Research workflow with confidential data: The ‘Expert’ BPLIM researcher Workflow

2023-12-18

Access to BPLIM's microdata

BANCO DE PORTUGAL EUROSISTEMA Banco de Portugal Microdata Research Laboratory - BPLIM

Search...

About BPLIM | Data | Resources | Projects | Events | Forum | INEXDA

Access

The **Banco de Portugal Microdata Research Laboratory (BPLIM)** provides two forms of access to the microdata sets managed by Banco de Portugal: **on-site** and **remote access**.

BPLIM has two servers. The Internal Server is primarily intended for use by Banco de Portugal own researchers (internal researchers) while the External Server has remote access capabilities and is intended for external researchers. Internal researchers are not bound by any restriction in terms of data access but restrictions apply to external researchers.

For complete information about all relevant procedures involving data access at BPLIM please consult the Guide for Researchers.

Application

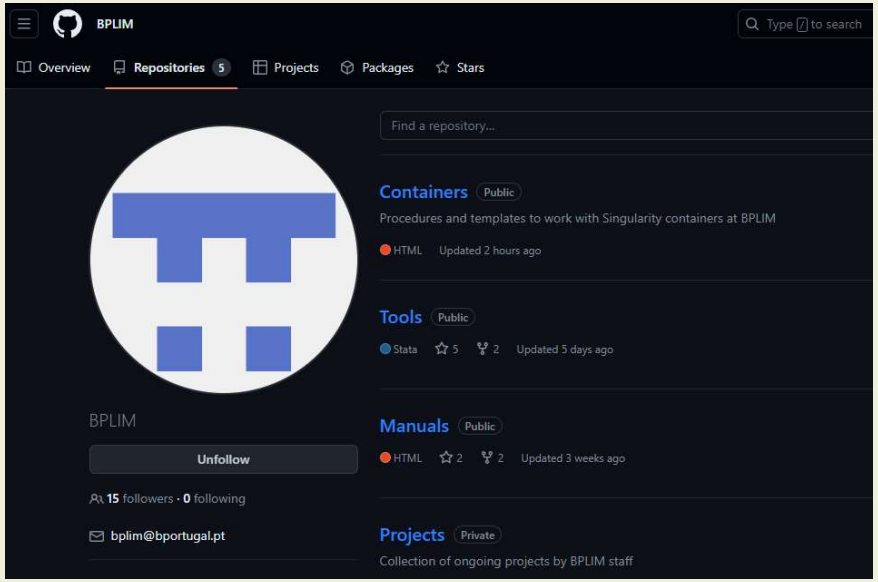
Access to BPLIM micro data sets implies the submission of an Application Form that can also be found in the Download section below. The application must:

- contain a short description of the research project;
- identify all researchers involved in the project and their affiliation;
- identify the datasets, timeframe, and variables required.

All external researchers with access to the data are further required to sign a Confidentiality Agreement and send a copy of their Curriculum Vitae. If the project consists of a master or doctoral dissertation then the supervisor(s) has/have to be identified and must also sign the Confidentiality Agreement.

1. https://bplim.bportugal.pt
2. Guide for Researchers
3. Application Form
4. Confidentiality Agreement

BPLIM's GitHub



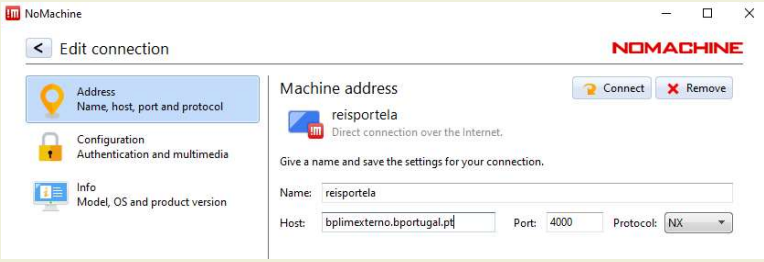
1. https://github.com/BPLIM
2. We have made available on GitHub the tools developed by the BPLIM team, statistical packages, and containers, as well as the documentation associated with each of the databases.

Remote access to BPLIM's computational facilities

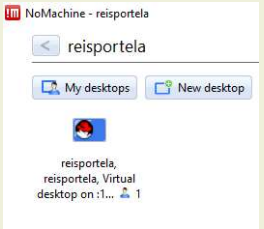
NoMachine: Download and install

NoMachine

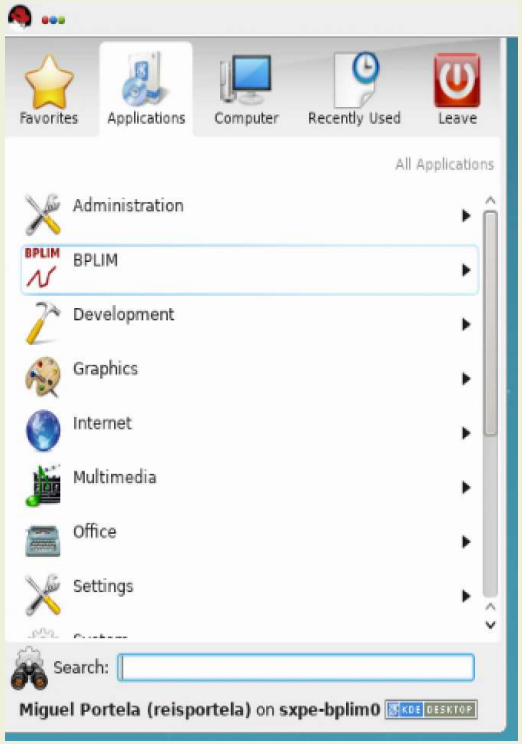
Remote server: NoMachine connection



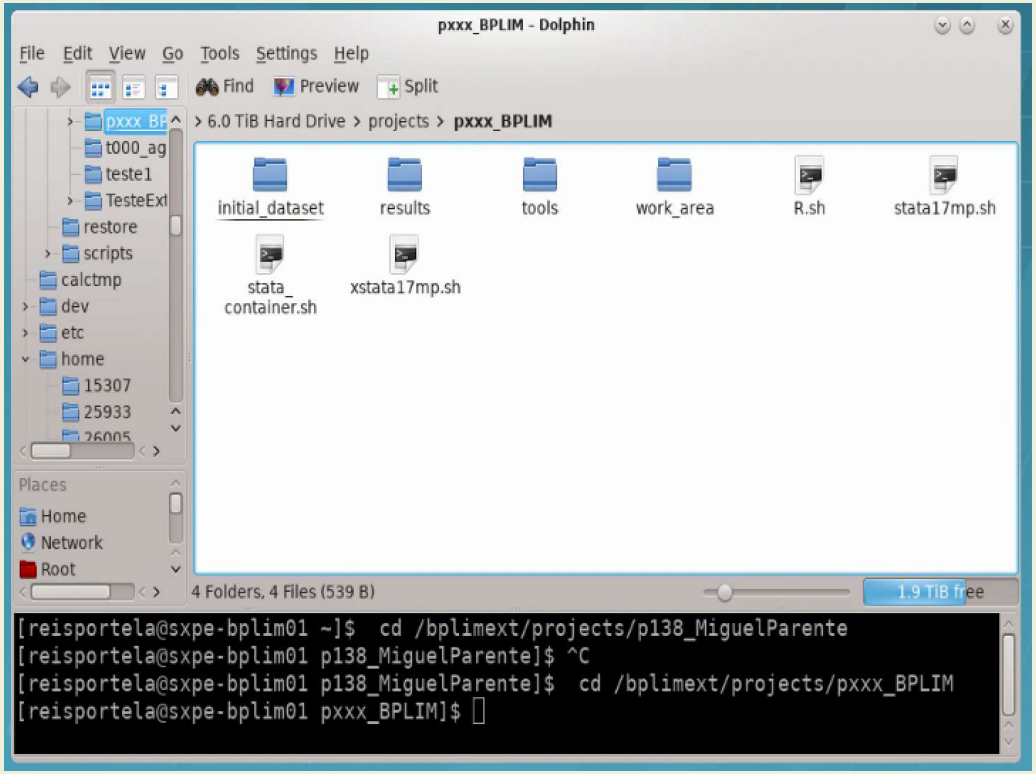
Add and configure connection



Remote server: Desktop

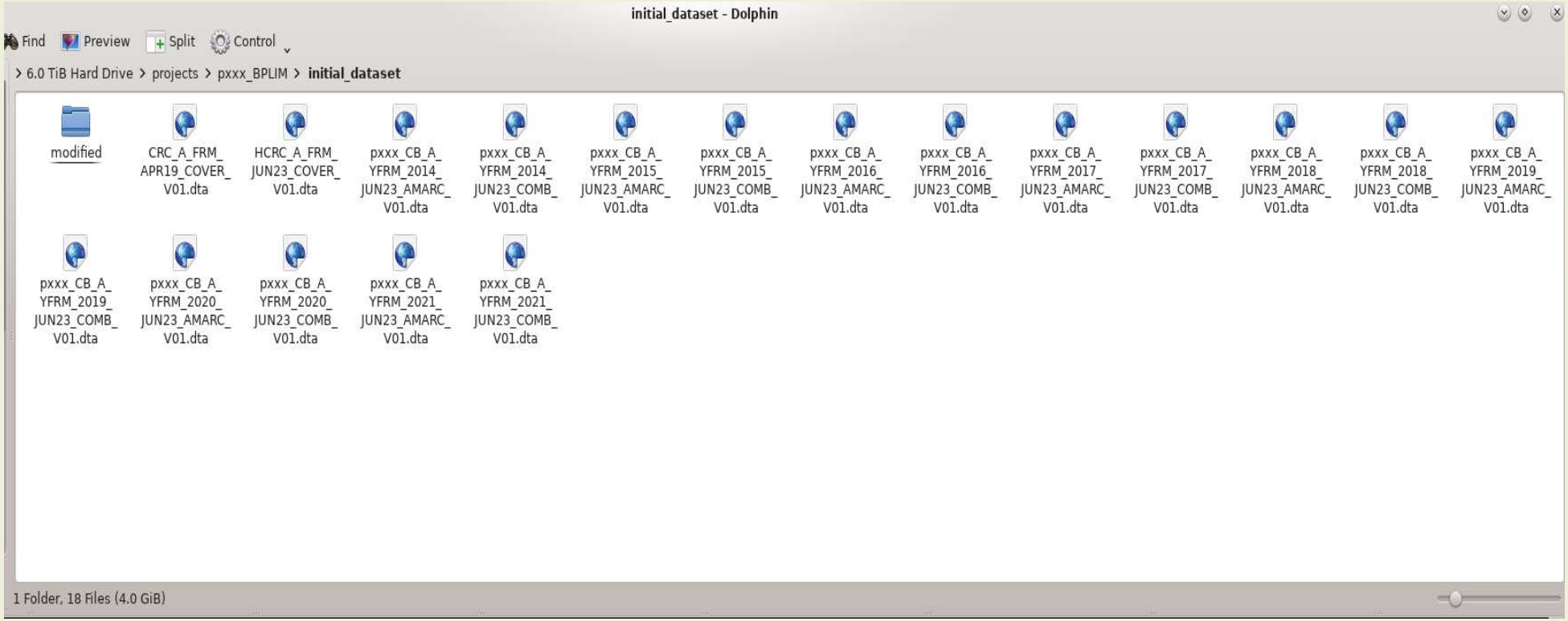


Remote server: File manager



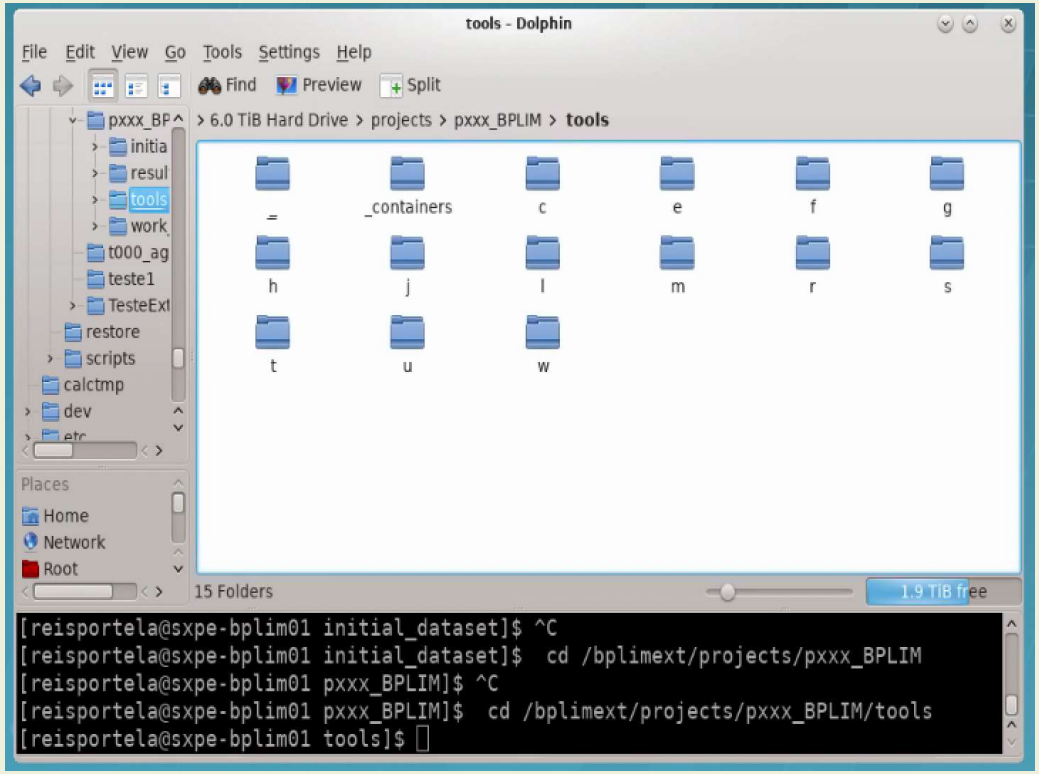
Dolphin

Remote server: Initial dataset



initial_dataset folder

Remote server: Tools



tools folder

Remote server: Stata

- Profile and template do-file

```
*******************************************************
*             Initialization
*******************************************************
version 17
clear all
program drop _all
set more off
set rmsg on
set matsize 10000
set linesize 255
capture log close
*******************************************************
*                Define globals                       *
*******************************************************
**** Path for replication ****
* Base path for replications
global path_rep "/bplimext/projects/pxxx_BPLIM/work_area/"

**** Paths for data ****
* Set the path for non perturbed data source
global path_source "/bplimext/projects/pxxx_BPLIM/initial_dataset"
* Set the path for perturbed data source
global path_source_p "/bplimext/projects/pxxx_BPLIM/initial_dataset/modified"
* Set the path for intermediate data source
global path_source_i "/bplimext/projects/pxxx_BPLIM/initial_dataset/intermediate"

**** Globals for type of modified dataset
* Perturbed
global M1 "P"
* Shuffle
global M2 "S"
* Randomized
global M3 "R"
* Dummy
global M4 "D"
/*******************************************************************
********* Example: using non-modified and modified data sets *********
* Anonymized (CB_A_YFRM_2010_JUN21_ROSTO_V01.dta)
use "${path_source}/CB_A_YFRM_2010_JUN21_ROSTO_V01.dta"
* Perturbed (CRC_P_MFRM_2010_APR19_COBR_V01.dta)
use "${path_source_p}/CRC_${M1}_MFRM_2010_APR19_COBR_V01.dta"
* Shuffle (PE056_S_rejected_applications.dta)
use "${path_source_p}/PE056_${M2}_rejected_applications.dta"
* Randomized (CRC_R_MFRMBNK_2007_APR19_CO_V01.dta)
use "${path_source_p}/CRC_${M3}_MFRMBNK_2007_APR19_CO_V01.dta"
* Dummy (SLB_D_YBNK_20102018_OCT20_QA1_V01.dta)
use "${path_source_p}/SLB_${M4}_YBNK_20102018_OCT20_QA1_V01.dta"
*******************************************************************/

**** Path for project specific ado files ****

adopath ++ "/bplimext/projects/pxxx_BPLIM/tools"
adopath ++ "/bplimext/projects/pxxx_BPLIM/work_area/ados"
```

```
* Project      :  pxxx_BPLIM
* Author(s)    :
* Date         :
* Description  :
* Dependencies :
* Modifications: (add date, author and change)

* Run profile (usually not needed, but just to be sure)
capture run "profile.do"

* Change to work path - global `path_rep` defined in profile.do
cd "${path_rep}"

/* You should create a `results` folder to save outputs (this is ideal for replications)
Always use capture when creating diectories in scripts
*/
capture mkdir results
* You may create the structure that you want, adding sub-diectories to `results`
capture mkdir results/tables
capture mkdir results/figures

/*
When defining globals for paths (if you do not want to use relative paths), remember to include
the global `path_rep`. This is the path where the analysis should run. See the two examples below,
where we define two globals for separate results folders
*/
global results_tables "${path_rep}/results/tables"
global results_figures "${path_rep}/results/figures"

* Creating a log file in the work area, where "logexample" is the log requested for extraction
log using "logexample.log", replace

*******************************************************
*                  Open data files                    *
*******************************************************
/*
Please note the VERY IMPORTANT use of globals `M1` and `M4`, ${M1} and ${M4},
in the file names of the modified data. The first is for perturbed data and
the second is for dummy data. Failing to use this globals when working with
modified data will cause the REPLICATION TO FAIL.
*/

* Example on how to read a non-perturbed data file provided by BPLIM:
use "${path_source}/P*###_CBHP_A_YFRM_20062016_JUN18_ROSTO_V01.dta", clear

* Example on how to read a perturbed data file provided by BPLIM:
use "${path_source_p}/P*###_${M1}_CRC2011_FRM_COBR_V01", clear

* Example on how to read a dummy data file provided by BPLIM:
use "${path_source_p}/P*###_${M4}_CRC2011_FRMBNK_COBR_V01", clear


*******************************************************
*            Start data analysis                      *
*******************************************************
*
* ...
* YOUR STATA CODE GOES HERE
```

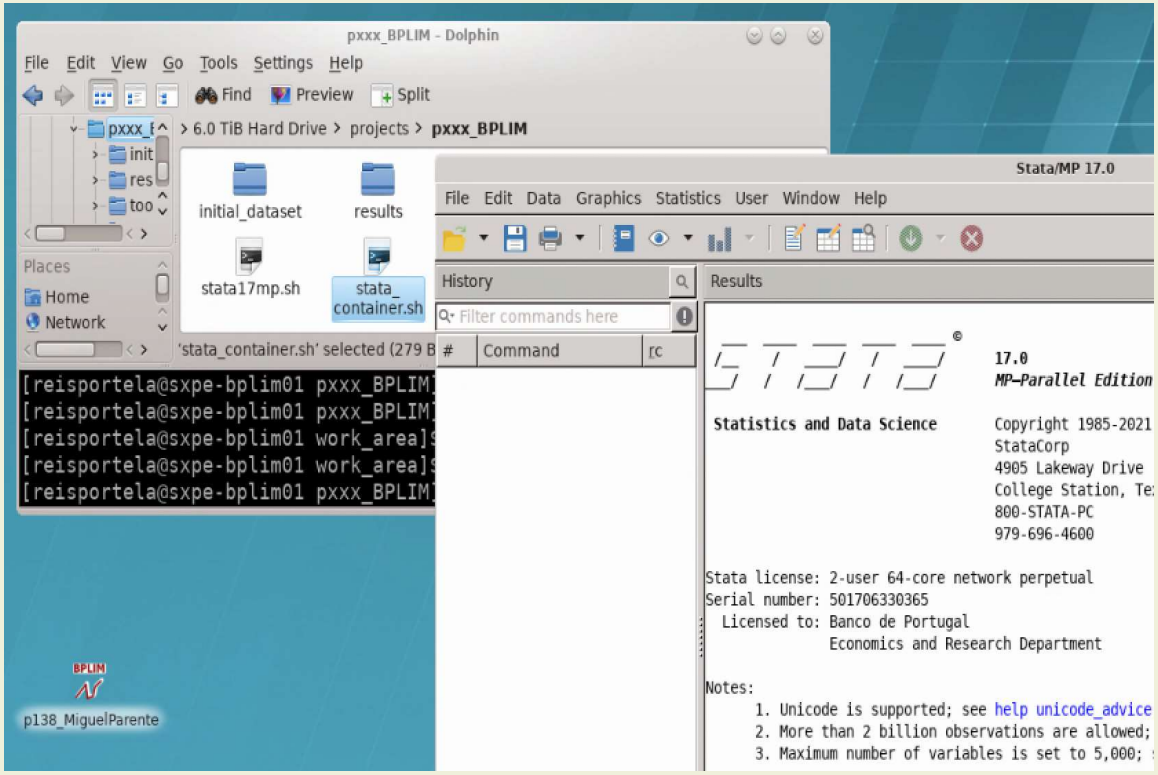
Remote server: Additional packages

- In case we need additional packages we ask BPLIM's staff to install them (tools folder)
- A more flexible approach is the use of containers – **reproducibility** and **autonomy** regarding packages and versions

Remote server: Containers, the concept

“A container is a lightweight, stand-alone, executable package of software that includes everything needed to run a piece of software, including the code, runtime, system tools, libraries, and settings. Containers are isolated from each other and the host system. This isolation allows for efficient, reliable, and consistent deployment of applications, regardless of the environment.” (ChatGPT, 2023)

Remote server: Stata running in a Container



How to build a container?

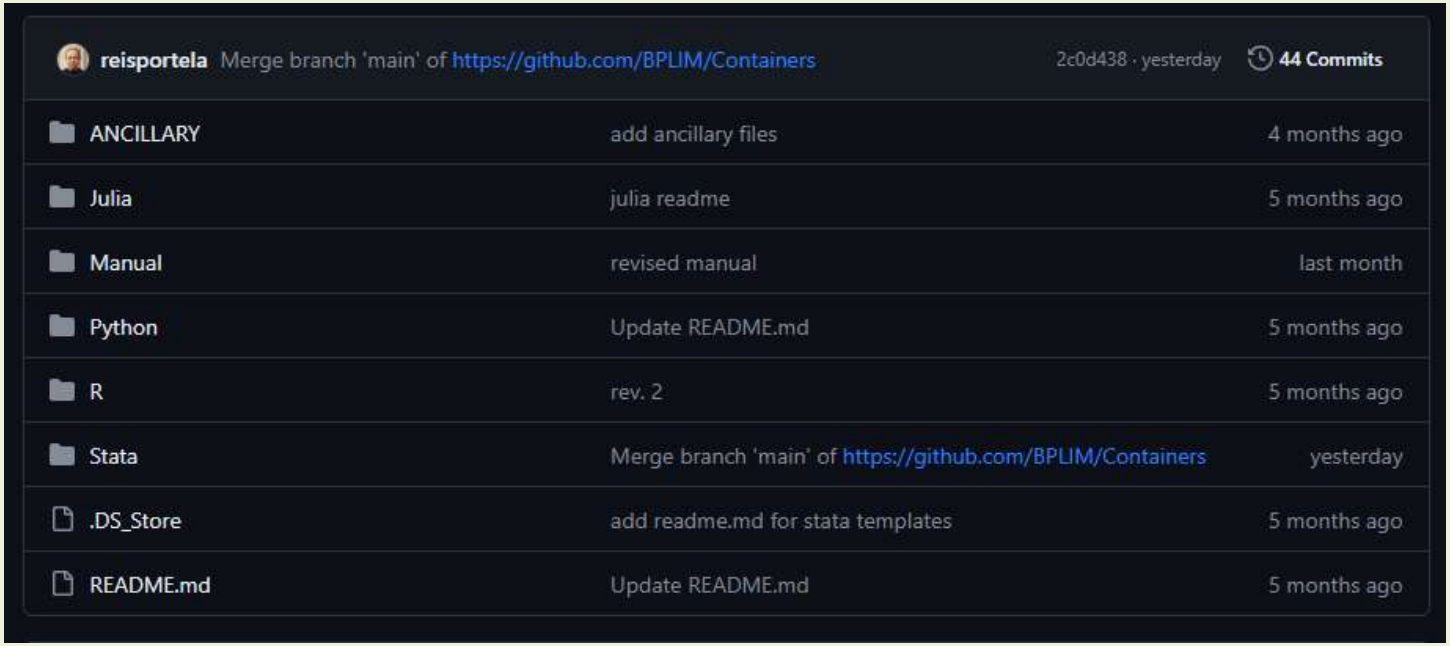
The concept of a definition file

- “Text document that serves as a blueprint for creating a Singularity container image. This file, typically having a .def extension, contains specific instructions and settings for the container. It outlines the base environment, including the base OS, any required applications, libraries, and dependencies.” (ChatGPT, 2023)
- A detailed manual on how to build and use containers is available at BPLIM’s GitHub:

https://github.com/BPLIM/Containers/tree/main/Manual

How to build a container?

Definition files are available at BPLIM's GitHub: https://github.com/BPLIM/Containers



reisportela Merge branch 'main' of https://github.com/BPLIM/Containers	2c0d438 · yesterday	44 Commits
ANCILLARY	add ancillary files	4 months ago
Julia	julia readme	5 months ago
Manual	revised manual	last month
Python	Update README.md	5 months ago
R	rev. 2	5 months ago
Stata	Merge branch 'main' of https://github.com/BPLIM/Containers	yesterday
.DS_Store	add readme.md for stata templates	5 months ago
README.md	Update README.md	5 months ago

Remote server: Git is available

The concept

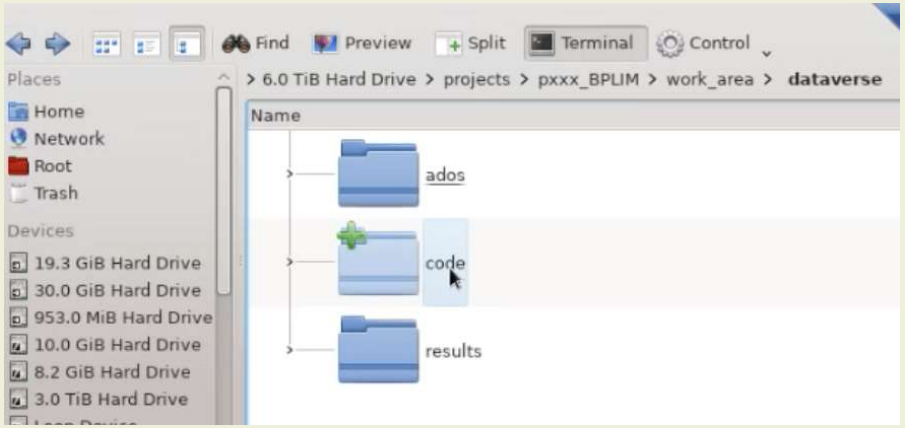
- “Git is a distributed version control system, primarily used for source code management in software development. It allows multiple developers to work on the same project simultaneously without interfering with each other’s changes. Git tracks the progress of changes in a series of snapshots, enabling users to revert back to previous versions of their work if necessary. It’s known for its speed, data integrity, and support for distributed, non-linear workflows.” (ChatGPT, 2023)
- A detailed manual on how to setup and use Git in the remote server is available at BPLIM’s GitHub:

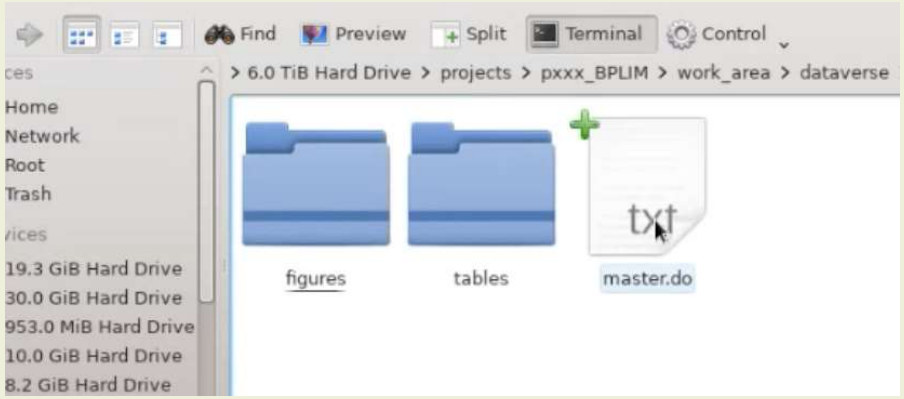
https://github.com/BPLIM/Manuals/tree/master/ExternalServer/Git

Replication App

BPLIM Team developed a tool to streamline the replicability of the research project.

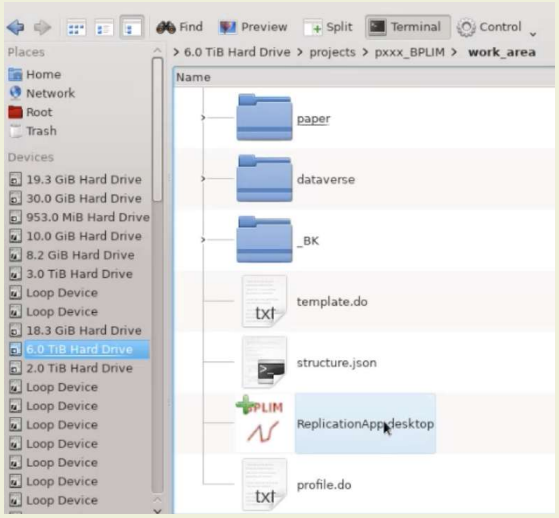
- Research project's folder structure





Replication App

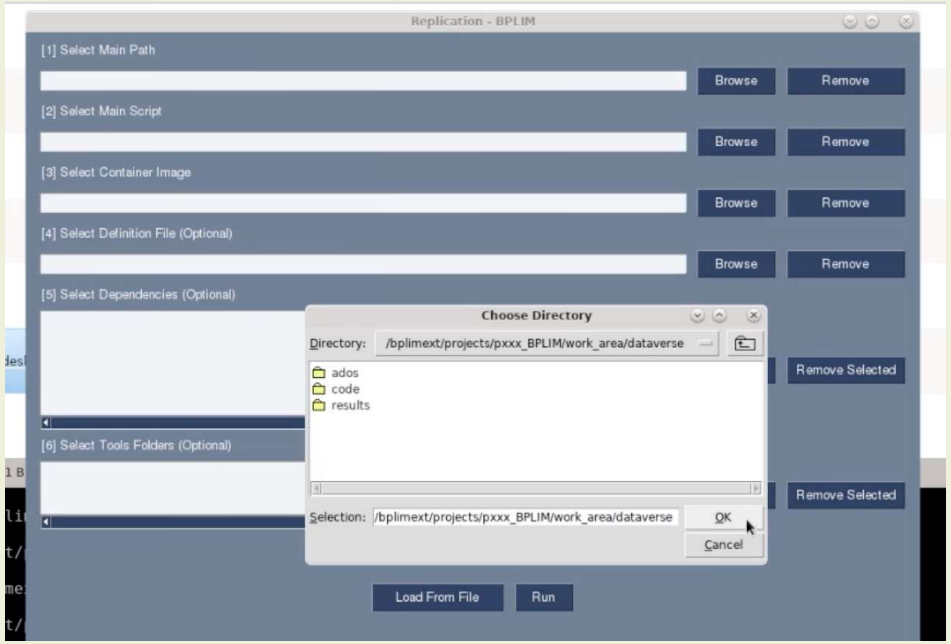
- Using Dolphin, go to `work_area` and click in `ReplicationApp.desktop`

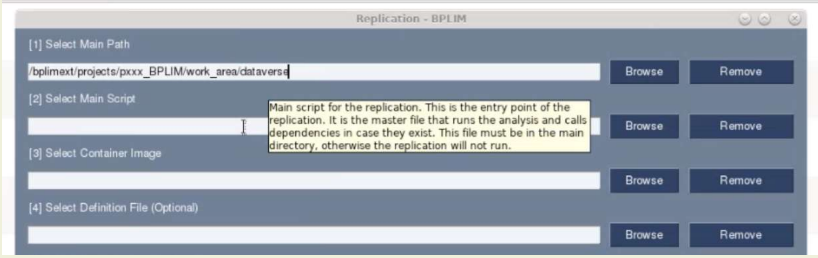


ReplicattionApp icon

Replication App

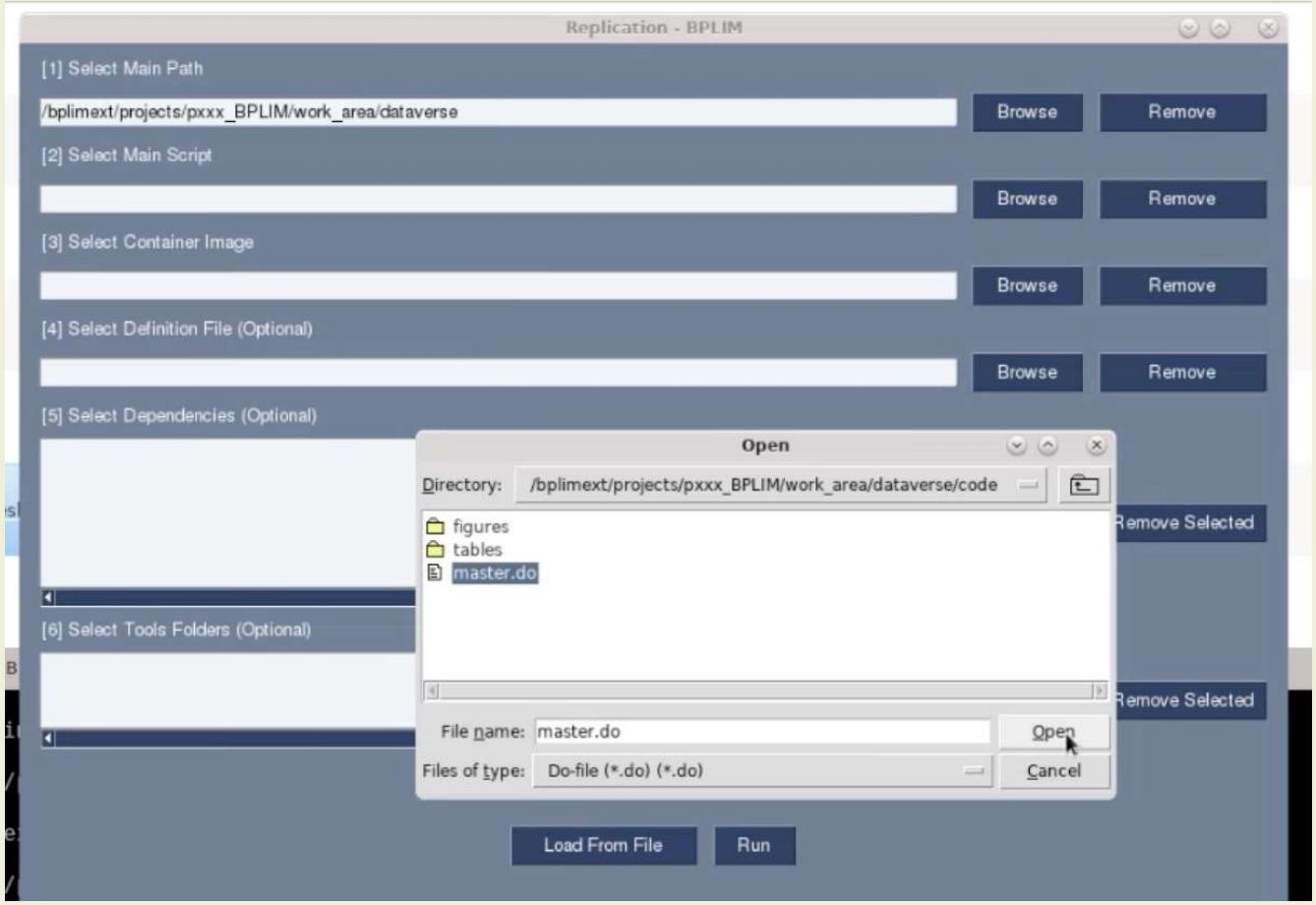
- Fill the boxes with the information from the project





Replication App

- Fill the boxes with the information from the project



Replication App

- `master.do` file

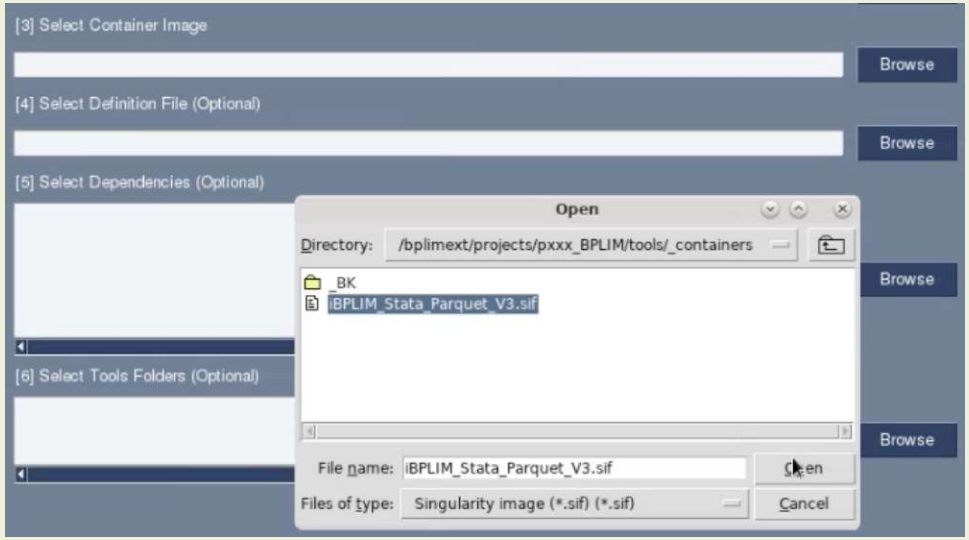
File Edit View Bookmarks Tools Settings Help

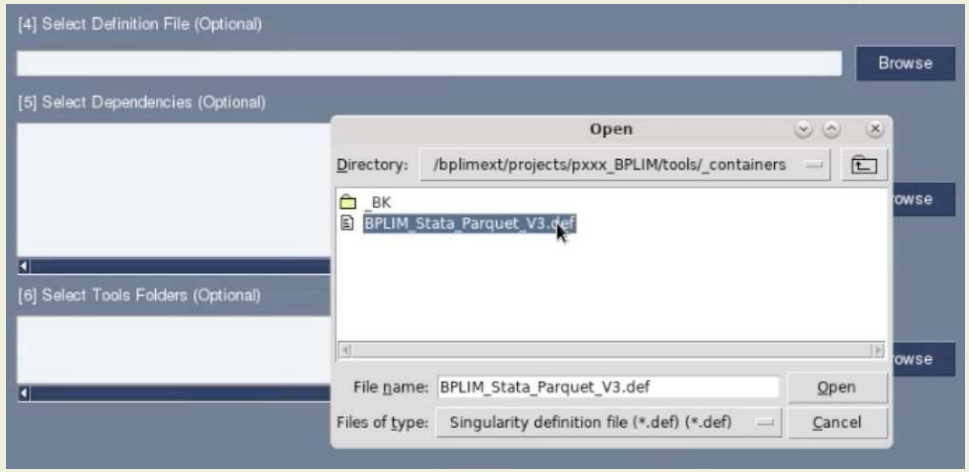
New Open Save Save As Close Undo Redo

```
* Guimaraes, P. and Barbosa, M.
* The state of Portuguese research in economics: 20 years after
* Portuguese Economic Journal, 2022, vol 21, n.3

* Create all figures

do profile.do

cd "${path_rep}"

capture mkdir results
capture mkdir results/tables
capture mkdir results/figures
program drop _all
clear

cd code/figures/
do Figures.do

cd ../tables/
do Tables.do
```

Replication App

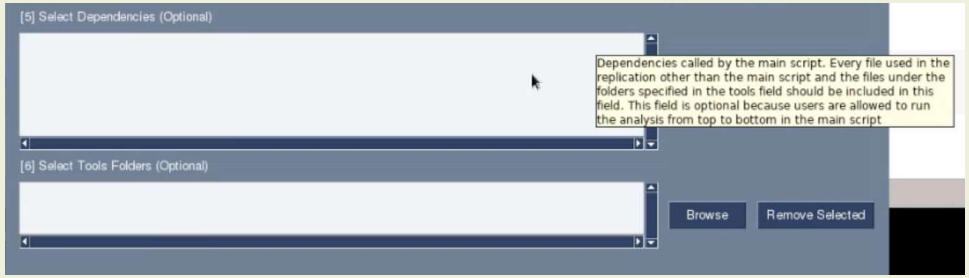
- Fill the boxes with the information from the project: Container and definition file

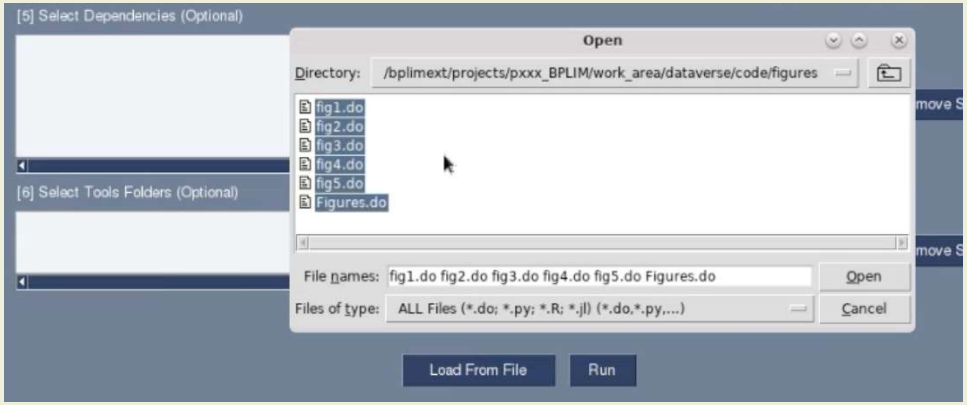




Replication App

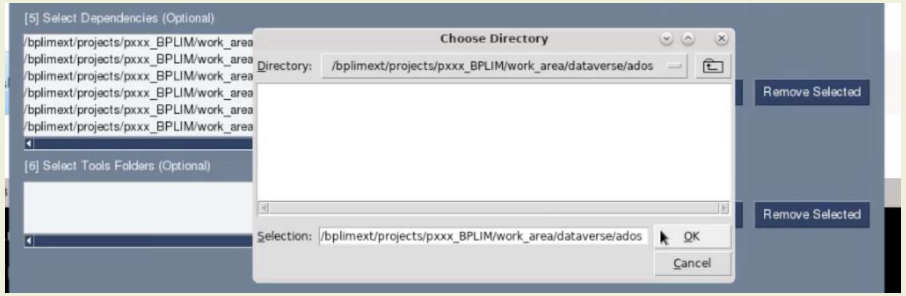
- Fill the boxes with the information from the project: Dependencies

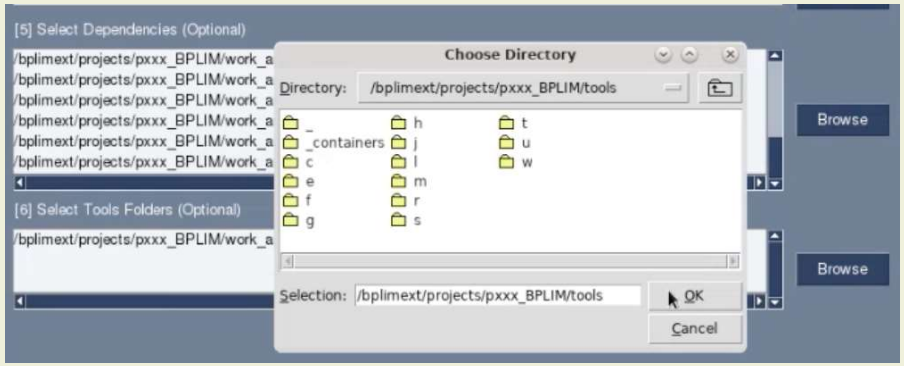




Replication App

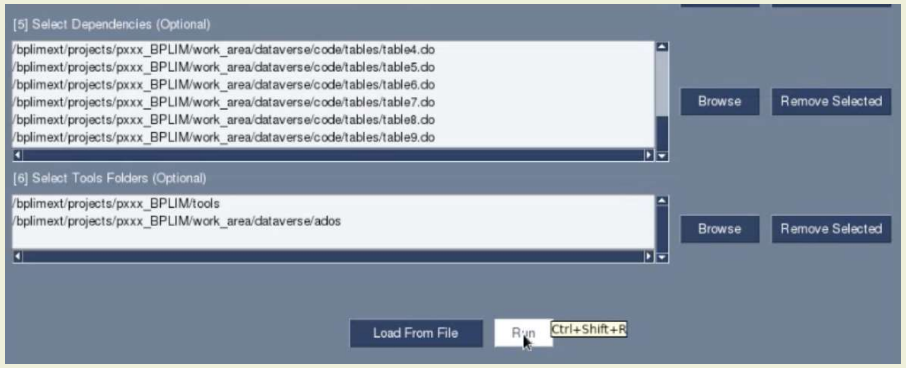
- Fill the boxes with the information from the project: Tools

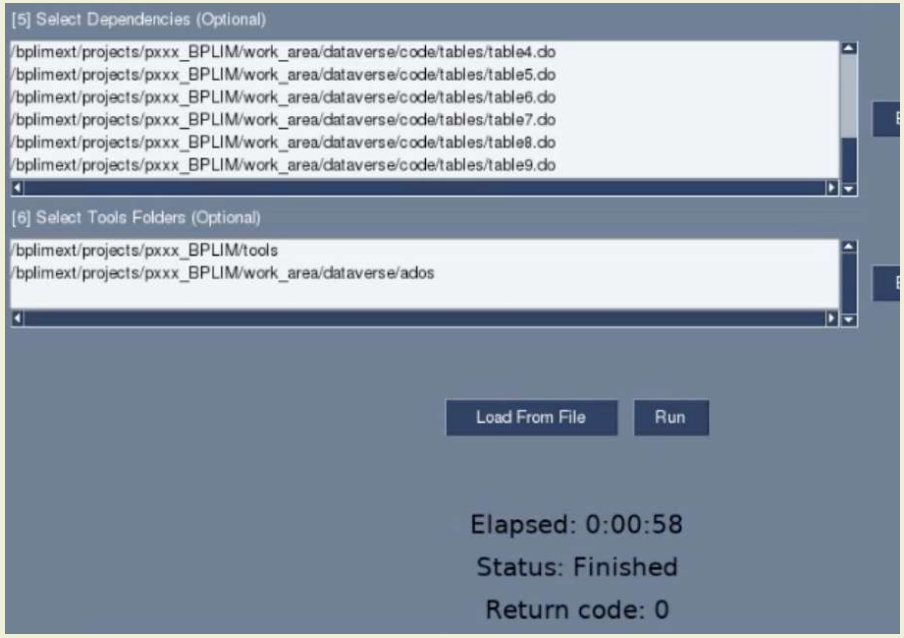




Replication App

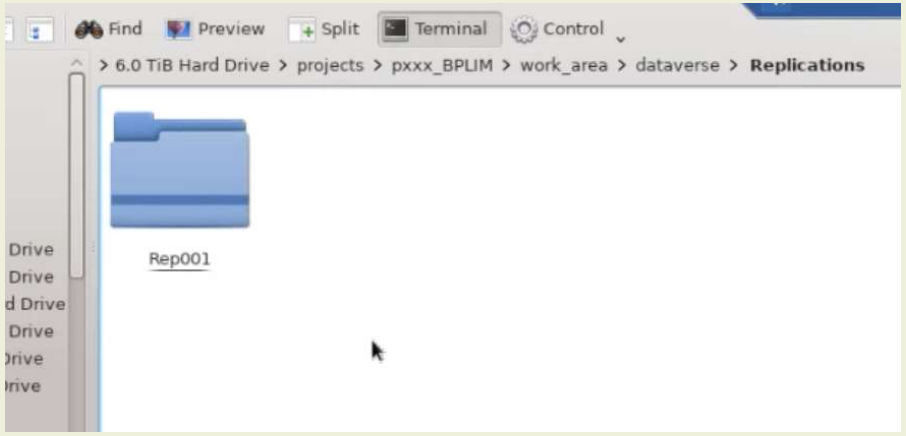
- Fill the boxes with the information from the project: Run

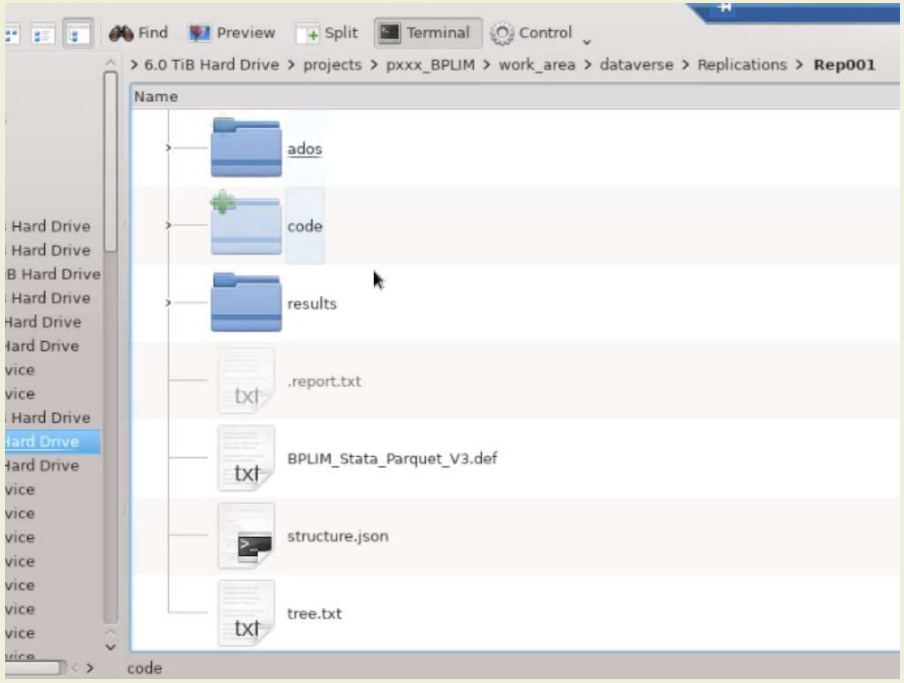




Replication App

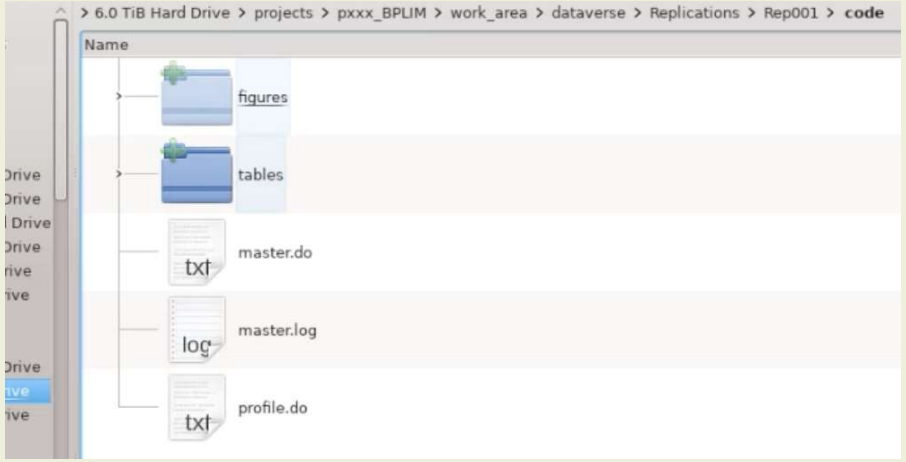
- Fill the boxes with the information from the project: Replication output

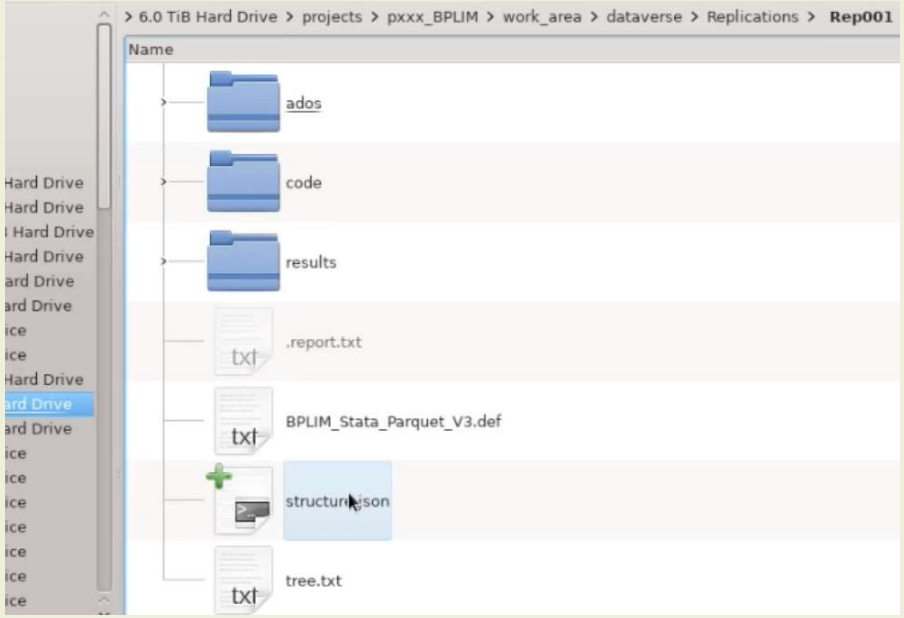




Replication App

- Fill the boxes with the information from the project: Replication output





Replication App

- Fill the boxes with the information from the project: Replication output

File Edit View Bookmarks Tools Settings Help

New Open Save Save As Close Undo Redo

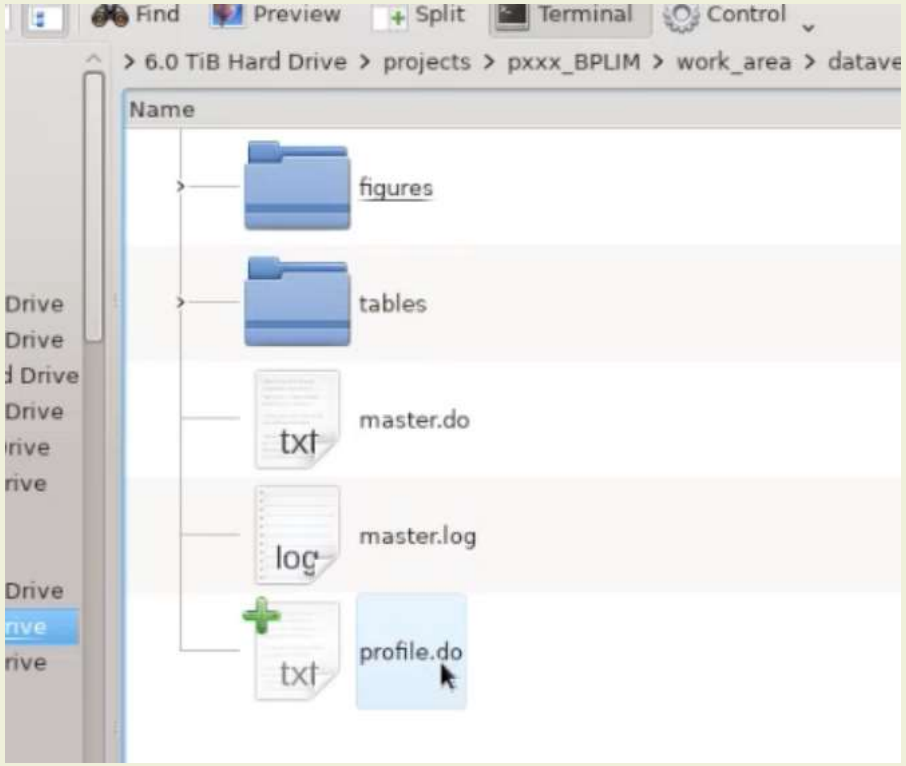

```
{
    "mainFolderInput": "/bplimext/projects/pxxx_BPLIM/work_area/dataverse",
    "mainScriptInput": "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/master.do",
    "containerImage": "/bplimext/projects/pxxx_BPLIM/tools/_containers/iBPLIM_Stata_Parquet_V3.sif",
    "containerDefinition": "/bplimext/projects/pxxx_BPLIM/tools/_containers/BPLIM_Stata_Parquet_V3.def",
    "dependencies": [
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/figures/Figures.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/figures/fig1.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/figures/fig2.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/figures/fig3.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/figures/fig4.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/figures/fig5.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/Tables.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table0.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table1.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table10.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table11.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table12.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table13.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table2.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table3.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table4.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table5.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table6.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table7.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table8.do",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/code/tables/table9.do"
    ],
    "tools": [
        "/bplimext/projects/pxxx_BPLIM/tools",
        "/bplimext/projects/pxxx_BPLIM/work_area/dataverse/ados"
    ]
}
```

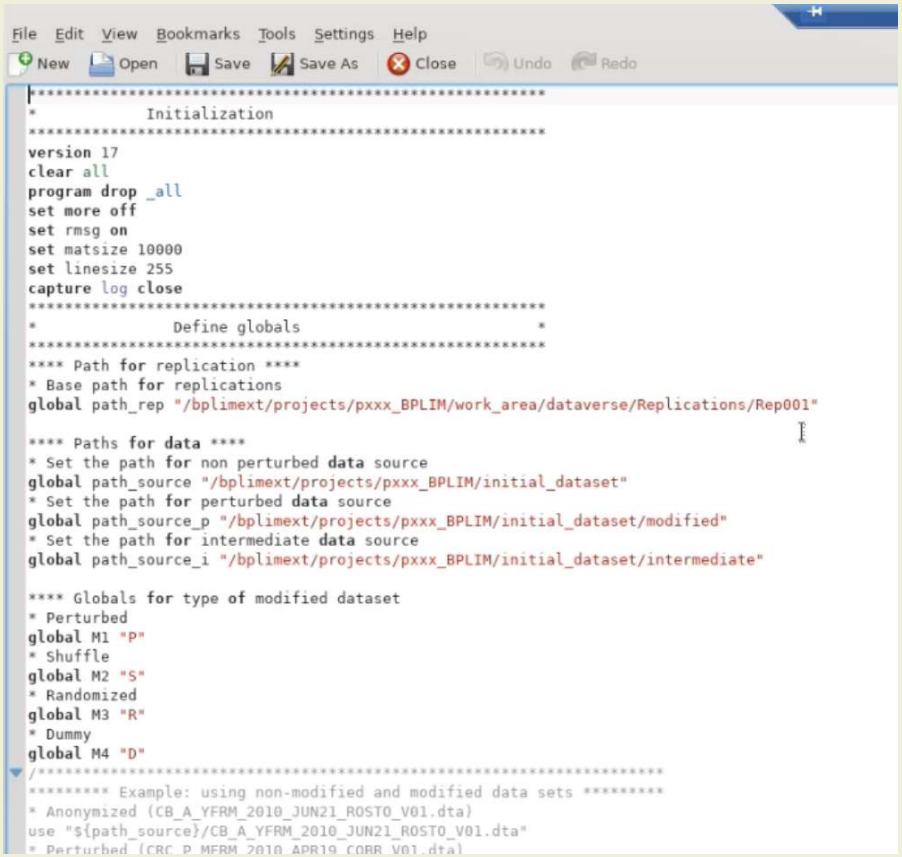

Open

```
Root: /bplimext/projects/pxxx_BPLIM/work_area/dataverse/Replications/Rep001
├── BPLIM_Stata_Parquet_V3.def
├── structure.json
├── ados/
│   ├── addcuraff.ado
│   ├── createvarsacad.ado
│   └── createvarspub.ado
├── code/
│   ├── master.do
│   ├── profile.do
│   ├── figures/
│   │   ├── Figures.do
│   │   ├── fig1.do
│   │   ├── fig2.do
│   │   ├── fig3.do
│   │   ├── fig4.do
│   │   └── fig5.do
│   └── tables/
│       ├── Tables.do
│       ├── table0.do
│       ├── table1.do
│       ├── table10.do
│       ├── table11.do
│       ├── table12.do
│       ├── table13.do
│       ├── table2.do
│       ├── table3.do
│       ├── table4.do
│       ├── table5.do
│       ├── table6.do
│       ├── table7.do
│       ├── table8.do
│       └── table9.do
└── results/
    ├── figures/
    └── tables/

7 directories, 28 files
```

Replication App

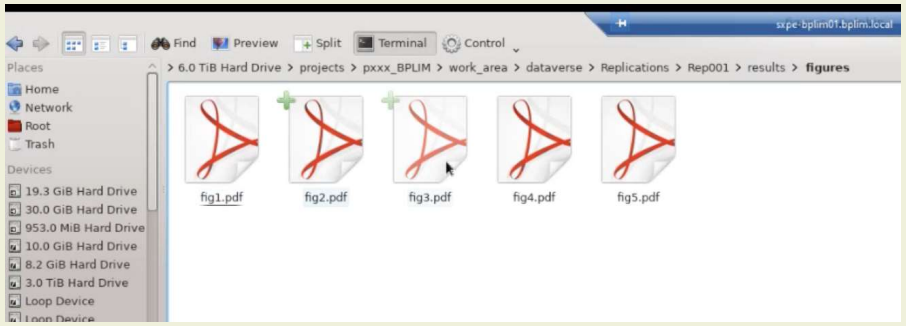
- Fill the boxes with the information from the project: Replication output

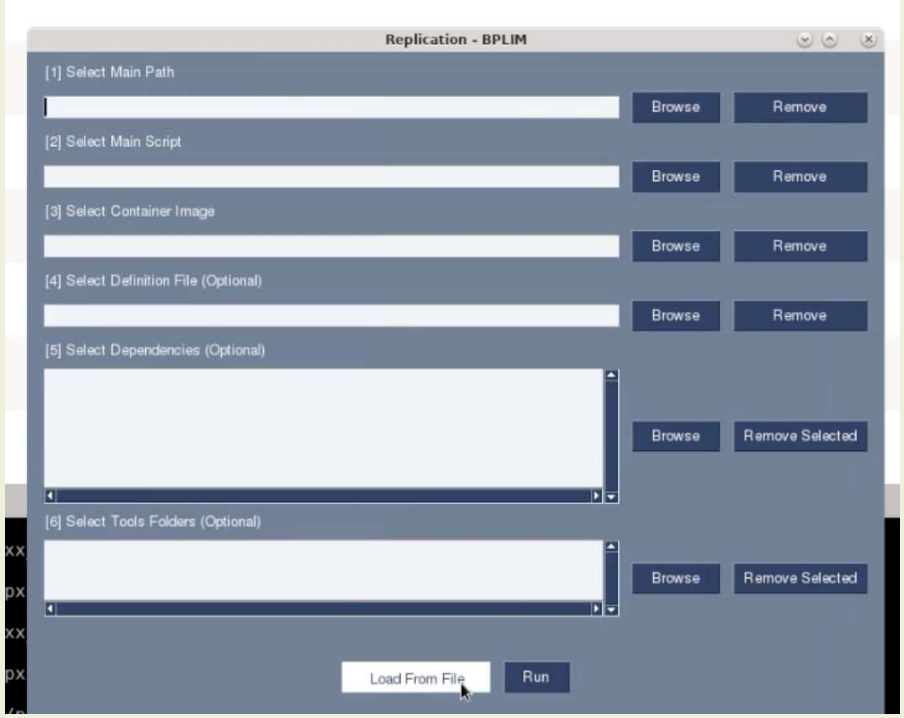




Replication App

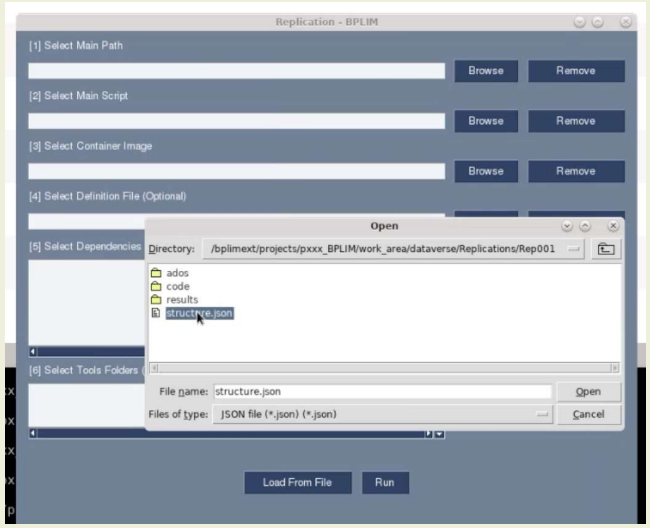
- Fill the boxes with the information from the project: Replication output

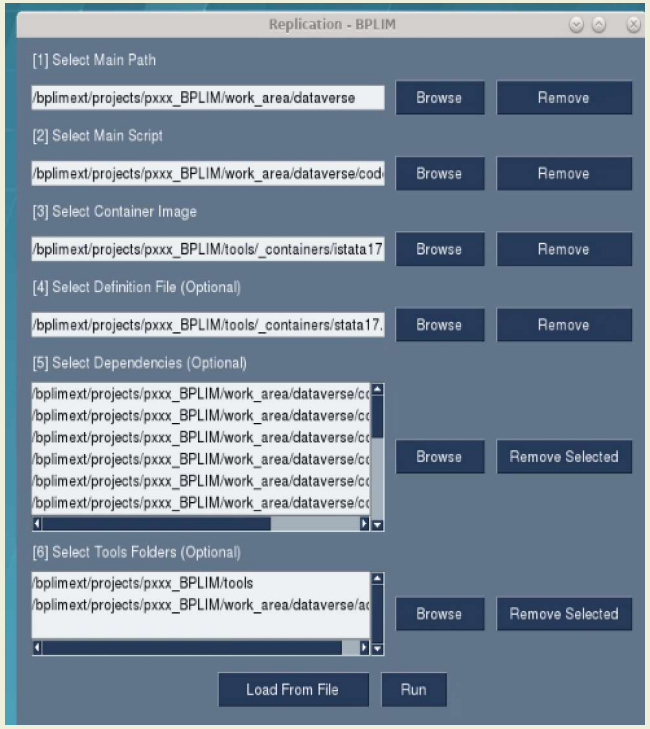




Replication App: `json` file

In `work_area` folder the file `structure.json` has the different sets of information





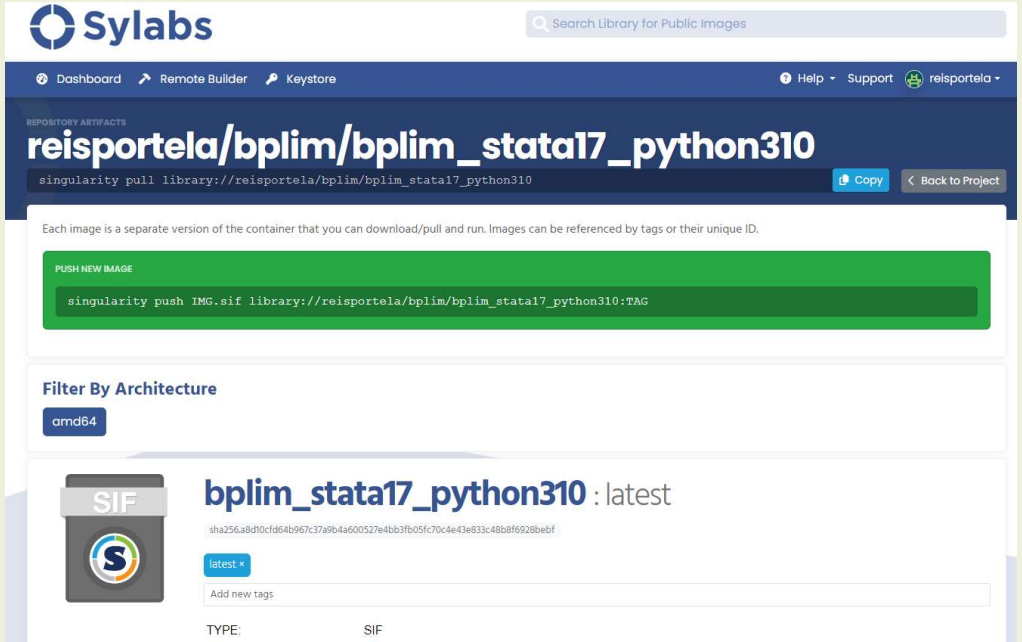
Replication App: Outcomes

- Folder `ados`: ado files programmed by the researcher.
- Folder `code`: contains the code used to replicate all the analysis performed by the researcher.
- Folder `results`: outcomes of the statistical analysis. This is the folder that will be shared with the researcher after output control.

Appendix

How to build a container?

Using the container available in Sylabs



Library for Public Images:
https://cloud.sylabs.io/library/reisportela/bplim/bplim_stata17_python310

How to build a container?

Build your container using Sylabs

1. Go to Sylabs, https://cloud.sylabs.io/, Sign up and Sign in
2. Go to Remote Builder
3. Copy/paste the definition file into the text box
4. Give a name to the container and click in Submit Build

How to build a container?

Build your container using your local machine

1. Use the following definition file as a template

`BPLIM_Stata17_Python310_from_Sylabs_V4.def`

2. To build the container you must have a valid Stata 17 license
3. When building the container the file `Stata_ados_BASE.do` is used to install the ado files you need
4. In case you need additional Linux packages in your container they can be added in the section `%post` of the definition file. See further details at https://github.com/BPLIM/Containers/tree/main/Stata

Big data in Stata: parquet files

The use of parquet files is made available by Mauricio Caceres and can be used in the remote server

1. Open a Terminal
2. Launch the container using the the command line
3. See the example that opens a Stata file, saves it as parquet and reads the parquet file