A toolbox for VAR analysis

Developed by Ambrogio Cesa-Bianchi

Link: https://sites.google.com/site/ambropo/MatlabCodes

Prepared by Cesar Ramos

Table of Contents

General info

The VAR Toolbox is a collection of Matlab codes to perform Vector Autoregression (VAR) analysis. Estimation is performed with OLS. The VAR Toolbox allows for identification of structural shocks with:

- zero short-run restrictions;
- zero long-run restrictions;
- sign restrictions;
- external instruments (proxy SVAR); and a
- combination of external instruments and sign restrictions.

Also provides:

- Impulse Response Functions (IR),
- Forecast Error Variance Decomposition (VD),
- and Historical Decompositions (HD) are computed according to the chosen identification.
- Confidence intervals are obtained with bootstrapping methods.

Description

The VAR Toolbox is a collection of Matlab routines to perform vector autoregressive (VAR) analysis. The latest version is available in the v3dot0 folder.

The codes are grouped in six categories (and respective folders):

- VAR: the codes for VAR estimation, identification, computation of the impulse response functions, FEVD, HD.

- Stats: codes for the calculation of summary statistics, moving correlations, pairwise correlations, etc.
- Utils: codes that allow the smooth functioning if the Toolbox.
- Auxiliary: codes that I borrowed from other public sources. Each m-file has a reference to the original source.
- Figure: codes for plotting high quality figures
- ExportFig: this is a toolbox developed by Yair Altman (https://github.com/altmany/export_fig) for exporting high quality figures. To enable this option, the Toolbox requires Ghostscript installed on your computer (freely available at www.ghostscript.com).

Installation

No installation is required. Simply clone the folder from Github and add the folder (with subfolders) to your Matlab path. This can be easily done as follows.

If you download the toolbox to `/User/VAR-Toolbox/`, you can simply add the following two lines of code at the beginning and end of your script

```
addpath(genpath('/User/VAR-Toolbox/v3dot0/'))
...
rmpath(genpath('/User/VAR-Toolbox/v3dot0'))
```

To save figures in high quality format, you need to download an install Ghostscript (freely available at www.ghostscript.com). The first time you'll be saving a figure using the Toolbox, you'll be asked to locate Ghostscript on your local drive.

Manual

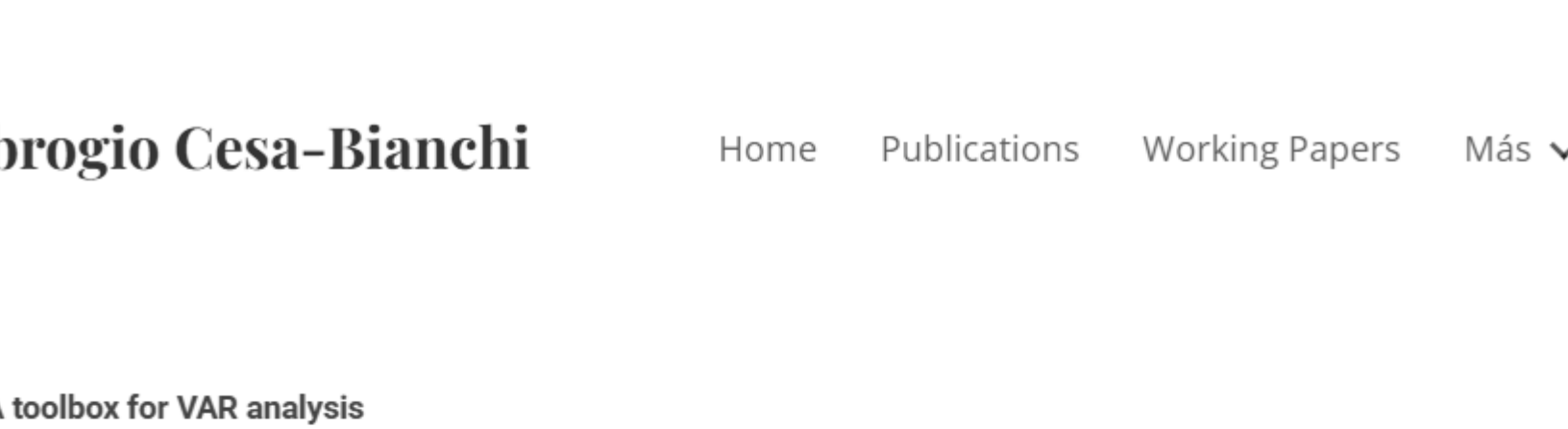
A manual will be available soon. In the meanwhile, you can information on how to use the VAR Toolbox in the VAR Primer, a slide deck describing the basics of VARs and how to estimate them using the VARToolbox.

Steps to Download and Add the Toolbox to MATLAB

Step 1

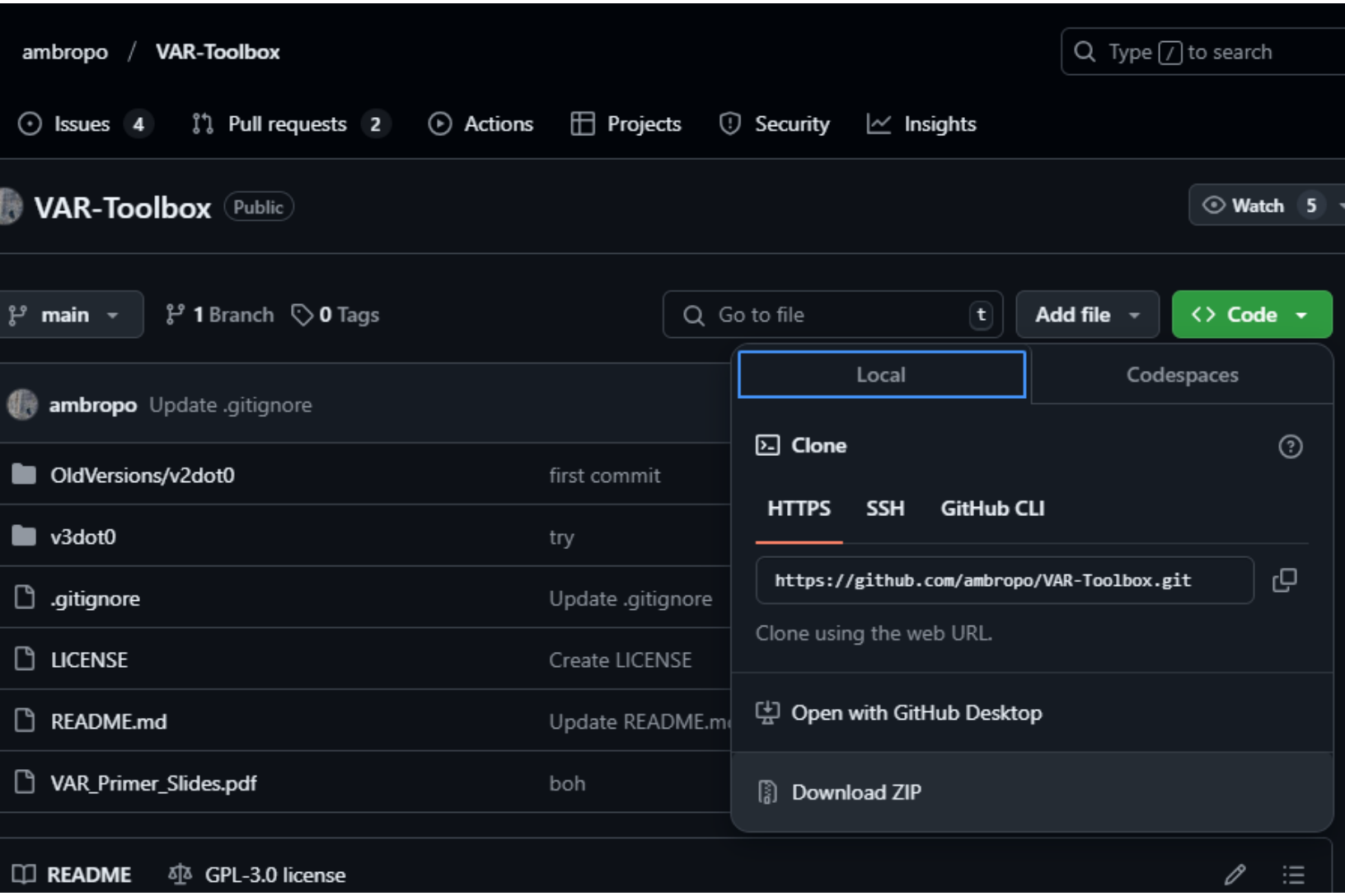
Enter to the link: https://sites.google.com/site/ambropo/MatlabCodes

Then click on [VAR Toolbox 3.0]



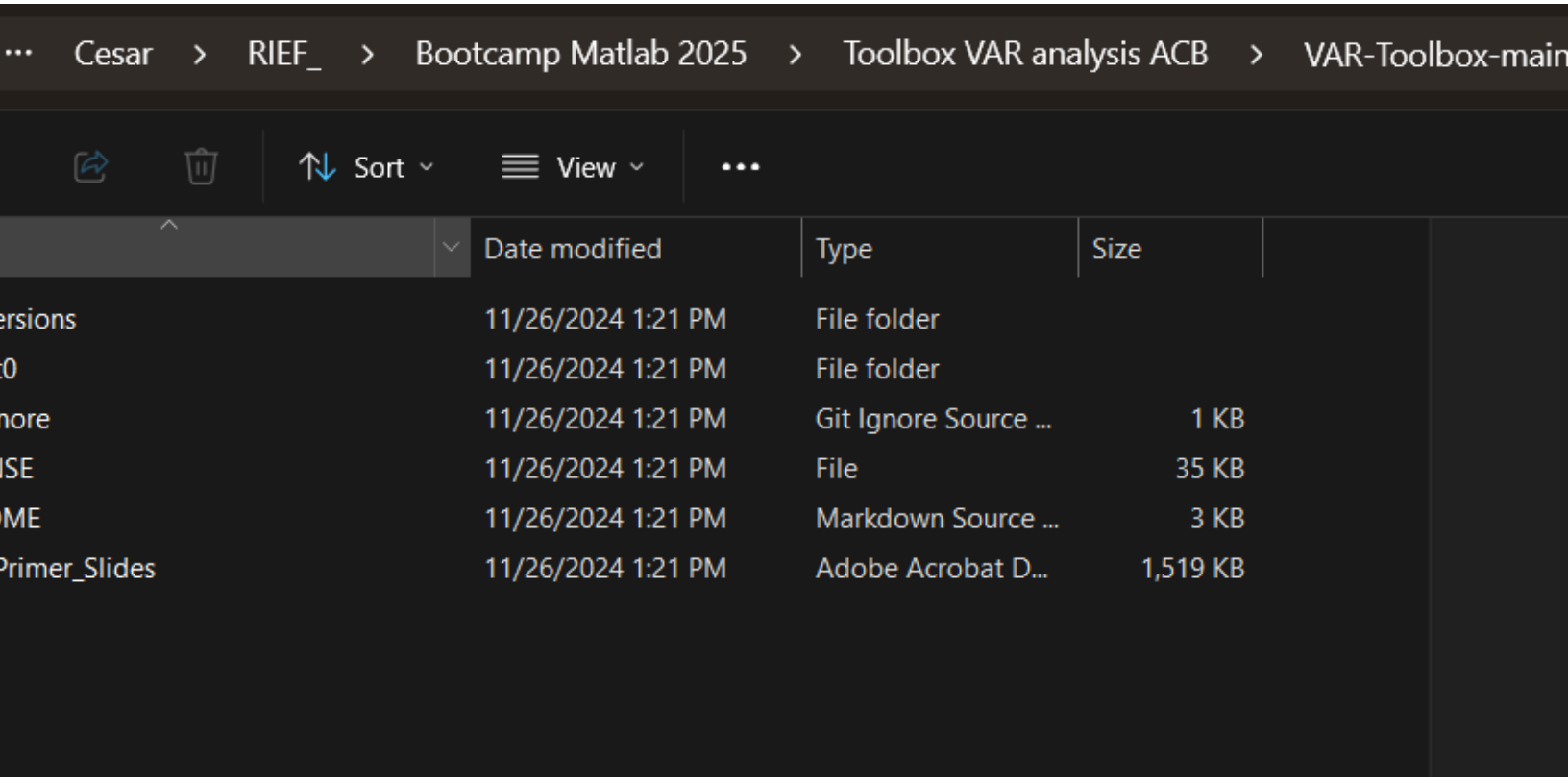
Step 2

Click on **`Code`**, and **`Download ZIP`**



Step 3

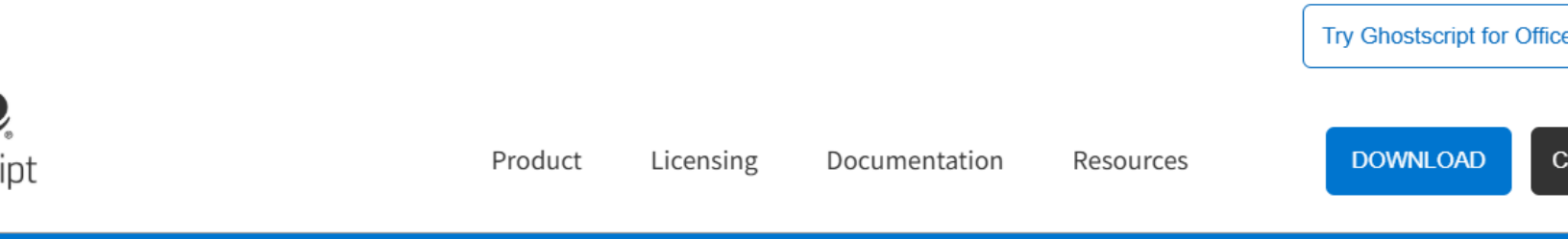
After downloading, locate the **`.ZIP`** file and copy it to a preferred folder. Then, extract the file, and you should have the following files in the folder named **`VAR-Toolbox-main`**.



Step 4

Download and install `Ghostscript`. ExportFig: The ExportFig feature is included in the toolbox for **exporting high-quality figures**. To enable this functionality, you need to install `Ghostscript` on your computer.

Ghostscript is freely available at www.ghostscript.com.



Then selected the `Postscript and PDF interpreter/renderer:` Ghostscript

For a full list of fixes and enhancements, please see the release notes.

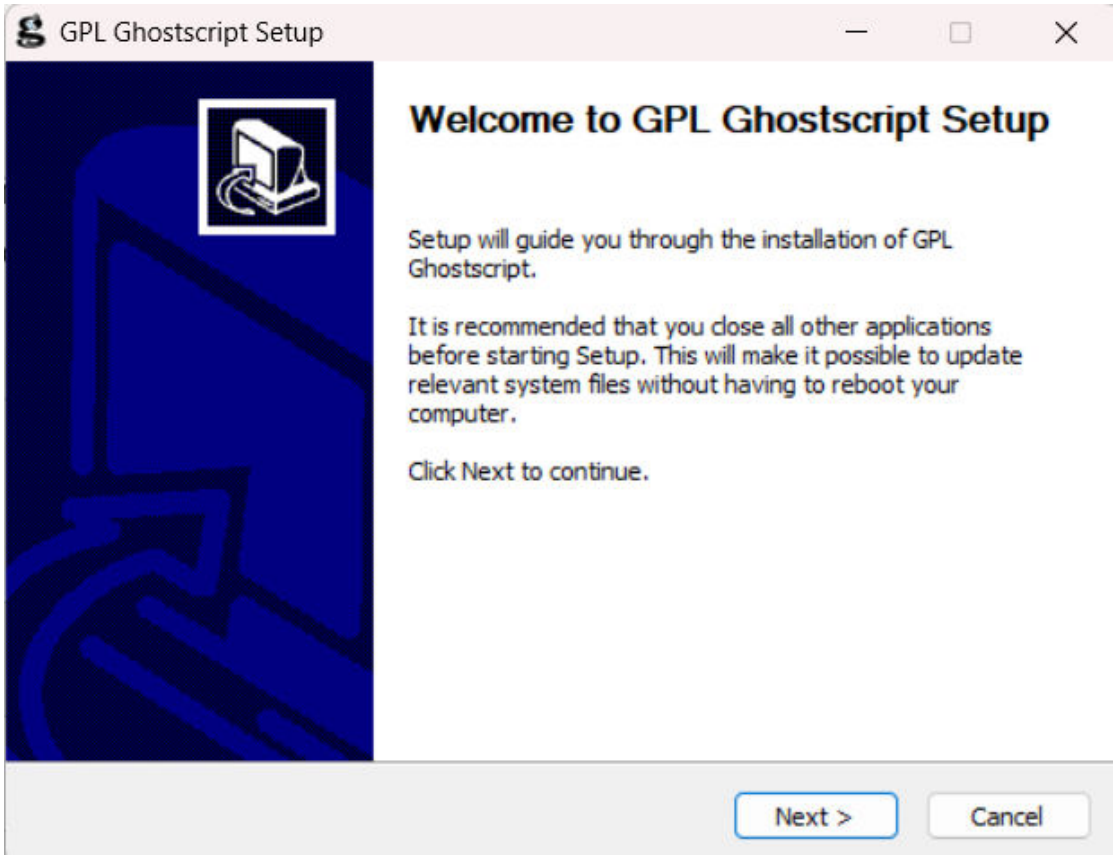
Downloads

The latest AGPL and commercial downloads for the Ghostscript family of products.

Postscript and PDF interpreter/renderer:
Ghostscript

Select the appropriate option based on your operating system (Linux or Windows), then download and install:

Platform/License	GNU Affero General Public License
Ghostscript 10.04.0 for Windows (32 bit)	Ghostscript AGPL Release
Ghostscript 10.04.0 for Windows (64 bit)	Ghostscript AGPL Release
Ghostscript 10.04.0 snap for Linux x86 (64 bit)	Ghostscript AGPL Release
Ghostscript 10.04.0 Source for all platforms	Ghostscript AGPL Release



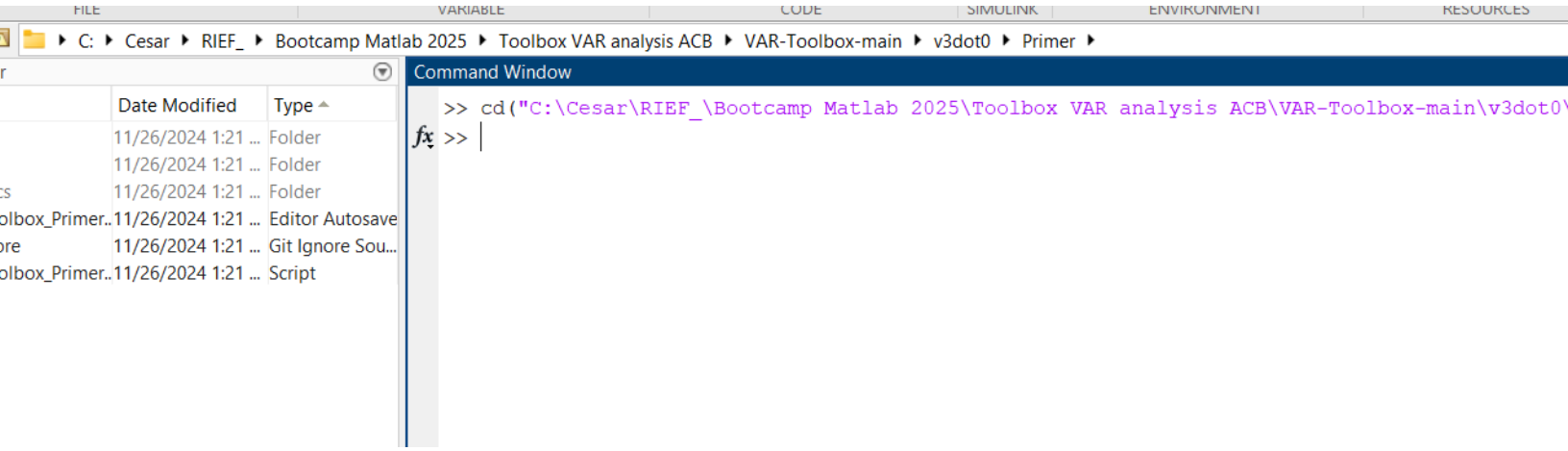
Step 5

Open MATLAB, navigate to the **`v3dot0`** folder, then to the **`Primer`** folder. Copy the folder path and paste it into the **Command Window** as follows:

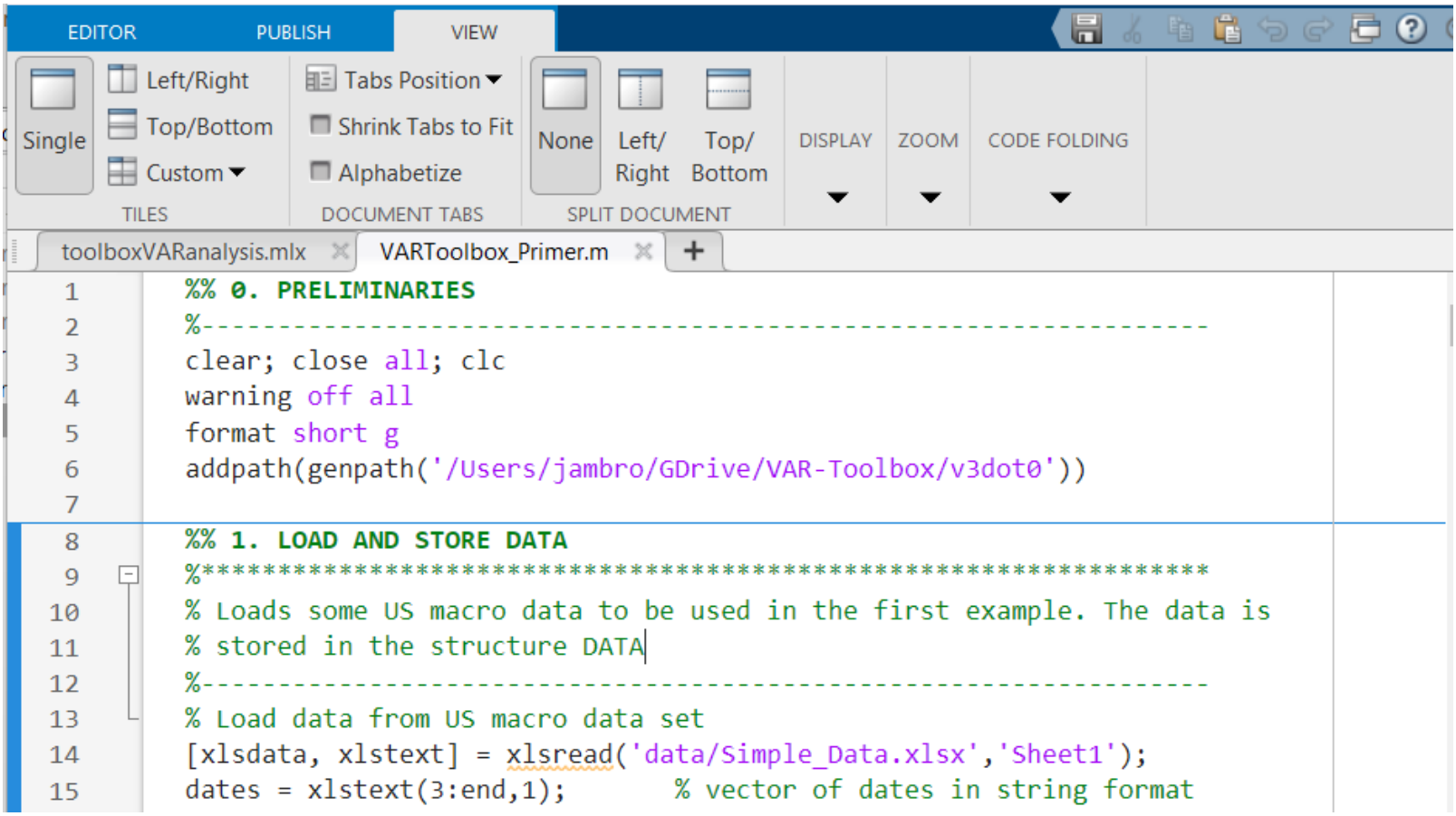
- VAR-Toolbox-main
- v3dot0

- Primer

```
cd("C:\Cesar\RIEF_\Bootcamp Matlab 2025\Toolbox VAR analysis ACB\VAR-Toolbox-main\v3dot0\Primer")
```



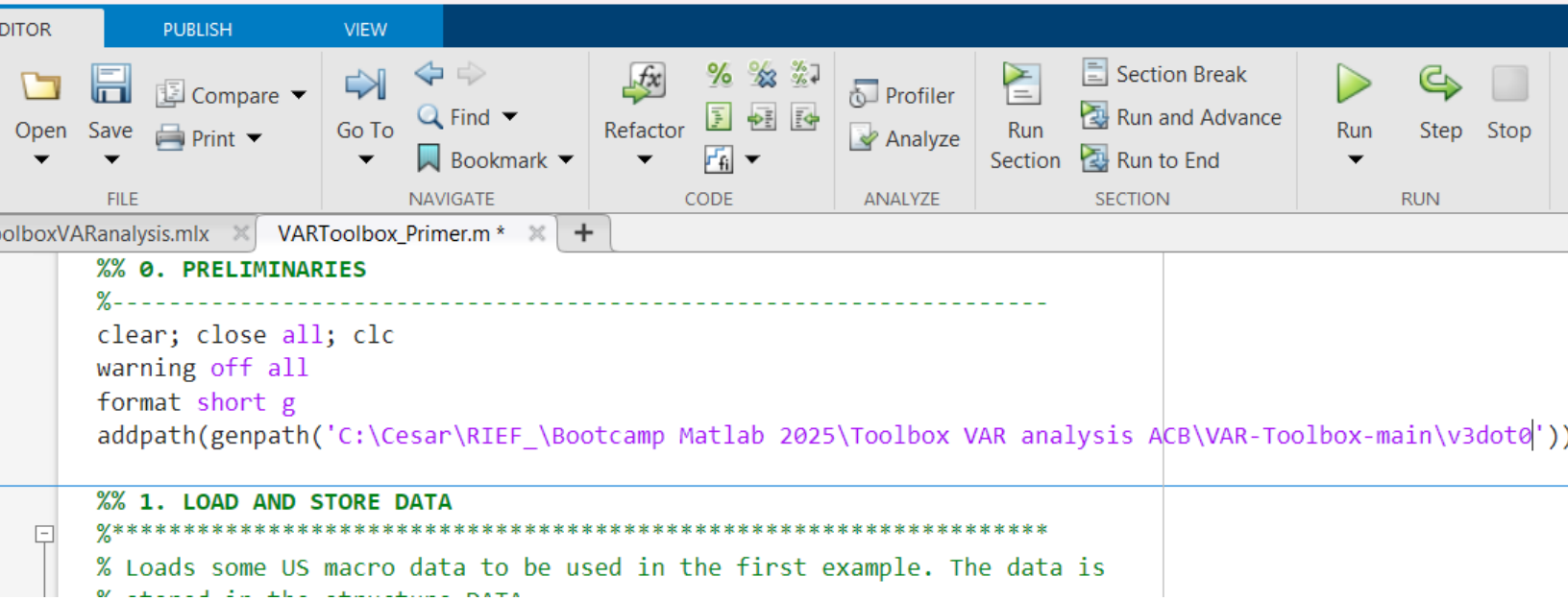
Open the file **`VARToolbox_Primer.m`**



Next, update the **path in line 6** by copying the location of the **`VAR-Toolbox-main\v3dot0 folder`**.

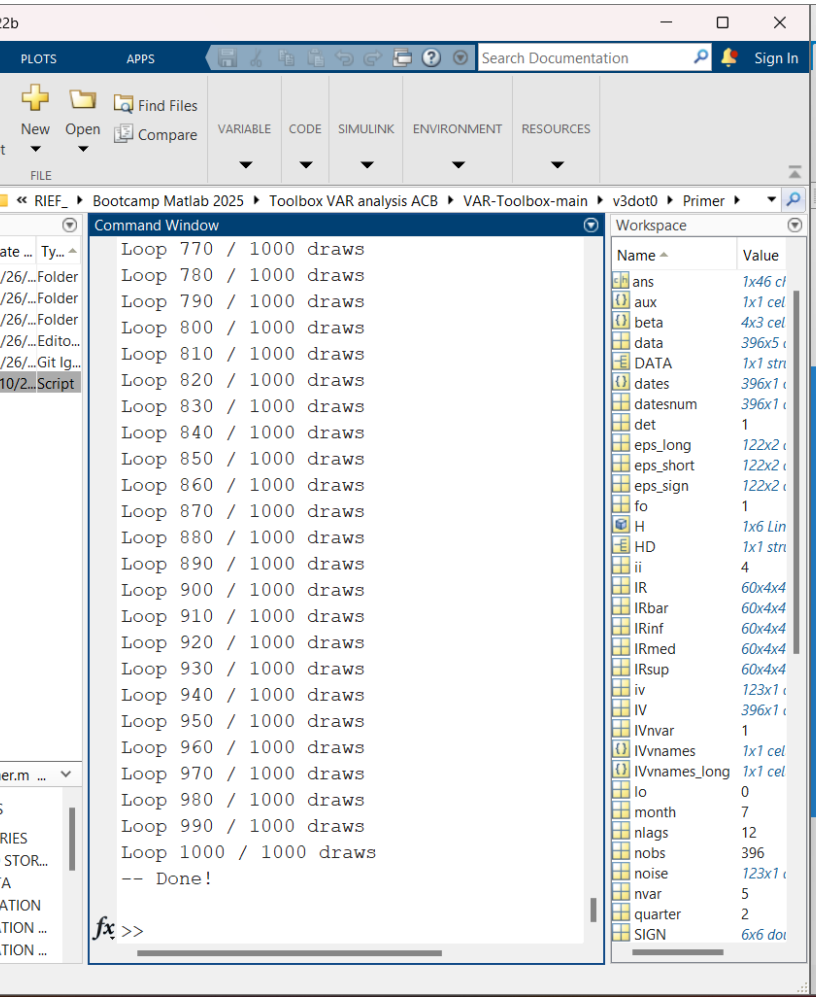
For example, in my case, the path is:

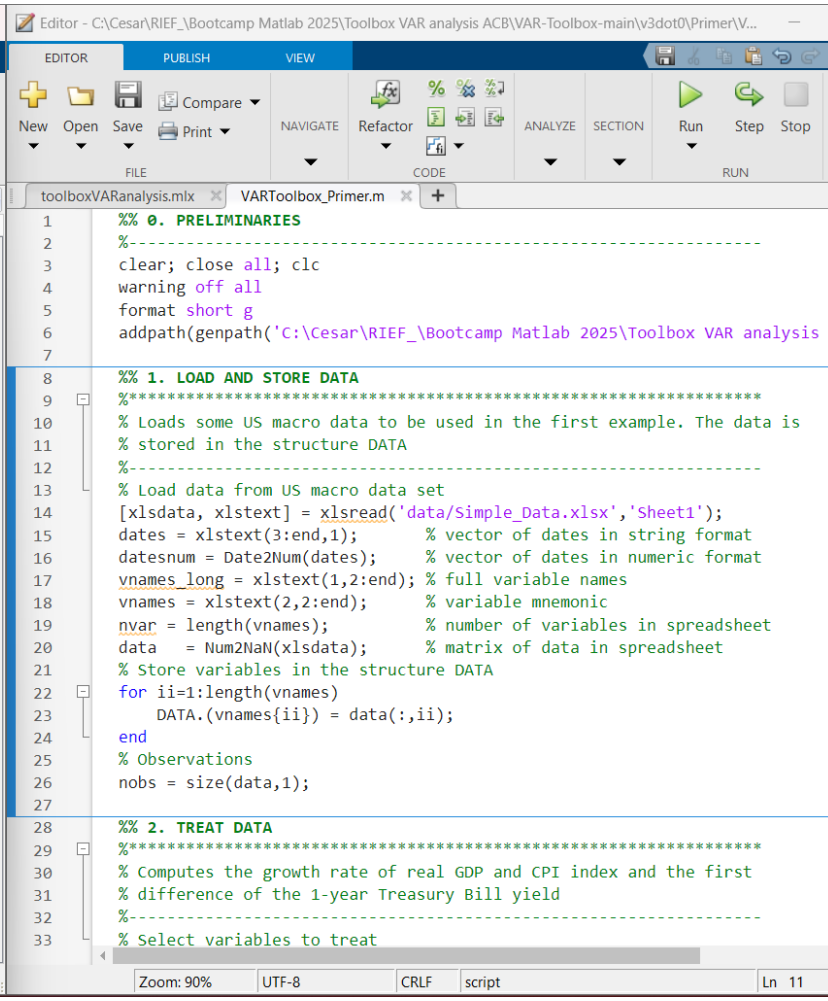
C:\Cesar\RIEF_\Bootcamp Matlab 2025\Toolbox VAR analysis ACB\VAR-Toolbox-main\v3dot0



Step 6

The VAR Toolbox is now ready to be executed!


Command Window
Loop 770 / 1000 draws
Loop 780 / 1000 draws
Loop 790 / 1000 draws
Loop 800 / 1000 draws
Loop 810 / 1000 draws
Loop 820 / 1000 draws
Loop 830 / 1000 draws
Loop 840 / 1000 draws
Loop 850 / 1000 draws
Loop 860 / 1000 draws
Loop 870 / 1000 draws
Loop 880 / 1000 draws
Loop 890 / 1000 draws
Loop 900 / 1000 draws
Loop 910 / 1000 draws
Loop 920 / 1000 draws
Loop 930 / 1000 draws
Loop 940 / 1000 draws
Loop 950 / 1000 draws
Loop 960 / 1000 draws
Loop 970 / 1000 draws
Loop 980 / 1000 draws
Loop 990 / 1000 draws
Loop 1000 / 1000 draws
-- Done!
>>
Workspace
Name
Value


Editor - C:\Cesar\RIEF_\Bootcamp Matlab 2025\Toolbox VAR analysis ACB\VAR-Toolbox-main\v3dot0\Primer\V...
toolboxVARanalysis.mlx
VARToolbox_Primer.m
%% 0. PRELIMINARIES
clear; close all; clc
warning off all
format short g
addpath(genpath('C:\Cesar\RIEF_\Bootcamp Matlab 2025\Toolbox VAR analysis
%% 1. LOAD AND STORE DATA
% Loads some US macro data to be used in the first example. The data is
% stored in the structure DATA
% Load data from US macro data set
[xlsdata, xlstext] = xlsread('data/Simple_Data.xlsx','Sheet1');
dates = xlstext(3:end,1);
datesnum = Date2Num(dates);
vnames_long = xlstext(1,2:end);
vnames = xlstext(2,2:end);
nvar = length(vnames);
data = Num2NaN(xlsdata);
% Store variables in the structure DATA
for ii=1:length(vnames)
DATA.(vnames{ii}) = data(:,ii);
end
% Observations
nobs = size(data,1);
%% 2. TREAT DATA
% Computes the growth rate of real GDP and CPI index and the first
% difference of the 1-year Treasury Bill yield
% Select variables to treat
Zoom: 90%
UTF-8
CRLF
script
Ln 11