Replication Archive for: "Dividing the Dollar with Formulas"

Gregory J. Martin

May 17, 2017

The replication archive available from the JOP Dataverse[1] includes code and datasets necessary to replicate the analyses in Martin (2017), "Dividing the Dollar with Formulas." The replication archive contains two folders, one containing code and the other containing datasets. Most of the code is written in `R` but there are a few components written in Python or `C`. The Python component parses the Library of Congress' bill description HTML files, and the `C` component implements the linear programming solver necessary for the simulation part of the paper. This part must be compiled and loaded into `R` as a module, as described below.

1 Replication code

There are three subfolders in the code directory:

1. `lpSimPack`
2. `python`
3. `R`

1.1 lpSimPack

This folder contains an `R` package that implements the linear programming solver required for the simulation exercise described in the paper. The folder

[1] `http://dataverse.harvard.edu/dataverse/jop`

contains functions written in both R and C (the C functions are compiled and then called from the R functions). To install it, enter the code directory in a terminal window and then type at the prompt:

```
R CMD INSTALL lpSimPack
```

This step will compile all the required C functions, and place the compiled code in the user's R library directory. Once installed, the package can be loaded into the R environment using the command `library(lpSimPack)`. The C functions require the GNU Linear Programming Toolkit (GLPK),[2] source code for which is included in the `src` subfolder. GLPK code will be compiled and installed to your R package directory prior to compilation of the simulation functions; note that depending on your system this step may require modification of the `Makefile` in the `src/` subdirectory to work properly. I have only tested it on a Linux (Ubuntu 16.04.2) system, but it should be portable to other operating systems.

1.2 python

Python code to download and process some of the raw datasets is included in this folder. `get_usaspending_data.py` automates the downloading of the grant database maintained at usaspending.gov. `LOCThomasParse.py` parses the Library of Congress' THOMAS database of congressional bills. `raster2shp.py` converts the landcover data files from raster (image) format to shapefile format.

1.3 R

The R folder contains the programs necessary to generate the tables and figures in the paper and appendices. There are three subfolders in the directory.

1.3.1 data_construction

The `data_construction` subdirectory contains code to process and merge the raw data into the form required for the estimation and results. Names of programs in this directory indicate the order in which they should be run; for example, `01_rollcall_data_input.R` depends on files generated by

[2] `http://www.gnu.org/software/glpk`

the programs named `00_*`, and so on. `02_match_spending_to_vote.R` also generates summary statistics tables included in the paper and appendix.

Note that due to their size, the raw data sets on which these programs rely are not included in the replication archive. (For example, the USAspending data alone is 65GB). The raw data are publicly available, and sources from which they can be downloaded are cited in the text and appendices.

1.3.2 model_estimation

The `model_estimation` subdirectory contains the code to run the simulation analysis described in the article, and produce Figures 1 and 2 in the text along with additional figures in the appendices. Code in this directory relies on the `lpSimPack` package, which must be compiled and installed first, as described above. Names of programs in this directory indicate the order in which they should be run.

1.3.3 regressions

The `regressions` subdirectory contains code to produce the regression analysis (Table 1 in the text and several tables in the appendices) as well as other descriptive analyses of the spending data described in the article. These programs can be run in any order.

2 Replication data

The data directory contains the datasets required to produce the tables and figures in the article and appendices. Most are in the `.RData` format and can be loaded directly into `R` using the `load()` command. A few of the raw data sources are included in `.csv` or text formats, but most are excluded due to their size, and only the processed, final datasets used in the analyses are included in the archive. There are five subfolders in the data directory:

1. `cfda/` includes data from the Catalog of Federal Domestic Assistance (CFDA). The raw data are in the files named `programs-full-datagov*`. `cfda_formulas.xlsx` contains a listing of textual sources for the statutory formulas of programs in the sample, and `cfda_progs_formula.csv` contains the coding of the number of factors in statutory formulas used to produce Table 2.

2. `clean/` contains the final, merged datasets used for the simulation and regression analyses.

3. `locthomas/` contains the source HTML files from the Library of Congress' THOMAS catalog of congressional bills, along with a single processed file, `bill2law.csv`, containing information on the mapping between bills and public laws.

4. `rollcalls/` contains information on roll call votes by members of Congress. Raw data from Keith Poole's `voteview.com` is in `csv` or text format; `.RData` files are the processed versions.

5. `usaspending/` contains the processed data on spending allocations generated from the raw USASpending data. Raw data is omitted due to size; it can be downloaded using the Python script included in the replication archive.