model is not controlling for the same variation in GDP as the same time trend in a growth model.

Now, this model has many problems associated with it that I think make it a bad idea to run. The first is that economic shocks in general affect a percentage of a country's GDP rather than its level of GDP. Think of some shock (e.g. a trade war, or anything): do we think it would cause a loss of e.g. $1,500 per capita whether you live in China, or Brazil, or the EU, or the US? No. Rather, we typically think of a loss of GDP of e.g. 5% as something that could be comparable across these places. (And data on GDP evolution support this intuition.) Two problems then fall out of this issue:

- *Problem 1:* GDP shocks should enter multiplicatively in the model rather than additively. i.e. the data generating process for country i in time t should be

(a) $GDP_{it} = some\,shock_{it} \times GDP_{it-1}$

and the data generating process should not be

(b) $GDP_{it} = some\,shock_{it} + GDP_{it-1}$

However model (b) is the model that is assumed when doing a levels regression. If you want to assume model (a), which I believe is the correct option, then you have to use the log model that the original authors use.

- *Problem 2:* Both models (a) and (b) above imply persistence of past shocks into the future. This introduces statistical concerns in the estimation process that also make a simple levels model (Model 1 above) a bad idea to run.

What do do about all this:

Both of these issues (Problem 1 and Problem 2) would be typical justifications for running a growth regression as the original authors did (and as Dell Jones and Olken did as well), and not a simple levels regression. But, we want to test the levels regression, so we are stuck with Problem 1. For Problem 2, however, we can try first differencing to attempt to cancel out the persistent, past shocks. The next model does this:

- Model 2 (extension "_ds421_firstDiff"): This model assumes data generating process (b) above, but takes first differences. Effectively: $GDP_{it} - GDP_{it-1} = \textit{some shock}_{it} - \textit{some shock}_{it-1}$. Importantly (for statistical purposes), GDP_{it-1} is no longer on the right hand side of the regression model. We could run a battery of tests to ensure that first differencing fixes the persistent shock problem, but that would be too much work for this assignment.

One other note for this model: here the time controls are country fixed effect and a linear country time trend. In a first differences model, this is essentially similar to modeling a quadratic country time trend for the level (non-differenced) model. So, we make that change for consistency with Model 1 above.

- Model 3 (extension "_ds421_firstDiff_quadT"): This tests the exact same model as Model 2, but includes quadratic country time trends, just to see the sensitivity to trends. Really none of these time controls is similar to having country quadratic time controls for the growth model, so Model 3 is included for the sake of being thorough.

Regarding the country time trends, we would have to do a lot more testing to actually determine what time controls are best given the data, and that is outside the scope of this assignment. Also, it might be futile given Problem 1 anyway.

Results and Implementation Progress & Stumbling Blocks

Results are included alongside the code below, and in the GitHub file that this Markdown document is associated with.

On the data generation side, we were able to generate the data itself with little problem. Credit goes to Andy for his use and editing of Stata to provide this solution. For collaboration, we uploaded the results of the Stata calculation (only one of us had easy access to Stata) and the R Markdown document we were working on onto Github, in a shared repository, along with all of the data, figures, and scripts provided by the authors. We used the "MakeFigure2.R" script as a baseline.

On the plotting side, we stumbled into some issues. Graph scales were coded manually in such a way that in switching the dimensions from dealing with natural logs to whole numbers, we had to play around with changing the hard-coded numbers used for graph scale. It also turned out to be the case that in "MakeFigure2.R", we could not find any code that generated the x-axis, y-axis, histogram, and country labels that are found in Figure 2 in the paper, which means that we would have to input that data ourselves. Some attempts to this effect were made. We were not able to manually label countries of interest as a result, however. We had to add additional lines in the code for the other labels.

The use of base R plotting code (instead of, for example, ggplot2 combined with tidyverse) also made it difficult to parse what was going on and, for instance, we were unable to get the histograms to display to a satisfactory degree partially because of non-intuitive base R code. Though the documentation is certainly very thorough, and the Readme that walked one through the different functions that would have to be run to replicate the data, the amount of work that went into both the original experiment and the documentation implies that we would not be able to do replication justice in a span of even one full day, and much less 3-4 hours. This highlights the importance of giving greater weight to replication exercises, as the due that replication and making replicable studies is given is not commensurate with the amount of work that goes into making studies replicable or replicating them.

Back to the results: the graph shows that with increasing temperature, the impact from higher levels of temperature on GDP remains about average and starts to decrease. Since, as implied above, using levels creates an additive model and doesn't account for the persistence of past shocks, and a first-order difference may not be a significant enough of a time lag, we might be able to interpret the graph as an example of the *marginal* impact that each additional unit increase in temperature might have on level of GDP. Even this interpretation is tenuous at best, however.

Code & Results

Setup

Download necessary libraries and set up Figure 2A frame

```r
require(maptools)
```

```
## Loading required package: maptools

## Loading required package: sp

## Checking rgeos availability: TRUE
```

```r
require(fields)
```

```
## Loading required package: fields

## Loading required package: spam

## Loading required package: dotCall64

## Loading required package: grid

## Spam version 2.2-0 (2018-06-19) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.

##
## Attaching package: 'spam'

## The following objects are masked from 'package:base':
##
##     backsolve, forwardsolve

## Loading required package: maps

## See www.image.ucar.edu/~nychka/Fields for
##  a vignette and other supplements.
```

```r
require(classInt)
```

```
## Loading required package: classInt

## Loading required package: spData
```

```r
require(plotrix)
```

```
## Loading required package: plotrix

##
## Attaching package: 'plotrix'

## The following object is masked from 'package:fields':
##
##     color.scale
```

```r
require(dplyr)
```

```
## Loading required package: dplyr

##
## 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
```

```r
# Delist

rm(list = ls())

#define a function for easy string pasting
"%&%" <- function(x, y) {
  paste(x,y,sep="")
}
```

Get Figure 2A Data

Get the data to make Figure 2, using outputs of .do files provided by Andy Hultgren, and focusing on Model 2 (first differences) specifically.

```r
# Model 2 Data

resp_diff <- read.csv("data/output/estimatedGlobalResponse_ds421_firstDiff.csv")
dta_diff <- read.csv("data/output/mainDataset.csv")
coef_diff <- read.csv("data/output/estimatedCoefficients_ds421_firstDiff.csv")
smpl_diff <- is.na(dta_diff$growthWDI) == F & is.na(dta_diff$UDel_temp_popweight) == F

# Original Data for comparison

resp <- read.csv("data/output/estimatedGlobalResponse.csv")
dta <- read.csv("data/output/mainDataset.csv")
smpl <- is.na(dta$growthWDI)==F & is.na(dta$UDel_temp_popweight)==F   #main estimation sample
coef <- read.csv("data/output/estimatedCoefficients.csv")
```

Edit data. x, mx, est, min90, and max90 should be slightly better explained in the code names.

```r
# center response at optimum
x = resp_diff$x
mx = max(resp_diff$estimate)
est = resp_diff$estimate - mx
min90 = resp_diff$min90 - mx
max90 = resp_diff$max90 - mx

# Countries of interest
ctys = c('USA','CHN','DEU','JPN','IND','NGA','IDN','BRA','FRA','GBR')
ctt = c('US','CHN',"GER","JPN",'IND','NGA','INDO','BRA','FRA','UK')
```

Plot

Note that there is repetition of variables, which is not ideal practice

```
pdf(file="figures/MainFigs_Input/Figure2.pdf",width=10,height=5.5,useDingbats=F)

mat <- matrix(c(1,1,2,3,1,1,4,5),nrow=2,byrow=T)
layout(mat)
ax = 1.5  #scaling for axes
par(mar=c(4,4,2,1))
#initialize plot, accounting for a change in levels for y-axis
plot(1,xlim=c(-2,30),ylim=c(min(min90)-4000, max(max90))-1000,type="n",las=1,cex.axis=0.9,
     xlab = "Annual Average Temperature (C)", ylab = "Change in GDP per capita")

# add vertical average temperature lines for selected countries
for (j in 1:length(ctys)) {
  tt = mean(dta_diff$UDel_temp_popweight[dta_diff$iso==ctys[j]],na.rm=T)
  segments(tt,min(min90),tt,max(max90),lwd=0.5)
}

# plot CI and main effect
polygon(c(x,rev(x)),c(min90,rev(max90)),col="lightblue",border=NA)
lines(x,est,lwd=2)

# now add histograms at bottom
# first calculate percent of population and global gdp produced at each temperature bin, for our estima
bins = seq(-7,30,0.5)
histtemp = dta_diff$UDel_temp_popweight[smpl_diff]
histpop = dta_diff$Pop[smpl_diff]
histgdp = dta_diff$TotGDP[smpl_diff]
pop = gdp = c()
for (j in 1:(length(bins)-1)) {
  lo = bins[j]
  hi = bins[j+1]
  pop = c(pop,sum(histpop[histtemp>=lo & histtemp<hi]))
  gdp = c(gdp,sum(histgdp[histtemp>=lo & histtemp<hi]))
}
pop = pop/sum(pop)
gdp = gdp/sum(gdp)

#parameters that set where histograms go
dis = 3000
base = min(min90) - 200

# now make histograms

#temperature
zz <- hist(histtemp,plot=F,breaks=bins)
cts = zz$counts/max(zz$counts)*2000  #sets the height of the tallest bar to 2000
text(x = 8, y = max(base+cts),
     labels = "Global distribution of temperature observations") # Added
rect(bins,base,bins+0.5,base+cts,col="red")

# pop
```

```r
cts = pop/max(pop)*2000
text(x = 5, y = max(base-dis*(1)+cts),
     labels = "Global distribution of population") # Added
rect(bins,base-dis*(1),bins+0.5,base-dis*(1)+cts,col="grey")

# gdp
cts = gdp/max(gdp)*2000
text(x = 4, y = max(base-dis*(2)+cts),
     labels = "Global distribution of GDP") # Added
rect(bins,base-dis*(2),bins+0.5,base-dis*(2)+cts,col="black")

dev.off()
```

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

Supplementary Material

Readme Text

Readme for replicating Burke, Hsiang, and Miguel 2015

The data and code in the zipped directory BurkeHsiangMiguel2015_Replication.zip replicate all figures, tables, and estimates in Burke, Hsiang, Miguel 2015 (with the exception of Figure 1, which is constructed from figures in the literature).

Full replication requires both R and Stata. User-written programs for both R and Stata are required at various points, as highlighted in script headers where relevant. Code was developed/implemented in Stata/MP 14, and R 3.2.0.

Users of these data should cite Burke, Hsiang, Miguel 2015. If you find a consequential error, or alternatively simply wish to congratulate us on a job well done, please email Marshall at mburke@stanford.edu.

Info on folders contained in replication data —data/input, where input data is stored —data/output, where generated data is stored that is then used to make figures.
—figures/MainFigs_Input where the panels for the figures in the main text are stored —figures/ExtendedDataFigs_Input where the panels for the figures in the extended data are stored —script/ where scripts are stored

The main data file for the regressions is provided in both Stata 14 format (.dta), Stata 13 format (_Stata13.dta) as well as .csv. When importing the .csv into Stata, the option "case(preserve)" is needed to maintain consistency with the provided .do files.

Instructions for replicating all figures/tables in main text and Extended Data For all Stata and R code, user needs to set working directory to the directory where the files were unzipped.

1. First run all the Stata do files in the script/ folder :
   a. GenerateFigure2Data.do
   b. GenerateBoostrapData.do
   c. MakeExtendedDataFigure1.do, MakeExtendedDataFigure2.do, MakeExtendedDataFigure3.do, MakeExtendedDataTable1.do, MakeExtendedDataTable2.do
2. Then run the R scripts generating the impact projections
   a. getTemperatureChange.R
   b. ComputeMainProjections.R
   c. ComputeDamageFunction.R
3. Then run the desired MakeFigure or MakeExtendedDataFigure R script.

Instructions for generating specific figures in the main text For all Stata and R code, user needs to set working directory to the directory where the files were unzipped.

To generate Figure 2, the user needs to run 1. GenerateFigure2Data.do 2. MakeFigure2.R

To generate Figure 3 or 4, the user needs to run: 3. GenerateBootstrapData.do 4. GetTemperatureChange.R 5. ComputeMainProjections.R 6. And then run the desired MakeFigure3.R or MakeFigure4.R

To generate Figure 5, the user needs to run 3-5 above, and then run: 7. ComputeDamageFunction.R 8. MakeFigure5.R