

Issuer-Level Ultimate Parent Aggregation Code for "Redrawing the Map of Global Capital Flows: The Role of Cross-Border Financing and Tax Havens"

This Version: V1.0 (June 2019)

1 Introduction

The files contained in this repository provide an implementation of the issuer-level ultimate parent aggregation procedure of Coppola, Maggiori, Neiman and Schreger (2019), which associates the universe of traded equity and debt securities with their issuers' ultimate parents, including those issued in tax havens.

There are four subfolders in this code repository:

1. The scripts in the folder `cmns_aggregation` implement the aggregation procedure and are the main focus of this repository. They take as input processed files that are at the path `<DATA_PATH>/aggregation_sources`. The required structure of these data files is described in the script `cmns_aggregation/UP_Aggregation.py`. The script also provides configurable parameters that allow to bypass certain source files if these are missing. We strongly recommend using all the aggregation sources if possible, since the quality of the final mapping will deteriorate as sources are removed.
2. The scripts in the folder `data_sources_build` import raw data in order to create the processed files at `<DATA_PATH>/aggregation_sources` that are used as inputs in the previous steps. These files could be created in alternative ways as well (depending on data delivery specifics), as long as they respect the structure described in `UP_Aggregation.py`. In that case, the files at `data_sources_build` can be used to inform alternative data processing implementations.
3. The folder `figi_download` contains a script that will allow you to map a given set of CUSIP identifiers to the corresponding FIGI identifiers from Bloomberg's OpenFIGI API and to other security-level information such as security name and security type. The security type information from OpenFIGI is used in some of the data-processing scripts included in `data_sources_build`. Since the OpenFIGI API one-time download script takes a long time to run for the universe of CUSIPs from CUSIP Global Services (CGS), it is included as a separate utility. The OpenFIGI download script relies on the output of the file `data_sources_build/cgs.do`, which generates the universe of CUSIP identifiers using CGS data: that script should therefore be run before starting the OpenFIGI download. As outlined below, the download also requires an API key, which should be obtained directly from OpenFIGI.
4. The folder `sample_directory_structure` illustrates the structure of the data directories that is referenced in this repository. The files in this folder are sample versions that only include a limited number of rows as these are included in order to demonstrate the structure of the data.

2 Executing the Code

The bash script `Run_Aggregation.sh` is the main executable file for the procedure. Launching `Run_Aggregation.sh` will run the full procedure. The scripts that import and process the raw data sources are turned off by default, but these can be activated by setting the appropriate flag in `Run_Aggregation.sh`.

This file should be called as: `./Run_Aggregation.sh`

3 Technical Notes

- The code is built for Unix systems and assumes that both the Stata and Python interpreters are configured on your executable path. The required version for Python is 3.6+; packages may need to be installed using a package manager (e.g. pip) as necessary.
- Prior to running the procedure, please be sure to perform a find-and-replace in the code folder for the following expressions. These are user- and system-specific procedure parameters that will need to be filled in accordingly. Individual files also point out these parameters whenever they are present:
 - `<CODE_PATH>`: Path to the copy of this code repository on the host system.
 - `<DATA_PATH>`: Path to the folder containing the data, in which the procedure is executed.
 - `<API_KEY>`: If using the script in `figi_download`, an OpenFIGI API key is required. This should be obtained directly from OpenFIGI and used in place of this placeholder.

4 Input Data

Please see the accompanying file `CMNS_Data_Guide.pdf` (available on GitHub) for a list of all the raw input files that are used by the CMNS aggregation algorithm. All paths in the guide are relative to the root data folder `<DATA_PATH>`. Users should use the folder structure outlined in the guide in order for the procedure code to run as-is.

References

Coppola, Antonio, Matteo Maggiori, Brent Neiman, and Jesse Schreger, “Redrawing the Map of Global Capital Flows: The Role of Cross-Border Financing and Tax Havens,” *Working Paper*, 2019.