Foreign Aid and Growth: A Sp P-VAR Analysis Using Satellite Sub-National Data for Uganda

Replication Codes Technical Note

Andrea Civelli
University of Arkansas

Andrew Horowitz
University of Arkansas

Arilton Teixeira
Ibmec

Abstract

This short technical note is the documentation in support of the replication codes provided for the paper *Foreign Aid and Growth: A Sp P-VAR Analysis Using Satellite Sub-National Data for Uganda*.

N1 Overview

The analysis in this paper has been conducted using Stata, MATLAB, and QGIS. Section N2 describes the codes necessary to replicate the results of the paper. The main results are obtained by the routine *main_pvar_analysis.do*, which requires the supporting code for the spatial P-VAR estimation *pvarcivo.ado*. These codes are optimized for Stata 14.

These two codes use as input the dataset *pvarDATA_main.dta*, which is cleaned and set-up in the right format by a series of MATLAB and GIS tools. The data manipulation starting from the raw data is documented in Section N3.

The full set of codes and datasets are provided in the two folders *CHT_JDE_replication_codes* and *CHT_JDE_setup_codes* that are found in the the zipped file *CHT_JDE_Documentation*. This archive has to be unzipped in the desired working directory. Before running any codes, remember to set the Stata's directory path to the right working directory where the files are copied. The codes assume this preliminary step has been made.

The codes rely on some user-developed tools which are not part of the standard Stata installation package. These tools are all freely available from the online Stata depository and their installation is implemented by the file *pre_installation_packages.do* which has to be executed before running any other *do* file. However, they can also be manually installed by the user typing the *ssc install* command or *search* in the Stata console, should any of them still be missing. The MATLAB codes also require some additional user-developed functions that are provided in the zipped folder.

N2 Main Results (in Stata)

This section refers to the content of folder *CHT_JDE_replication_codes*.

Main results and robustness checks:

- The file *main_pvar_analysis.do* produces most of the results in the main paper, in the online Appendix, and the robustness checks. The code is a standalone routine which runs immediately after unzipping the files in the desired directory. This code uses the data from *pvarDATA_main.dta*. Do not forget to run *pre_installation_packages.do* before anything else to install a few required packages.

- *noelec_pvar_analysis.do* runs the robustness check without electricity and power-supply projects. It uses the dataset *pvarDATA_noelec.dta*.

- *WB_pvar_analysis.do* runs the robustness check with the ODA data from the World Bank dataset. It uses the dataset *pvarDATA_WB.dta*.

- *short_pvar_analysis.do* runs the robustness check with the early-impact ODA projects for the reduced sample of districts. It uses the dataset *pvarDATA_short.dta* and the codes *irfshort1.ado*, *irfshort2.ado*, and *irfshort3.ado* to compute the residual impulse response functions starting from the early-impact responses.

- The file that executes the spatial autocorrelation analysis is *moran_spatial_autocorr.do*, which has to run after *main_pvar_analysis.do* because it uses the dataset *resid_moran.dta*, which is generated by *main_pvar_analysis.do*. The contiguity matrix for this analysis is provided by the file *CC36.dta*.

P-VAR code with spatial lag: The P-VAR model with spatial lags is estimated by the Stata program code *pvarcivo.ado*. This is a modified version of the well-known Love-Zicchino Stata package for the estimation of a panel VAR which includes the following new features:

1. The option to add to the regression model spatial lags of the endogenous variables (SP P-VAR).

2. An option to adjust the residuals covariance matrix for within group autocorrelation and heteroskedasticity.

3. The computation of the estimated residuals series as part of the standard output of the code.

The first two new features introduce some changes in the syntax of the P-VAR estimation command *pvarcivo* with respect to the original *pvar*. In the call line of the program, the options now also include:

- Spatial(*varlist*) allows to declare a list of spatial variables to be added as controls to the regression model. These variables must be first-order spatial lags of the endogenous variables. The spatial terms have to be created externally to the code and provided as panel variables with the correct structure. The program is not able to aggregate the spatial terms as of now, but this is a relatively easy extension to it. The command gives the flexibility to include only a subset of spatial lags of the endogenous vector if needed. For example, in the paper this option corresponds to spatial(lights36 odas36), where the spatial lags of lights and aid are used.

- IVspt(*varlist*) allows to declare a list of exogenous spatial variables to be used as additional instrumental variables in the GMM procedure. As above, the spatial variables must be created externally to the code and they cannot be used as exogenous regressors at the same time. This feature is added because it is very useful to have such exogenous spatial IVs to instrument the spatial terms in the GMM estimation procedure. For example, in the paper this option corresponds to ivspt(rains36), where the spatial lag of rainfall is used.

- sig2 is an option to compute the covariance matrix of the residuals allowing for within individual autocorrelation and heteroskedasticity. This option mimics the correction used in the estimation for the coefficient standard errors vce(cluster *clustvar*) and the GMM weighting matrix wmatrix(cluster *id*) in Stata. Therefore, even though the code allows one to use sig2 independently of the other options, these options should be jointly selected.

Finally, the third feature produces a series of residuals resid00‘var’ for each of the endogenous variables ‘var’ of the model.

The P-VAR analysis then relies on a set of other codes as well for the impulse responses computation, the stability analysis, the variance decomposition, etc. Some of these codes remain the same as in the original package, while others require simple modification to fit the new program *pvarcivo.ado*. The following is the list of the subsidiary codes:

1. Adjusted codes: *pvarsocivo.ado* for testing optimal lag structure; *pvarstablecivo.ado* for testing stability in the time dimension of the VAR model. These follow the original command syntax.

2. Original codes: *pvarirf.ado* for impulse response analysis; *pvarfevd.ado* for variance decomposition.

3. Unverified codes: *pvargranger.ado* was not used.

The time-space IRF: The time-space IRF are computed by the code *pvarirfcivo.ado*, which starts from the time-only impulse response code *pvarirf.ado* and adds the spatial spillover to it. The codes generates an average time-space IRF as explained in Section 2.3 of the main paper. The code is written to accommodate only the Sp P-VAR(1,1) model estimated in the paper, but it can be relatively easily extended to a higher order VAR.

This code requires the specification of a few additional arguments in the call line of the program, and accept all the options already used by *pvarirf.ado* :

- Sorder(*varlist*) allows to declare the structural order in which the spatial variables in the model are going to be used in the computation of the IRF. The order of these variables cannot differ from that of the endogenous variables in the Cholesky identification.[1] For example, in the paper this argument is specified as sorder(odas36 lights36), which corresponds to the main identification porder(odas lights).

- Spmat(*matrix*) identifies the spatial matrix to be used in the computation of the spatial spillover of the IRF. Even though in principle any spatial matrix can be selected, the only feasible choice is the spatial contiguity matrix used to compute the spatial lags of the variables in the model. For example, in the paper this argument corresponds to spmat(CC36).

Predictive model estimation: The estimation of the predictive model is conducted by *reg_predictive_model.do*, which runs the regression for the two measures of household expenditure (weekly consumption and monthly non-durable goods). The underlying datasets of this code are *expenditureDATA_weekly.dta* and *expenditureDATA_monthly.dta* respectively.

N3 Datasets Construction (in Stata, MATLAB, and QGIS)

This section refers to the content of folder *CHT_JDE_setup_codes*. The datasets for the different aspects of the Sp P-VAR analysis are generated in two steps. First, the raw data is elaborated in MATLAB and/or QGIS and saved in a series of *csv* files. Second, these files are imported to Stata and converted into the final *dta* files used in the analysis described above.

Data elaboration:

1. The four csv files for the VAR analysis (*pvarDATA_uganda*, *pvarDATA_noelec_uganda*, *pvarDATA_short_uganda*, *pvarDATA_wb_uganda*) and the two files for the predictive regressions (*monthlyDATA* and *weeklyDATA*) are obtained from the two MATLAB routines *prepareDATA.m* and *prepareDATA_weekly.m*. These codes are the final step in the data processing stage, and they assemble the datasets in the right structure to export them to Stata. They use as intermediate input a set of cleaned-up and pre-processed data files described in the next point.

2. The codes in point 1 require some pre-processed intermediate input obtained from different sources of raw data. These intermediate input files are already provided in the main folder JDE_setup_codes for convenience, and they are replicated by the processes and codes described below:

[1] In principle, the code would be able to execute even with a different order, but it would generate theoretically incorrect IRF.

- Nighlights by district (in the folder *cleanup*): First, luminosity data for Uganda is imported by *countrybox.m* starting from the satellite images. Given the very large size of the image files, these images are supplied in the separate zipped folder *nightlights.zip* and must be unzipped in this folder. Lights by district are computed by *bydistrictlight.m* taking the output of *countrybox.m* and generating the district series stored in *ugandadistL.mat*.
- ODA by district in its multiple forms (in the folder *cleanup*): Total ODA, ODA excluding energy related projects, early-impact ODA, and ODA from the World Bank are constructed following the same procedure.
  A. From the AidData data source, projects are matched to the administrative districts and filtered according to the desired definition: precision 1-3 for total ODA and WB aid, no energy projects (purpose code 311), only early-impact projects (see codes listed in the Appendix of the paper). This is done in the excel files *CleanUpUgandaRV4* for the non-World Bank data and *CleanUpWBUganda* for the World Bank data.
  B. Multiple disbursements of a single project are distributed over districts by population size as explained in the Appendix of the paper. This is done by the codes *prjsplit.m* for the non-World Bank data and *prjsplitWB.m* for the World Bank data.
  C. The projects can be then aggregated by districts by the codes *bydistrict.m* for the non-World Bank data and *bydistrictWB.m* for the World Bank data, which produce the output for *prepareDATA*: *ugandadisbdist.mat*, *ugandadisbdist_noelec.mat*, *ugandadisbdist_WB.mat*, and *ugandadisbdist_short.mat*.
- Rainfall series at the district level are generated by *bydistrictrain.m* starting from the satellite raw data for precipitation (see folder *cleanup* as well). Given the very large size of the image files, these images are supplied in the separate zipped folder *precipitations.zip* and must unzipped in this folder. The output of this code is the file *ugandadistRF.mat*.
- Expenditure data from the household surveys (in the folder *household*): the files *espenditure.xlsx*, *espenditure_dark.xlsx*, *espenditureweek.xlsx*, and *espenditureweek_dark.xlsx* are generated by the Stata files *harmonization.do*, *harmonization_dark.do*, *harmonization_30days.do*, *harmonization_30days_dark.do*. The starting raw data is given by the responses to the survey questions in *GSEC1.dta*, *GSEC15b.dta*, *GSEC15c.dta* and *sec1.dta*, *sec7a.dta*, *sec7b1.dta*. The households' geo-locations are matched to the districts *survey09tostata.csv* and *survey09tostata.csv*.
- Population data is aggregated by district in QGIS and provided in table format by *PopUgandaDistricts.xlsx* for years 1990, 1995, 2000, 2005, 2010, 2015.
- U.S. price deflator and exchange rate of Ugandan shilling to the $ are obtained from FRED and found in *PindexUS.xlsx*.

Contiguity matrix: The contiguity matrix for Stata is provided by the file *CC36.dta*. The neighbor districts are found from the administrative map of Uganda using the python code *contiguity_matrix.py* available in the folder *GIS*. This code identifies the neighbor links and its output is the text file *mat36dark.txt*, which is then used by *prepareDATA.m* in Excel format to build the contiguity matrix. *prepareDATA.m* also takes care of *CC22.dta*, the contiguity matrix used for the smaller sample of districts in the early-impact ODA analysis. The *GIS* folder also provides some additional resources and geographic analysis used throughout the paper.

Final datasets: Table N1 illustrates the codes that implement the transformation of the final datasets.

.csv input file	.do file conversion	.dta final output
pvarDATA_uganda	setup_datasetPVAR	pvarDATA_main
pvarDATA_noelec_uganda	setup_noelecPVAR	pvarDATA_noelec
pvarDATA_short_uganda	setup_shortPVAR	pvarDATA_short
pvarDATA_wb_uganda	setup_wbPVAR	pvarDATA_wb
monthlyDATA	setup_datasetMONTH	expenditureDATA_monthly
weeklyDATA	setup_datasetWEEK	expenditureDATA_weekly

Table N1: From intermediate input data to the final data for the analysis in Stata