IWH
Halle Institute for Economic Research
Member of the Leibniz Association

Dynamic General Equilibrium Model for Climate Resilient Economic Development (DGE-CRED)

Installation and Execution Manual

with Practical Exercises

Global Programme "Policy Advice for Climate Resilient Economic Development" (CRED)

Halle (Saale), December 2021

Table of Contents

1 About this Manual

This manual is intended to guide you through the process of using the DGE-CRED model step by step, from installing the different software needed to executing model simulation to designing and implementing customised scenarios. Furthermore, a section of the manual includes practical exercises so that the user can gain a deeper understanding of how to change variables, run different scenarios and interpret results. No prior experience with *Dynare* or *MATLAB* is needed.

The resources of this manual include written descriptions of each step combined with pictures and diagrams that will help the user understand how to operate the model. Also, a series of *YouTube* video tutorials has been developed to provide a more detailed description of certain sections. A link to each video is provided at the beginning of the sections, which include it. Furthermore, all the videos can be accessed in the next section, which includes a complete list of the tutorials. The videos are intended to be followed in order as they appear in the list.

To use the DGE-CRED model, you will need *Dynare* (at least version 4.6.1) and MATLAB (at least 2018b). While it was possible to run previous versions of the model on the open-source software Octave, the current version does not admit Octave. Thus, we recommend using MATLAB to simulate the model smoothly and avoid high computational costs. Additional software needed for the model includes MS Excel, used to specify parameters and save the results. Furthermore, results tables are saved in LaTeX format.

2 List of Video Tutorials

A list of all the video tutorials that accompany the manual can be found below. You can access them by clicking on the different items.

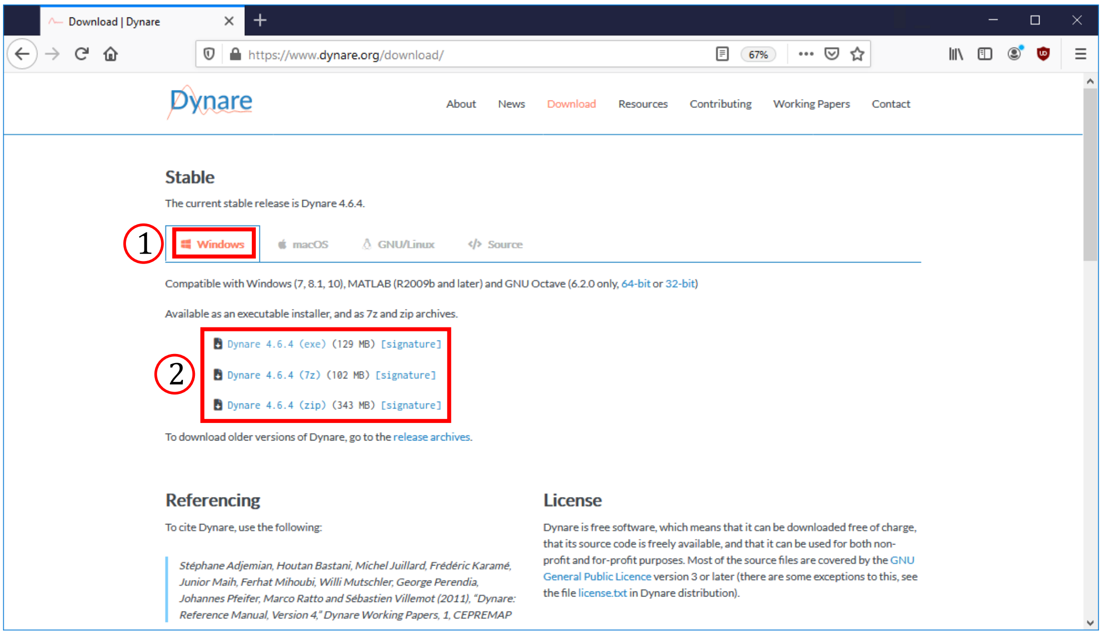
- Installing Dynare on Windows
- Installing Dynare on MacOS
- Downloading the DGE-CRED model files

3 Dynare

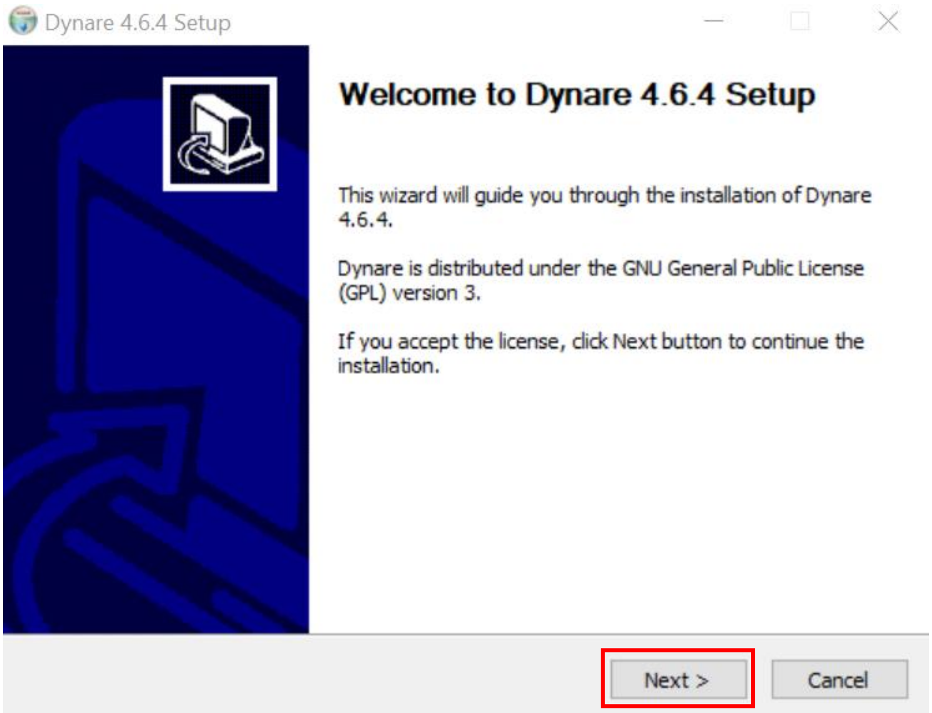
3.1 On Windows

A video tutorial of the installation of *Dynare* on Windows can be accessed here.

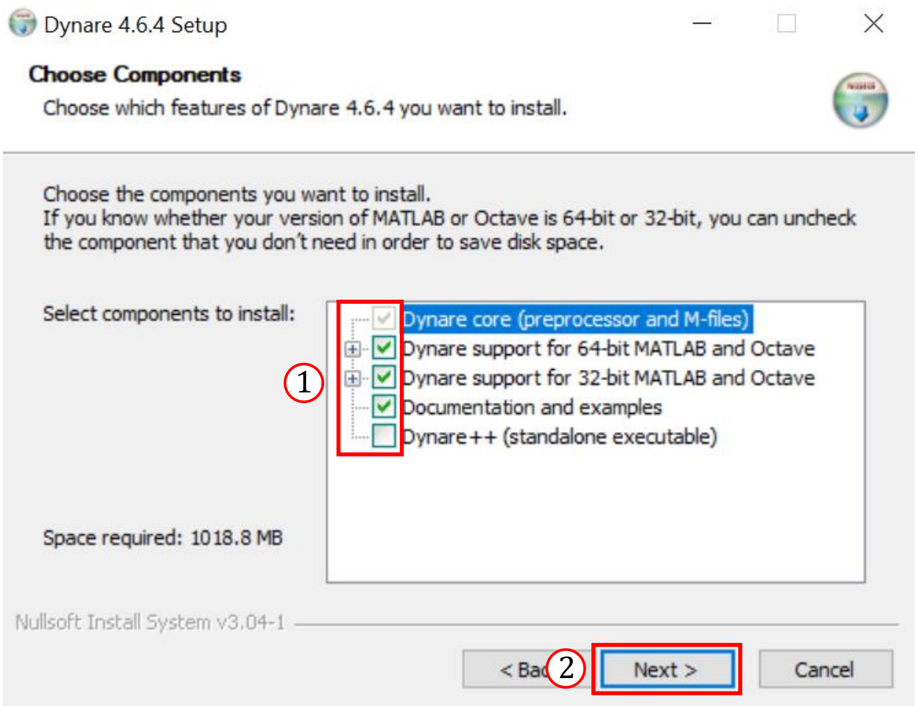
1. Go to https://www.dynare.org/download/ in your internet browser. **(1)** In the windows tab, **(2)** click on one of the download package options and wait until the downloading process is completed.



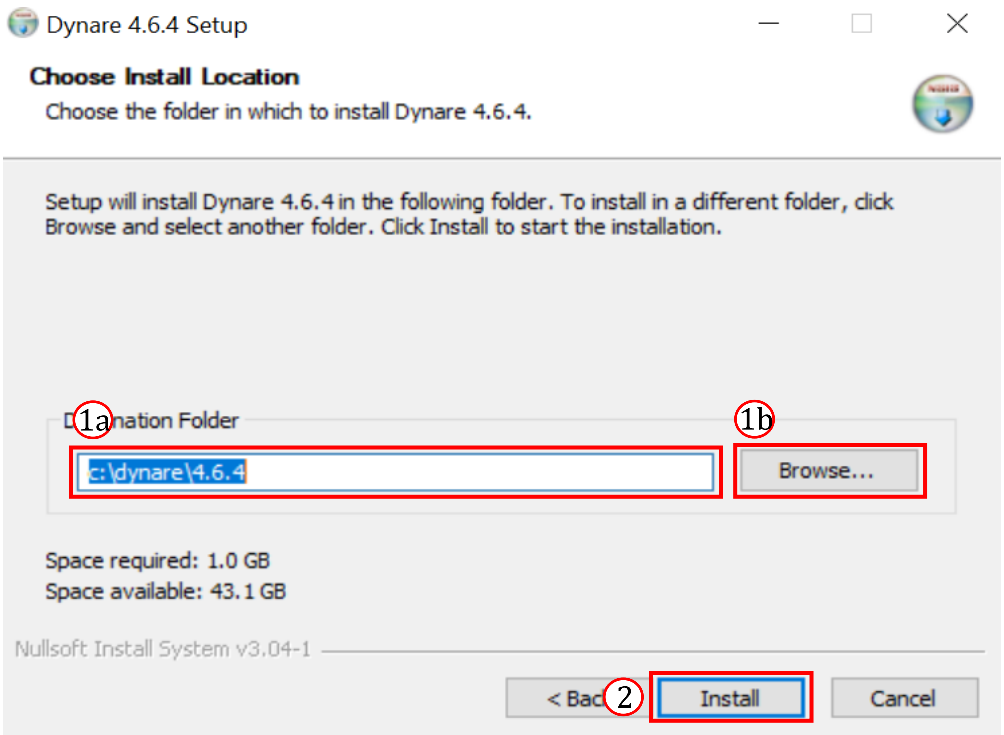
2. Select the *Dynare* package file from the downloads folder. You must allow Dynare to make changes on your device in order to install it.
3. In the installer window, click on 'Next'.



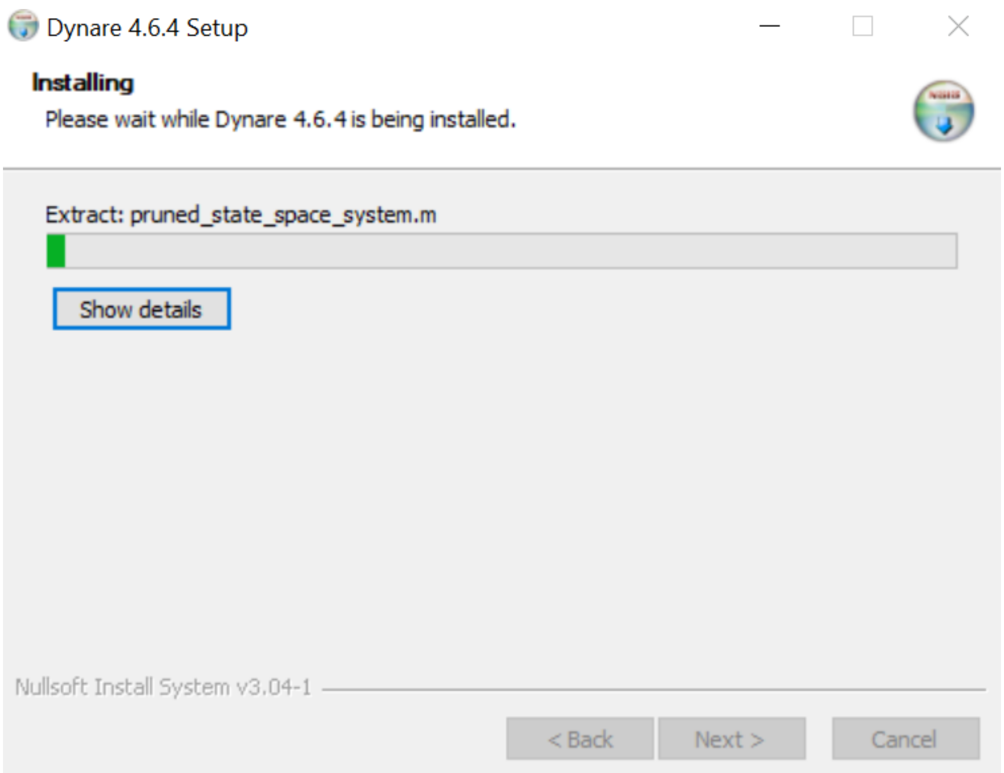
4. **(1)** Select the components you would like to install, depending on whether your MATLAB or Octave is 32 or 64-bit. If unsure, select both. Then, **(2)** click on next.



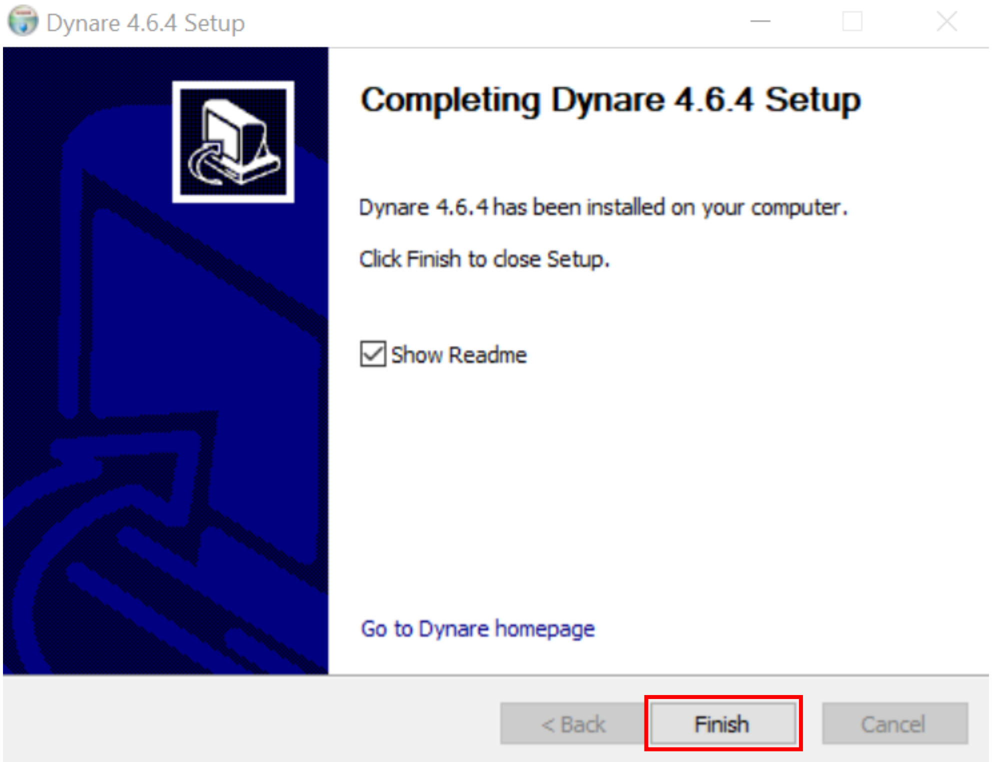
5. *Dynare* will be installed in your default disc unless otherwise indicated. You may specify a different location by **(1a)** typing or **(1b)** browsing a new directory. To begin with the installation, **(2)** click on 'install.'



6. You may check the installation process in the resulting window.



7. Once the installation has been completed, click on 'Finish.'

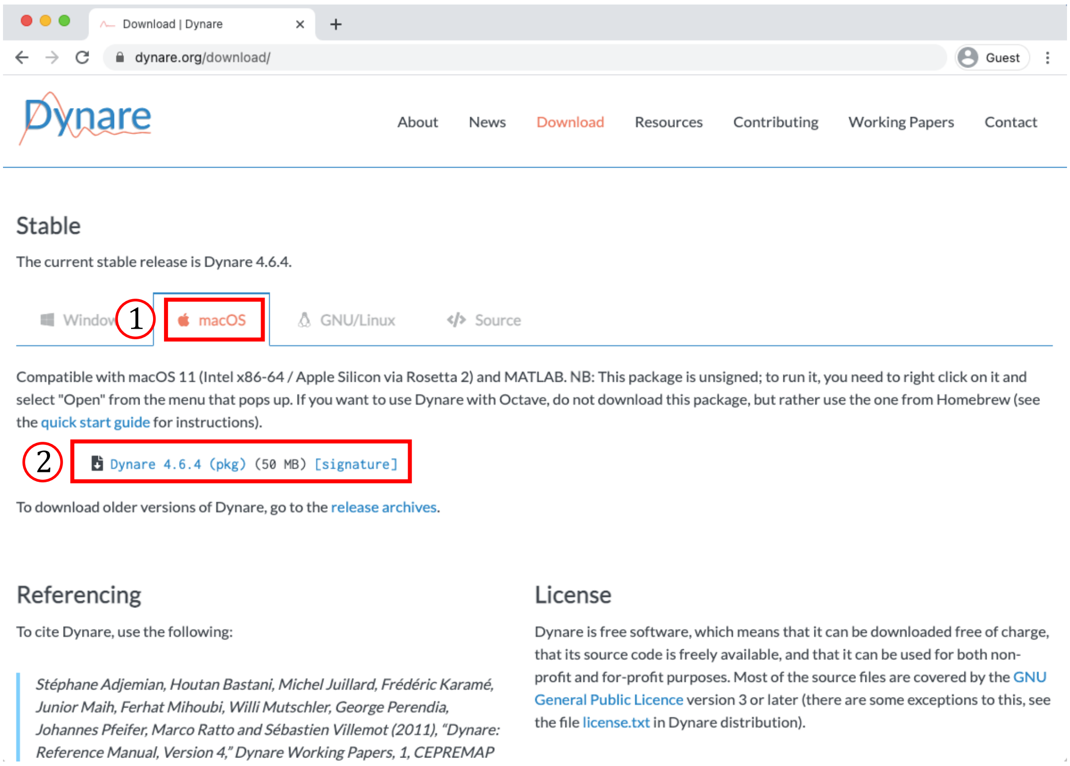


8. The installation of *Dynare* is completed.

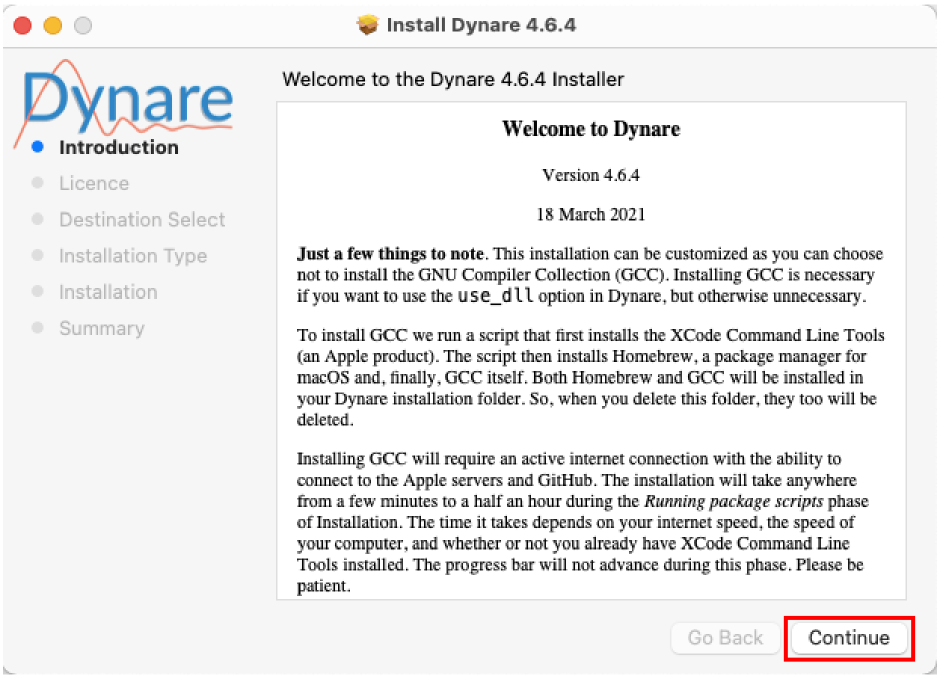
3.2 On macOS

A video tutorial of the installation of *Dynare* on macOS can be accessed here.

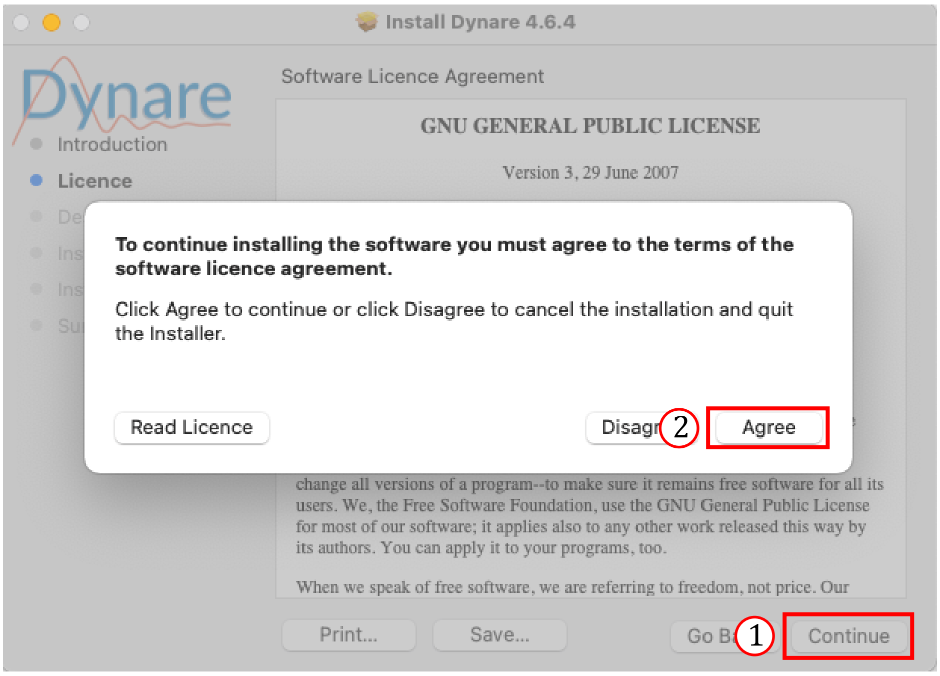
1. Go to https://www.dynare.org/download/ in your internet browser. (1) In the macOS tab, (2) click on the pkg file and wait until the downloading process is completed.
2. Select the dynare-4.x.y.pkg file from the downloads folder, where 4.x.y is the Dynare version you downloaded.



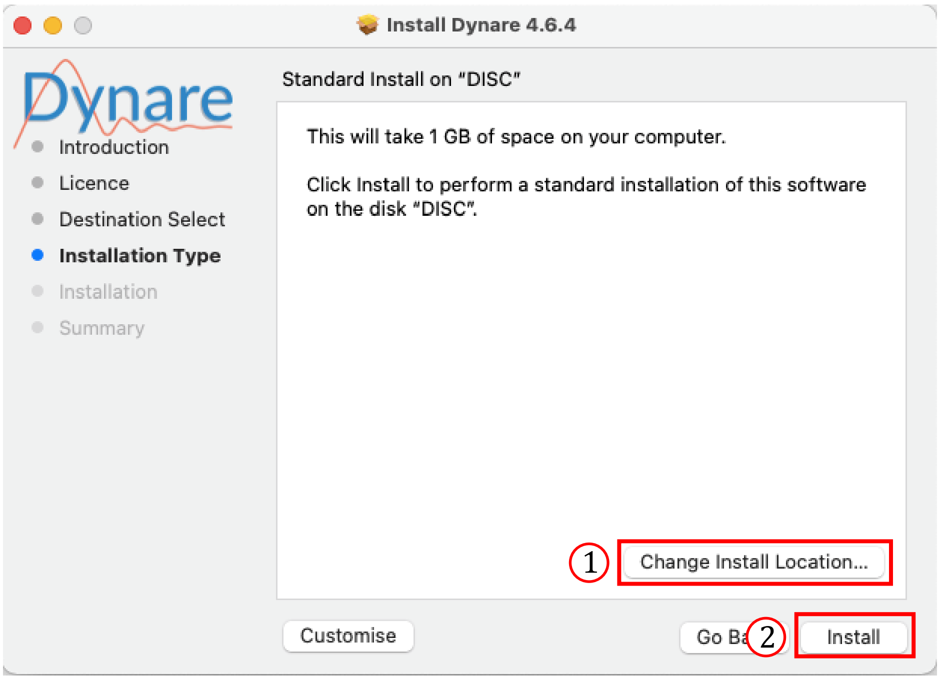
3. If your computer cannot open dynare-4.x.y.pkg because it is from an unidentified developer, continue with the next item. Otherwise, skip to item 5.
4. Go to System Preferences -> Security & Privacy. In the General tab, click on 'Open anyway.'
5. On the Dynare Installer window, click 'Continue.'



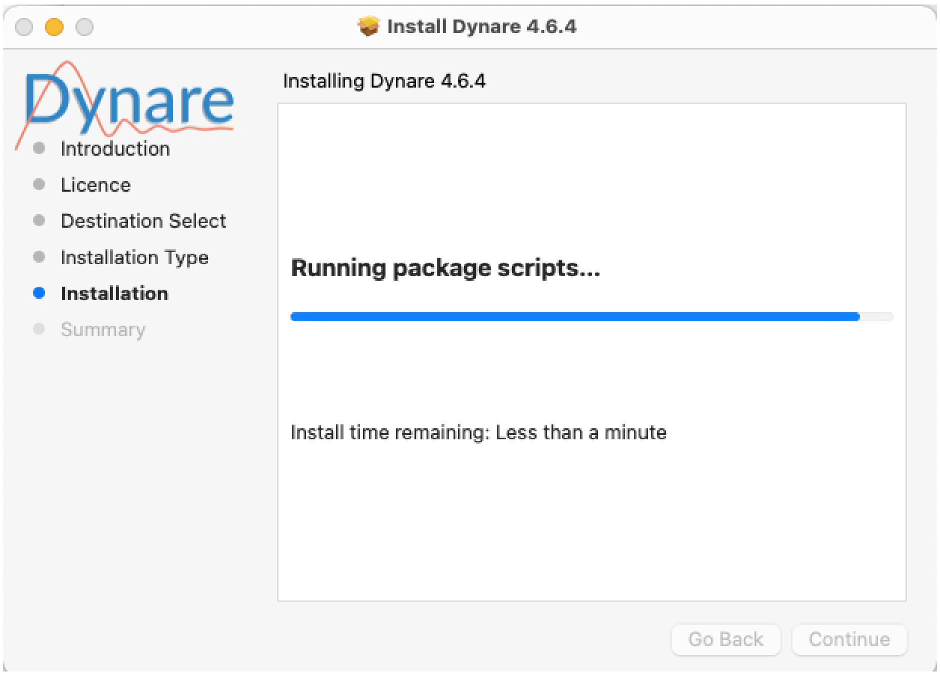
6. Review the GNU General Public License and **(1)** select '*Continue*.' A window will pop up. **(2)** To continue, you must click on '*Agree*.' You may click on '*Read Licence*' to access the license agreement before continuing with the installation.



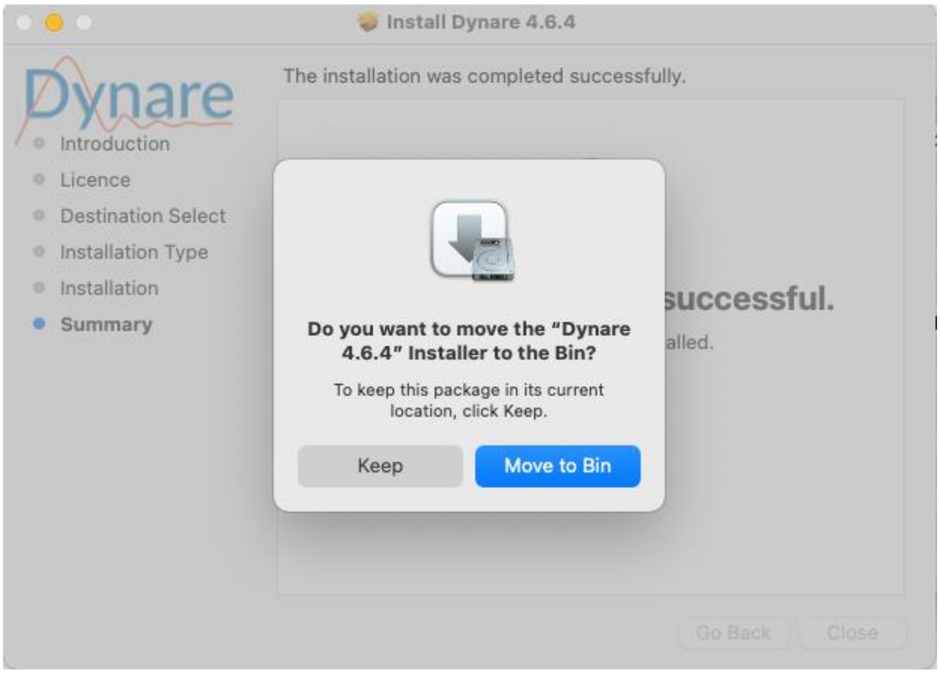
7. *Dynare* will be installed in your computer's default disk unless otherwise indicated. You can specify a different location by **(1)** clicking on '*Change Install Location*' and selecting your preferred disk. To proceed with the installation, **(2)** click on '*Install*.'



8. You will be required to introduce the user's password to proceed with the installation. The installation will start automatically when the password is introduced correctly. You may check the installation progress and the remaining time in the resulting window.



9. Once the installation has been completed successfully, click on '*Close*.' You may move the installation package to the bin to save space on your computer. To do so, select '*Move to Bin.*'



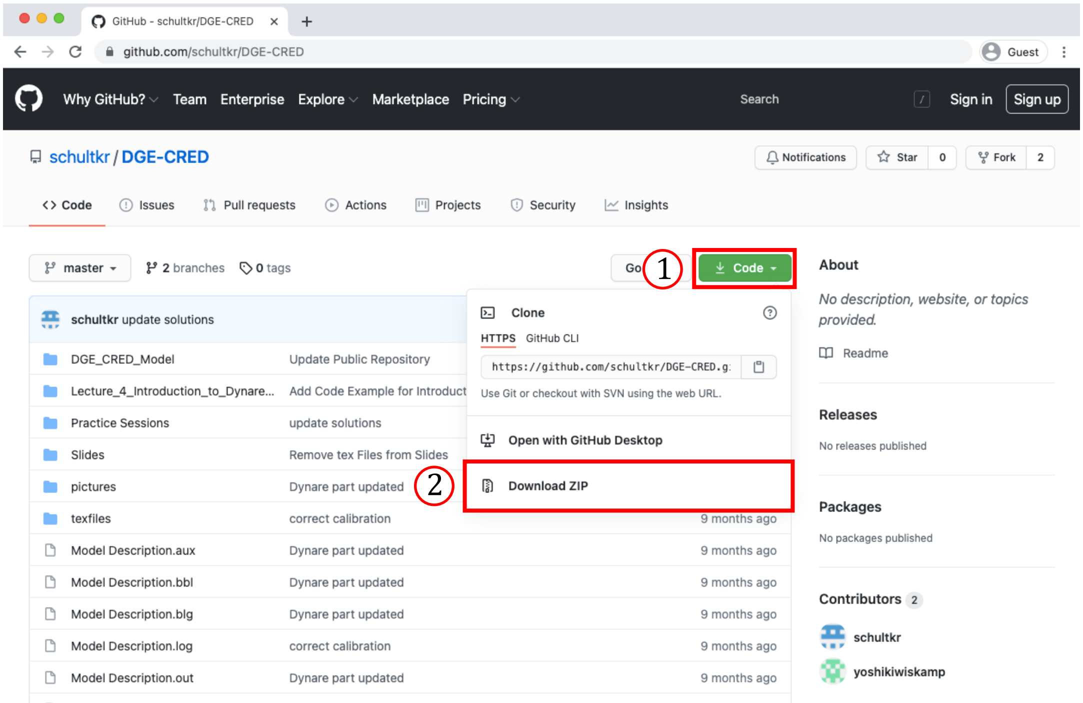
10. The installation of *Dynare* is completed.

4 DGE-CRED Model

4.1 Downloading the DGE-CRED Model

A video tutorial of the downloading of the model files can be accessed here.

1. Go to https://github.com/schultkr/DGE-CRED in your internet browser to access the DGE-CRED repository in GitHub. **(1)** Click on the 'Code' button and **(2)** select 'Download ZIP' to download the files in .zip format. You can skip this step if you already have the zip folder.



2. Open the DGE-CRED-master.zip file from the downloads folder to unzip it.

4.2 DGE-CRED model Files

The following subfolders can be accessed via *DGE-CRED-master/DGE_CRED_MODEL*.

4.2.1 ModFiles

RunSimulations.m defines the different scenarios and contains a call to ***DGE_CRED_Model.mod***, which is used to adjust the number of productive sectors and regions to be included in the simulation. This .mod file also includes a call to the following files stored in the *ModFiles* folder

- i ***DGE_CRED_Model_AuxiliaryVaribles.mod*** defines climate shocks as well as productivity shocks.
- ii ***DGE_CRED_Model_Declaration.mod*** declares all endogenous and exogenous variables and structural parameters of the model.
- iii ***DGE_CRED_Model_Equations.mod*** contains the equations of the model.
- iv ***DGE_CRED_Model_LatexOutput.mod*** produces *LaTeX* output for documentation of the declared variables and model equations.
- v ***DGE_CRED_Model_Parameters.mod*** assigns values to the structural parameters of the model.

4.2.2 Functions

The following programs are responsible for finding initial and terminal conditions. A more detailed description of this program is included in the *Appendix*

- i ***Calibration.m*** finds the initial conditions to reflect a specific year.
- ii ***FindK.m*** looks for a capital allocation across sectors and regions to fulfil the static equations of the model.
- iii ***SteadyState.m*** a Matlab script to find the initial steady state of the model and to calibrate it. It calls the function ***Calibration.m.***
- iv ***Simulation.m*** is a Matlab script to run the perfect foresight solver in small steps and differently for the Baseline and climate change scenarios. It uses the ***FindK.m*** function.

4.2.2.1 Auxiliary

The following programs are necessary to run the model. The scripts and functions are called during the simulation of the model.

- i ***AssignPredeterminedVariables.m*** defines predetermined variables of the model to find the new steady state. The function is called by DGE_CRED_Model_steadystate.mat.

ii ***ChangeModFile.m*** changes the DGE_CRED_Model.mod file during the simulation. It allows changing the number of sectors and regions and whether productivity is endogenous or exogenous.

iii ***ComputeAggregates.m*** computes aggregate variables from regional and sectoral levels for the steady-state computation. The function is called by DGE_CRED_Model_steadystate.mat.

iv ***DefineAuxiliaryExpressions.m*** defines integer arrays with the position of endogenous, exogenous variables and parameters in the dynare model and results structure. The arrays are used to update the respective dynare structures during the simulation.

v ***LoadExogenous.m*** reads in the exogenous variables for each scenario and updates the dynare structures.

vi ***LoadGrowthRates.m*** reads in the growth rates for the baseline scenario.

vii ***TaxIncome.m*** computes the tax income in the steady-state for the model. The function is called by DGE_CRED_Model_steadystate.mat.

4.2.2.2 Miscellaneous

The programs contained in this folder are used to define Excel input files for simulations, even though they are not necessary for the model simulation itself. Please refer to each file individually to understand its purpose.

4.3.1 *ExcelFiles*

This subfolder contains data and results for different simulations with K sectors and R regions. ***ModelSImulationandCalibrationKSectorsandRRegions.xlsx*** includes the following sheets

i *Content* includes links and descriptions for all the sheets in the file. Additionally, it maps the different regions and subsectors to their associated reference numbers.

ii *Data* contains calibration values for all the sectors in each region as well as for climate variables.

iii *Start* includes initial values for the model variables of all sectors in each region.

iv *Structural Parameters* defines the model parameters for all sectors in each region.

v *Baseline* contains the path for exogenous variables in the baseline scenario.

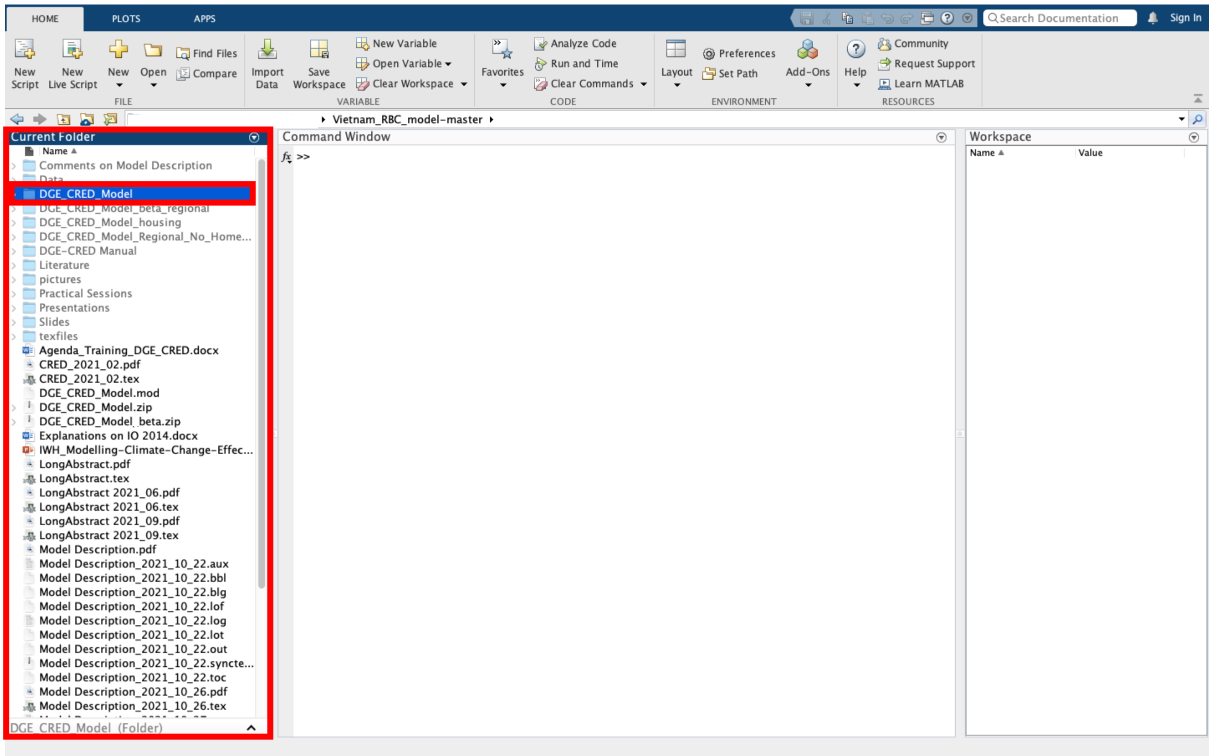
vi *SSP585* contains the path for exogenous variables in the high development pathway.

vii *SSP126* contains the path for exogenous variables in the green development pathway.

Furthermore, ***ResultsScenariosKSectorsandRRegions.xlsx*** stores the simulation results in the scenario sheets defined above.

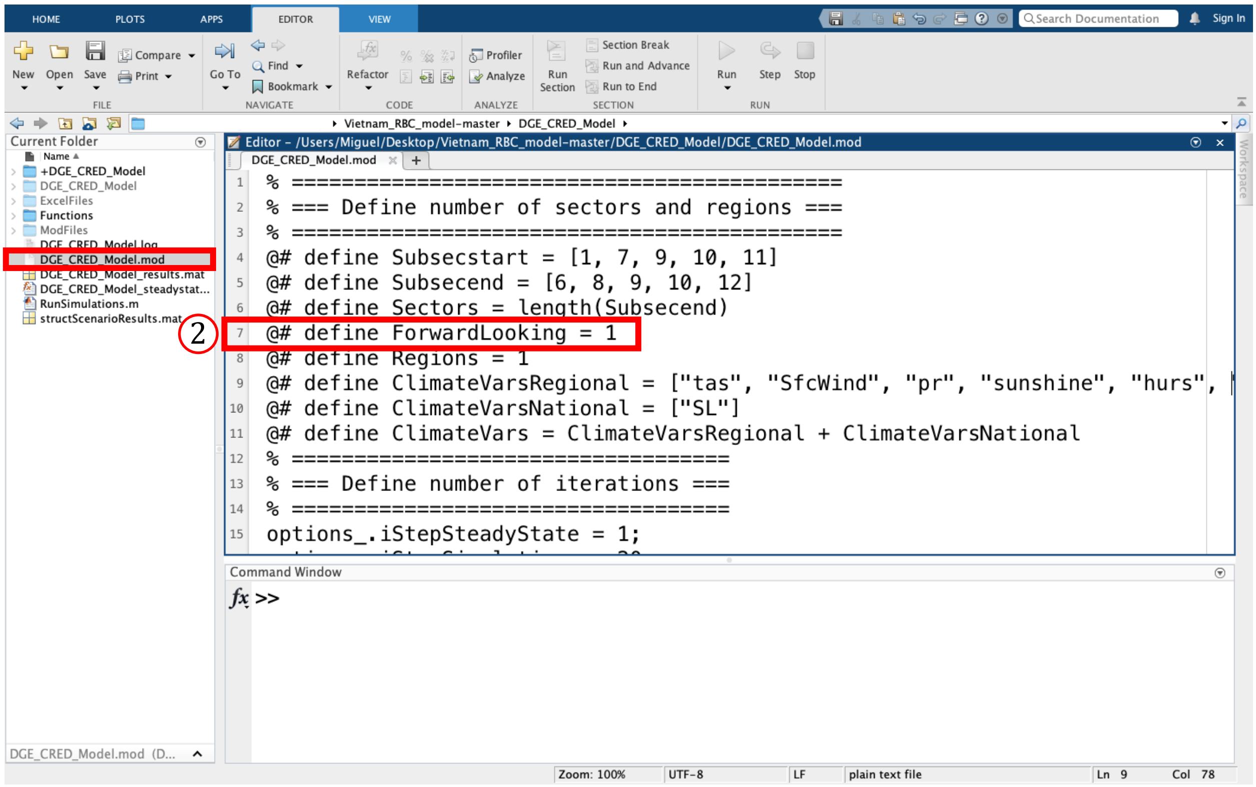
4.3 Model Simulations

1. Initiate *MATLAB* and open the ***DGE_CRED_Model*** folder, located in the ***DGE_CRED-master***. You can navigate the file browser to find this directory.



2. Find the ***DGE_CRED_Model.mod*** file and **(1)** click on it to open it. While this file does not need to be modified before executing the program, it allows the user to set different simulation parameters.

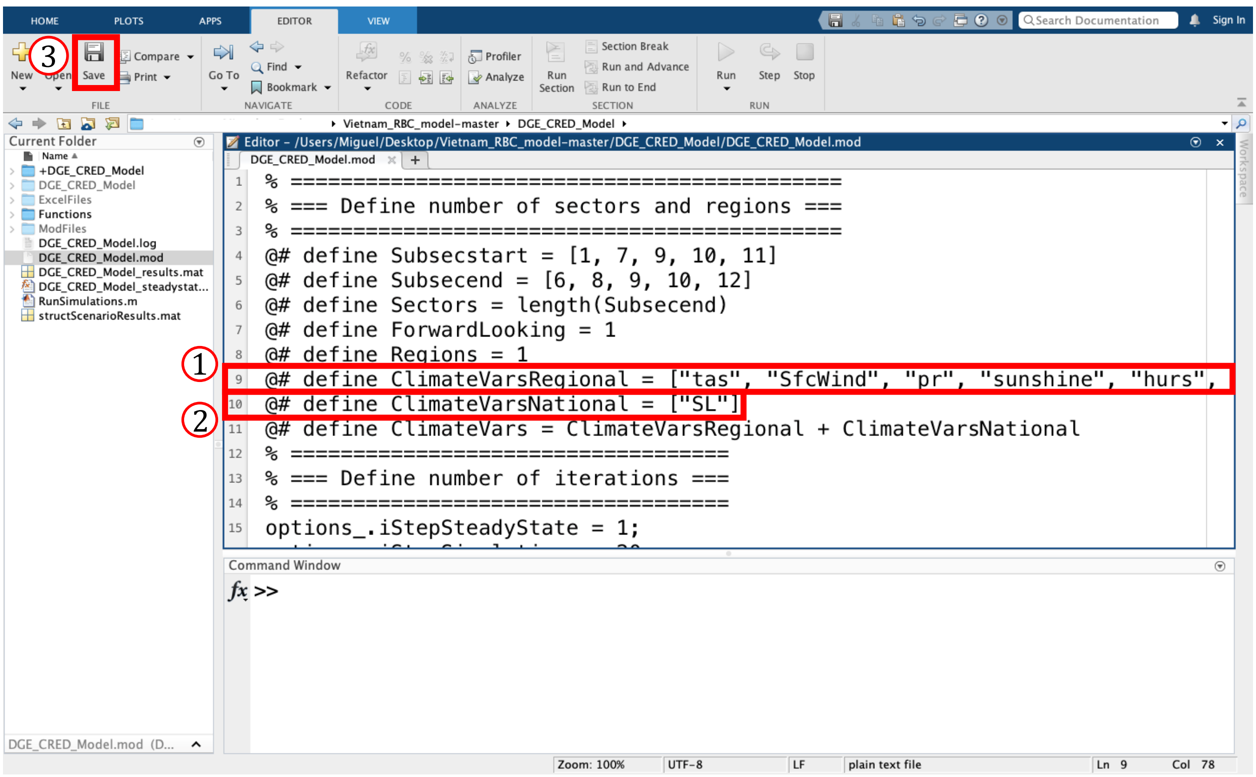
3. For instance, Next, the user can **(2)** determine whether agents in the model use rational expectations or form their expectations based on the previous period by setting line 7 of the *.mod* file to *'@# define ForwardLooking = 1'* or *'@# define ForwardLooking = 0'*, respectively.



4. An additional feature of this *.mod* file allows the user to define the climate variables that will be included in the model. **(1)** The climate variables that affect the economy at the regional level can be defined by setting line 9 of the *.mod* file to '@# define ClimateVarsRegional = [ ]' with the particular climate variables you would like to include separated by a comma inside the brackets. These are the commands for climate variables at the regional level.

   – "tas" accounts for changes in surface temperature.
   – "SfcWind" accounts for changes in surface wind speed.
   – "pr" accounts for changes in precipitation.
   – "sunshine" accounts for changes in hours of daily sunshine.
   – "hurs" accounts for changes in relative surface humidity.
   – "heatwave" accounts for changes in the number of yearly heatwaves.
   – "maxdrydays" accounts for changes in maximum consecutive dry days in a year.
   – "maxwetdays" accounts for changes in maximum consecutive wet days in a year.
   – "storms" accounts for changes in the number of yearly storms.
   – "floods" accounts for changes in the number of yearly floods.
   – "fires" accounts for changes in the yearly burned area.
   – "landslide" accounts for changes in the number of yearly landslides.

   (2) Line 10 of the .mod file can be modified to account for changes in the sea level nationally. To do so, set it to *'@# define ClimateVarsNational = ["SL"].'* Once you have adjusted the *.mod* file, **(3)** save the file by either clicking on the '*Save*' icon in the menu bar or pressing *Command + S*.



5. **(1)** Open the **RunSimulations.m** file. **(2)** Line 5 of the *.m* file should be set according to the *Dynare* version installed in your computer, as well as the operating system.

   The program is set to run on Windows by default. Therefore, if using *Windows*, you just need to make sure that line 5 includes the appropriate *Dynare* version. For instance, if you installed *Dynare* 4.6.3, line 5 should be set to '*addpath('C:\dynare\4.6.3\matlab').*'

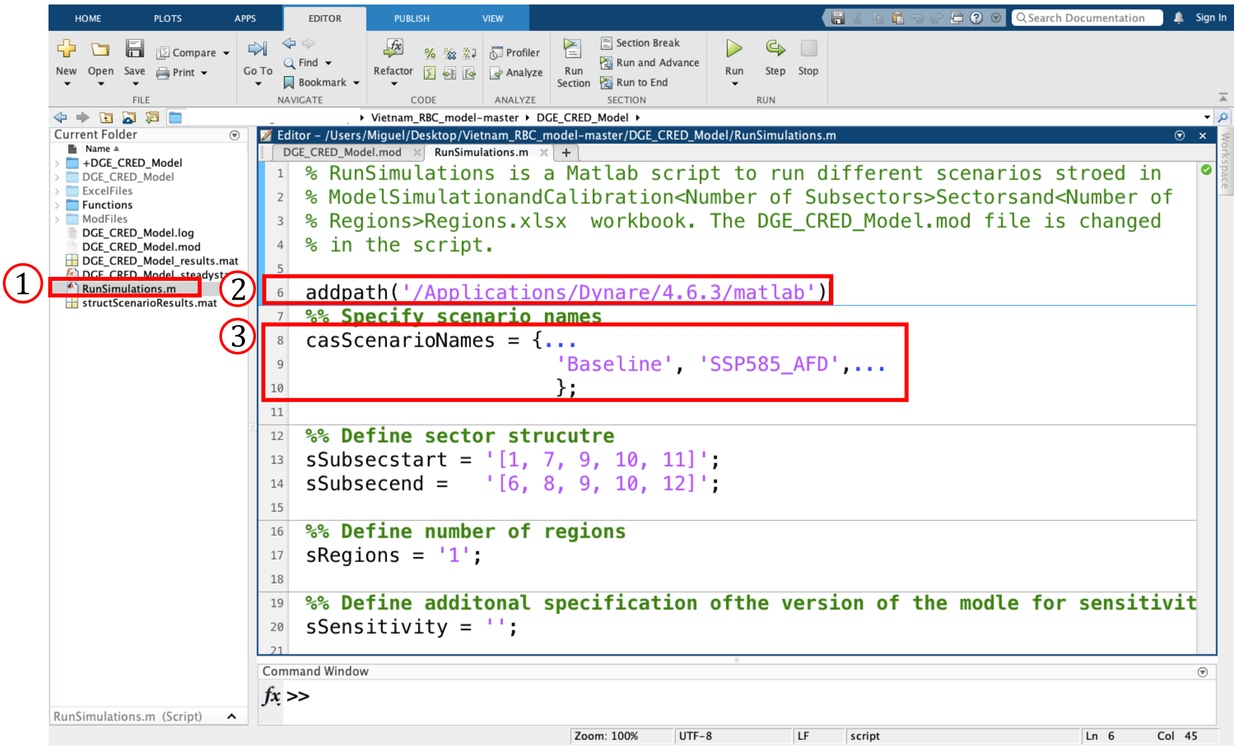
   To execute the program on macOS, you must set line 6 of the *.m* file to '*addpath('/Applications/Dynare/4.x.y/matlab')*', where 4.x.y is the *Dynare* version installed on your computer.

   Furthermore, the simulation of different scenarios is achieved by **(3)** modifying line 8 of the *.m* file to *'casScenarioNames = { }'* with any combination of the following commands separated by a comma inside the curly brackets.

   – "Baseline" only accounts for population growth following the medium GSO projection, which forecasts an increase of the Vietnamese population from 95 million people in 2016 to 108 million by 2050. After 2050, the population will stay constant.

   – "SSP126" represents the low end of the range of plausible future pathways, depicting the 'best case' future from the sustainability perspective. In this sense, global greenhouse gas emissions evolve such that low radiative forcing (2.6 W/m2) is achieved by 2100.

   – "SSP585" represents the high of plausible future pathways, with emissions high enough to produce an 8.5 W/m2 level of radiative forcing by 2100.

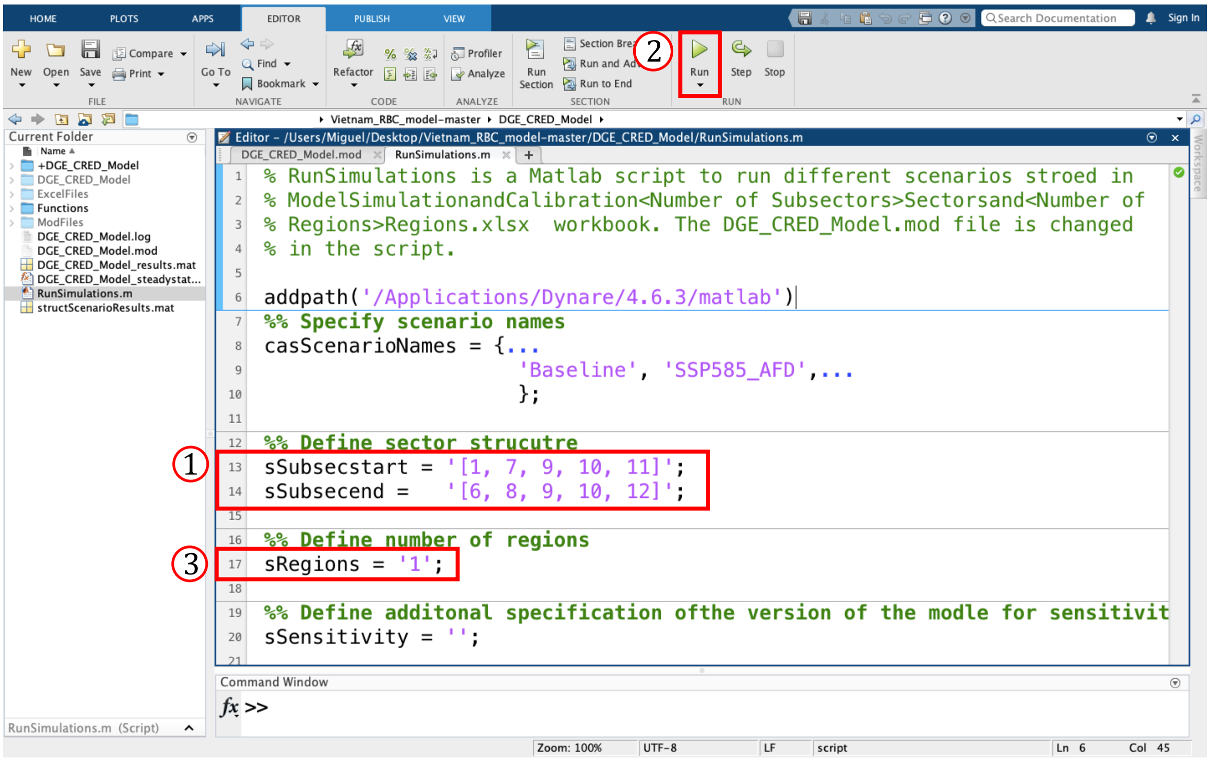


6. Furthermore, the ***RunSimulations.m*** allows for adjusting the number of sectors and subsectors by modifying lines 13 and 14. The model considers a maximum of five sectors; 1) Basics, 2) Construction and manufacturing, 3) Transport water, 4) Transport Land, and 5) Services and Health. The number of sectors is given by the length of the vectors in lines 13 and 14, which must be the same.

   These sectors include subsectors, which can be broken down as follows: 1) Rice, 2) Agriculture excluding rice, 3) Aquaculture, 4) Forestry, 5) Water, 6) Energy, 7) Manufacturing, 8) Construction, 9) Transport Water, 10) Transport Land, 11) Health, and 12) Services. If we want to simulate the model with 5 sectors, a possible option is to **(1)** set line 13 of the *.m* file to *'sSubsecstart = [1, 7, 9, 10, 11]'* and line 14 to *'sSubend = [6, 8, 9, 10, 12].'* This implies that sector 1 includes subsectors 1 through 6, sector 2 contains subsectors 7 and 8, sector 3 incorporates subsector 9, while sector 4 includes subsector 10, and sector 5 contains subsectors 11 and 12.

   Once you have made the necessary changes to the .m file, **(2)** execute the program by clicking on the '*Run*' icon in the menu bar. You may open the *Command Window* to see the program output as the simulations are performed.

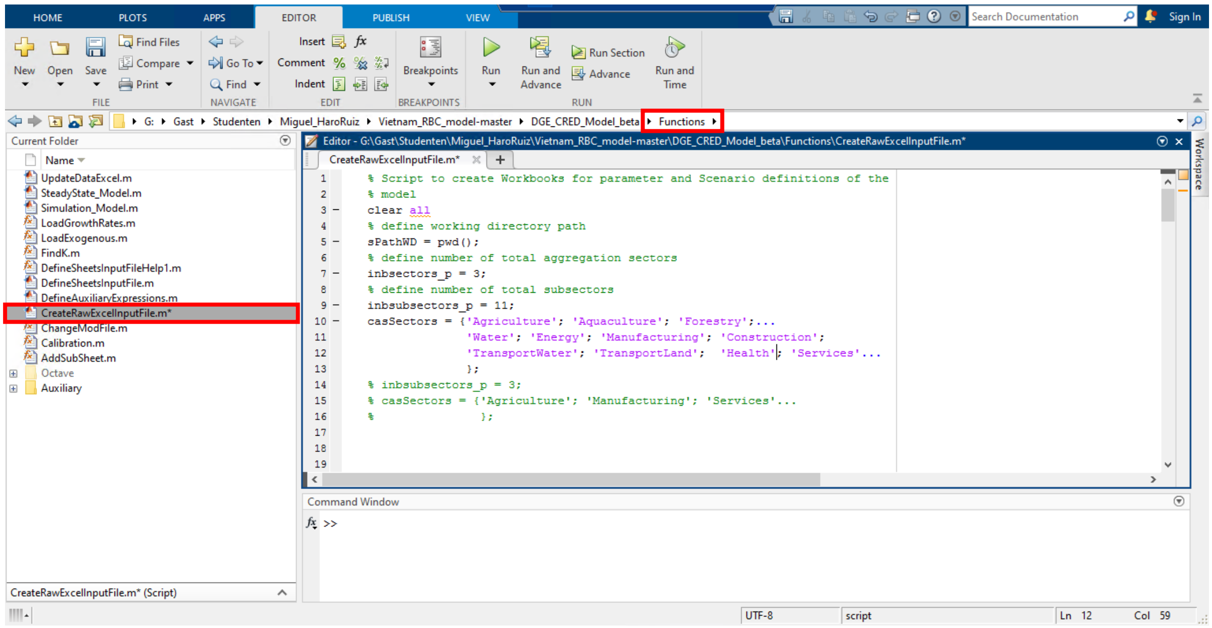
   Furthermore, **(3)** define the number of regions to be included in the simulation by adjusting the variable "sRegions" in line 17. For example, to simulate 1 region, line 17 of the .m file should be set to "sRegions = 1."



4.4 Create Customised Scenarios

A feature of the model allows the user to create customised excel sheets that can be used to calibrate the model. IMPORTANT: The *CreateRawExcelImput.m* defines certain variables more than once to provide examples on how to customise them. Once you have defined the variables to your need, make sure the remaining calls to the variables are deleted from the file or silenced by writing '%' in front of the lines containing them.

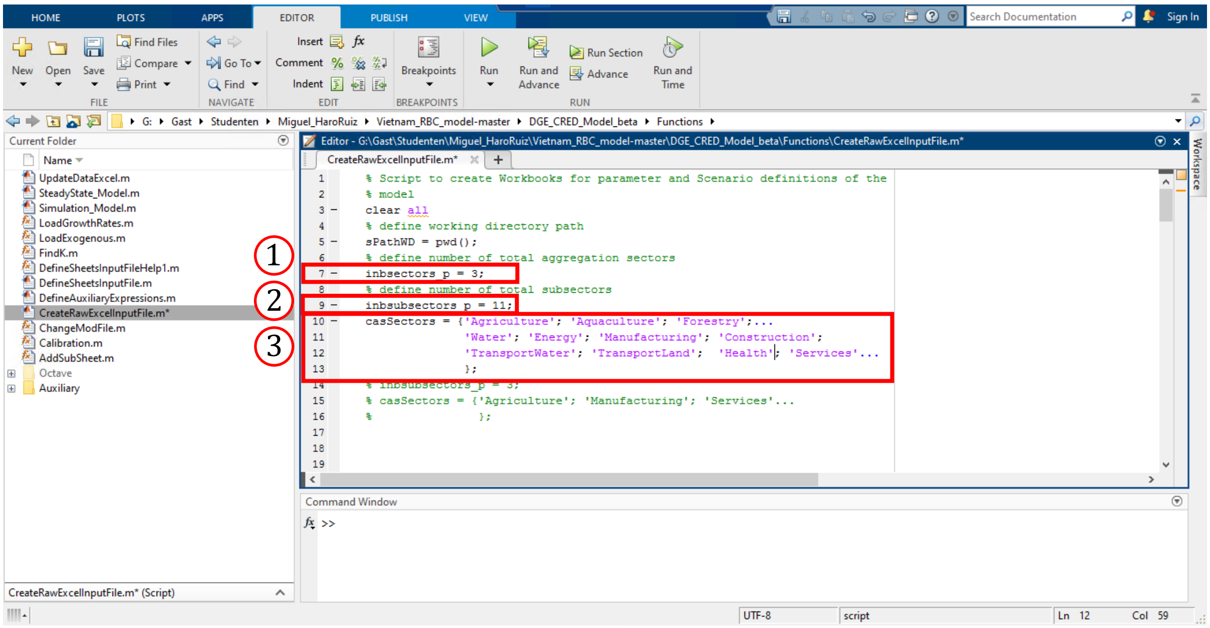
1. Initiate *MATLAB* and open the ***DGE-CRED-master*** file in the file browser. In the ***Functions*** folder, open the *CreateRawExcelInput.m* file.



2. The number of sectors can be **(1)** set by using the variable *'inbsectors_p'*. For instance, to create an excel sheet with three sectors, set line 7 of the *.m* file to *'inbsectors_p = 3;.'* Furthermore, the subsectors' characteristics are controlled by two variables.

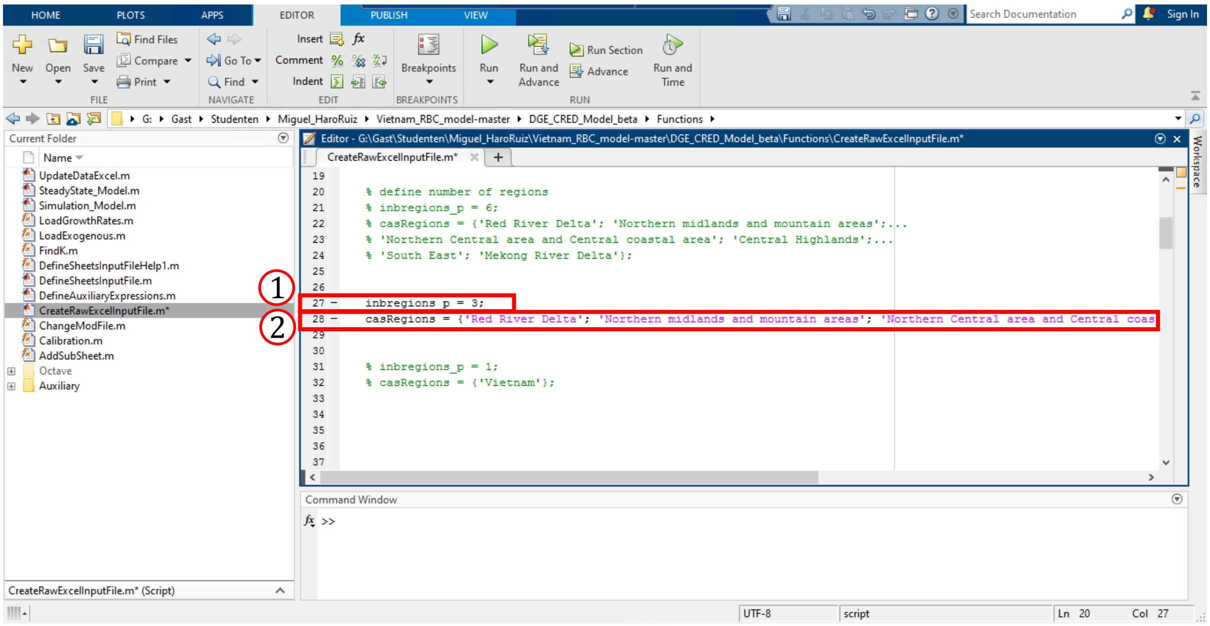
   (2) *'inbsubsectors_p'* determines the number of subsectors and must be at least equal to the number of sectors defined with *'inbsectors_p.'* Following the example with three sectors, to create an excel sheet with three subsectors (one in each sector), set line 9 of the *.m* file to *'inbsubsectors_p = 3.'*

   (3) *'casSectors = { }'* defines the specific subsectors by assigning names to each of them. Thus, the number of strings included in this variable must be equal to the number set for *'inbsubsectors_p.'* Continuing the example, to create an excel sheet with 3 sectors and 3 subsectors, include three subsector names in quotation marks, separated by commas inside the curly brackets.



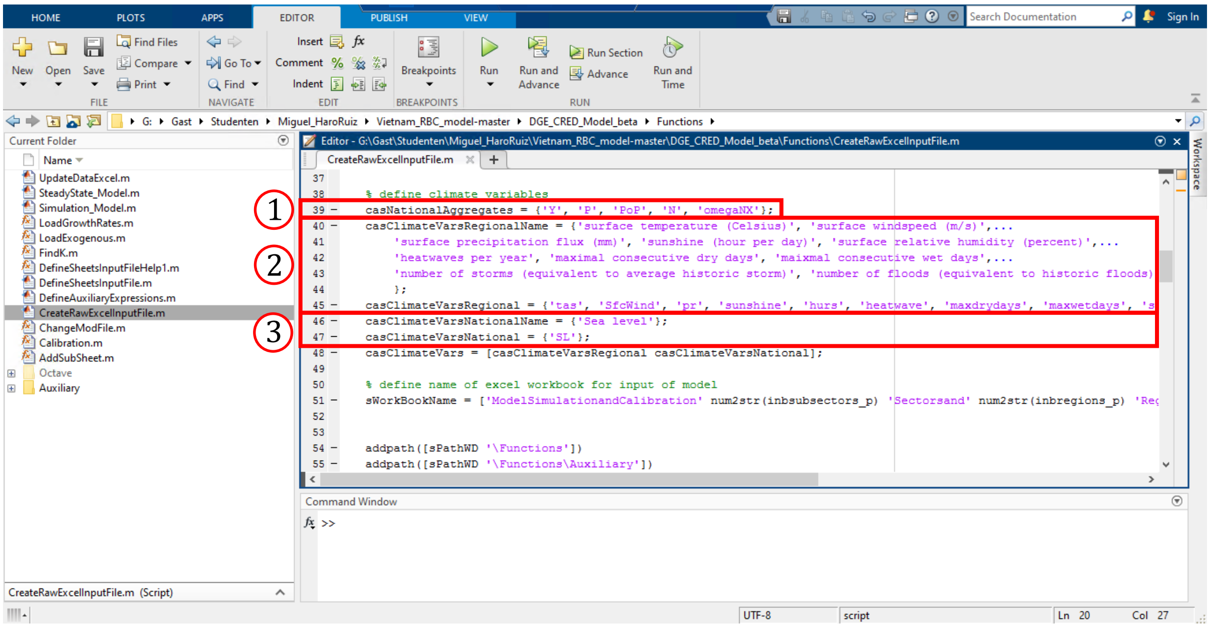
3. **(1)** The number of regions is defined using the variable' *inbregions_p.'* For instance, to create an excel sheet with 3 regions, set line 27 of the *.m* file to '*inbregions_p= 3;.'*

   Furthermore, **(2)** *'casRegions = { }'* allows the user to name each of the regions defined with '*inbregions_p.'* Following the example with three regions, include three region names in quotation marks, separated by commas inside the curly brackets.



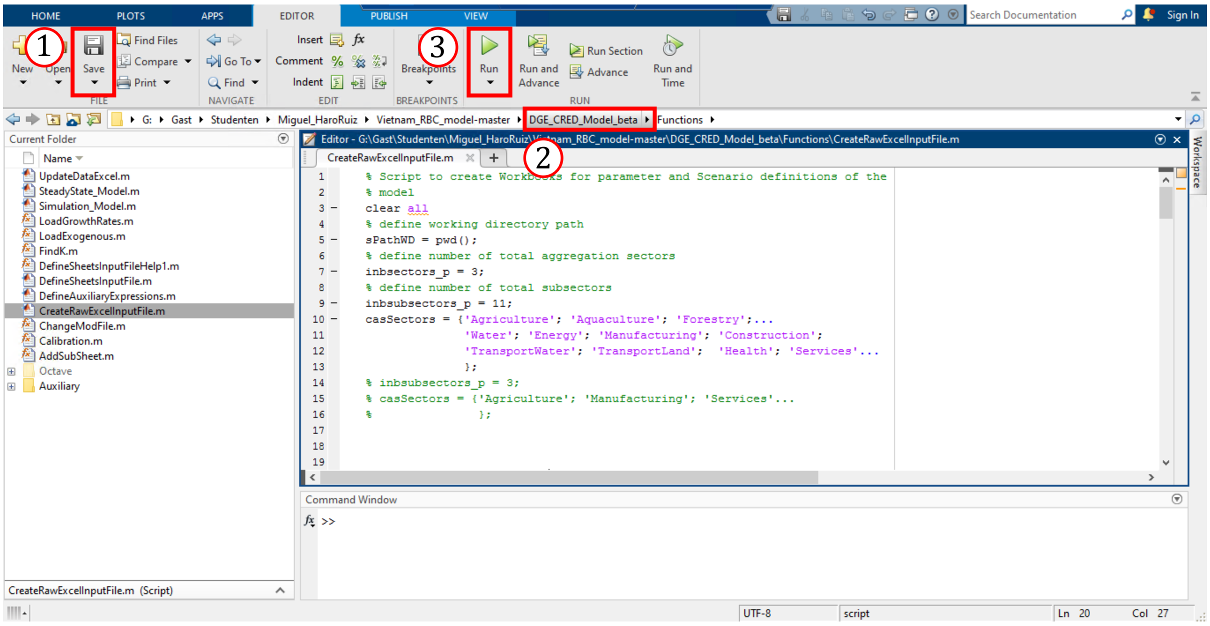
4. A final set of customisable elements the user can define are the national aggregates, as well as the regional and national climate variables. **(1)** To create an excel sheet with different national aggregates, use *'casNationalAggregates = { }* and include them in quotation marks, separated by commas inside of the curly brackets.

   (2) Regional climate variables are controlled by *'casClimateVarsRegionalName = { }'* which defines the name of each climate variable, and casClimateVarsRegional = { }, which specifies the variable symbol used in Matlab for each climate variable. Make sure both variables have the same number of elements. **(3)** National climate variables are defined analogously, using the *'casClimateVarsNationalName = { }'* and *'casClimateVarsRegional = { }.'*



5. Once you have made the necessary changes to the *CreateRawExcelInput.m* file and made sure that none of the variables is defined more than once, **(1)** save it by clicking on the 'Save' icon in the menu bar. Next, **(2)** go back to the 'DGE_CRED_model' folder in the file browser and **(3)** click on the 'Run' icon in the menu bar. A message window will appear; select 'add path' to execute the program.

   An excel sheet named ***ModelSImulationandCalibrationKSectorsandRRegions.xlsx***, where K and R are the numbers of defined subsectors and regions respectively, will be saved in the *ExcelFiles* folder.



5 Sectoral damages and adaptation measures

This section describes the calibration process of the model so that it reflects the way in which the Vietnamese economy responds to climate change-related events. Furthermore, the climate scenarios used in the model are explained. The following sections are dedicated to four sectors of the Vietnamese economy: agriculture, forestry, housing and transport. Each of the sections reports how climate variables affect the sector under consideration, as well as how adaptation measures can reduce climate-induced damages.

For a complete description of the calibration process, as well as the mechanics of climate scenarios with respect to each sector, please refer to Section 3—scenario Analysis of the Model Description.

5.1 Calibration

In order to examine how different climate change variables affect the production process in Vietnam, it is necessary to calibrate the model such that it reflects the current situation of the Vietnamese economy. We do so with data for the year 2014 since it constitutes the last year for which we have observed data.

In terms of production, the model uses a Cobb-Douglas production function. Therefore, we must calibrate the output elasticities of capital and labour such that they match observed parameters, i.e. the share of the wage bill on gross value added for a given elasticity of substitution. Furthermore, we specify how labour supply responds to wage changes, as well as the rate of capital stock depreciation, which is assumed to be equal across sectors and regions. Taxation in the Model is simplified, such that all tax revenues of the government come from a tax on consumption, which is set to 20%. This rate reflects the relation between total tax income and consumption in Vietnam in 2014.

The initial population used by the model reflects the population in Vietnam in the baseline year, as reported by the country's General Statistical Office, which is about 90.7 million people. The initial GDP is equal to 186 billion dollars as reported by the world bank. The model also requires the initial share of hours worked, which we obtain from the average weekly hours worked, as well as the share of employed individuals. Furthermore, the share of investment into housing with respect to GDP is defined.

5.2 Climate Scenarios

For a detailed description of how to simulate different scenarios, see Section 4.3.

A variety of climate scenarios can be implemented through the ***DGE_CRED_Model.mod*** file, using abbreviations for climate variables. New climate variables can be added to the mod-file depending on whether data is available at the regional or national level.

The simulation of different climate scenarios can be adjusted using the RunSimulations.m file. In addition to the baseline scenario, which only considers population growth, new climate pathways can be included using the commands "SSP126" and "SSP585". Figure 1 illustrates the main steps in developing the SSPs, including the narratives, socioeconomic scenario drivers (basic SSP elements), and SSP baseline and mitigation scenarios.

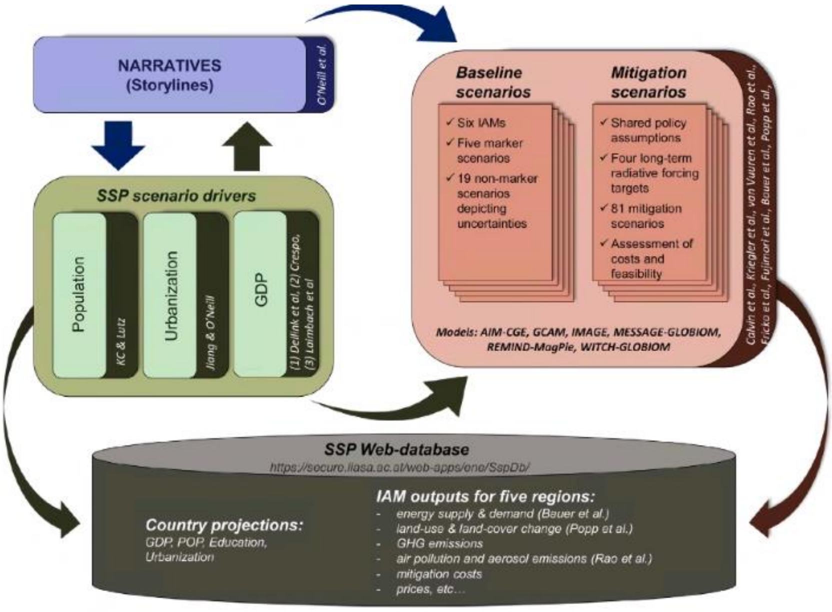


Figure 1. Schematic illustration of steps to develop SSPs scenarios.

The data for the different climate pathways is obtained from the Excel sheets named "SSP126" and "SSP585", included in the ***ModelSImulationandCalibrationKSectorsandRRegions.xlsx*** file.

5.3 Agriculture Sector

Damages to the agricultural sector are implemented in the model as a result of changes in temperature as well as changes in sea level rise. Firstly, a reduction of crop yields due to an increase in temperature will reduce the maximum number of crops produced in a given period using the same amount of production factors. Secondly, land loss due to sea-level rise will force farmers to produce with less land. This implies that the same number of workers and the same capital stock can only be used in smaller areas. The impact of sea-level rise and temperature increase affects total factor productivity through D_1 and D_2.

For example, we specify the damages D_1 in the rice sector based on the damage that resulted from an increase in atmospheric temperature and sea level. With an increase in temperature by 1°C, the damage will increase by 3.1%. For sea level, a step-function relationship is expected according to the land loss bins.

$$
\begin{aligned}
D_{1,r,t} &= \eta^D_{1,r,t} = 0.031\,\eta^{tas}_{r,t} + 1(\eta^{SL}_t \in b)\,ll_{b,1,r},\\
D_{2,r,t} &= \eta^D_{2,r,t} = \underbrace{0.006}_{=1-(1-0.031)^{0.017}\,(1-0.08)^{0.0657}}\,\eta^{tas}_{r,t} + 1(\eta^{SL}_t \in b)\,ll_{b,2,r}.
\end{aligned}
$$

soybean maize share in gross value added

For D_2, i.e. the damage in agriculture excluding rice, we can directly use the values from the literature (see Table 1) and multiply them by the share of value-added to the agricultural sector as given in the input-output table (See Model Description for details). For instance, the share of soybean is 1.7% and maise 6.6%.

Crop	Loss ($\%/^{\circ}C$)	Region
wheat	-2.6	China
rice	-3.0	Vietnam
maize	-8.0	China
soybean	-3.1	China

Table 1. Crop yield loss
Source: Zhao et al. (2017).

Adaptation in the agricultural sector occurs in different ways, for instance, by switching production from the more affected sectors to the less affected sectors or regions in Vietnam. This is possible by assuming that agents switch from highly vulnerable to less vulnerable sectors through optimal behaviour. Additionally, the loss in total factor productivity can be compensated by investing more into the capital stock.

5.4 Forestry sector

Damages to the forestry sector in Vietnam are included in the model through the hazard of forest fires. The frequency of forest fires is computed using the so-called Keetch-Byram Drought Index (KBDI), which is used to predict the risk of forest fires depending on the daily maximum temperature, as well as daily and annual average rainfall.

Using historical data on the burned area in Vietnam, we determine that a KBDI value larger than 150 indicates good conditions for a forest fire to develop. In this sense, a fire occurs in a given region with a probability of 42% if the cumulative KBDI for a week exceeds 150, and only with a probability of 15% if the KBDI is below the threshold. The damage due to forest fire in the forestry sector for each region is computed by multiplying the number of forest fires in the region for the respective year with the fraction of land burned from the historical database.

The adaptation measure in the forestry sector consists of changing one hectare of pure eucalyptus forests to mixed forests with eucalyptus and pine. After the adaptation measures have been implemented, the burned area per fire in the region is reduced. However, the adaptation measures only reduce the burned area after completion of the adaptation measure, which is assumed to be 10 years. **Error! Reference source not found.** shows the damage caused by forest fires with and without adaptation measures.

5.5 Housing sector

Damages in the housing sector are implemented in the model as a result of storms as well as changes in the sea level. On the one hand, the frequency of storms is different across regions for which the model simulates the storm occurrences through an exogenous variable. Furthermore, it is assumed that storms will destroy 10% of the houses in a region, for which the regional damage depends on the number of houses in each region. In terms of sea level, it is assumed to only affect housing in coastal regions and that 10% of the houses affected by floods in a region will be destroyed. The share of construction land loss for a specific range of sea-level rise is used to estimate the costs to the housing stock.

The model includes two adaptation measures for housing. Firstly, it is possible to build houses with reinforced walls and bricks to reduce the vulnerability of houses to storms. Additionally, raising a house on stilts will reduce the impact of an increase in sea level. The damages caused by sea-level rise and storms are subtracted from the benefits associated with the necessary cumulative costs for the adaptation measures.

5.6 Transport sector

Damages to the road stock in Vietnam are incorporated into the model through damages to the capital stock of the land sub-sector. There are three ways in which roads can be damaged. First, a rise in sea level can flood some of the roads. Second, consecutive days with temperatures above 30°C can destroy the asphalt. Finally, landslides constitute another factor that can destroy transport-related infrastructure. Roads are classified by type (highways, national and provincial roads), as well as their vulnerability to climate events (very low, low, medium, high, very high). For the model simulation, damages and costs across road types and vulnerability levels are aggregated such that subjective probabilities are assigned to different vulnerability classes in order to compute expected damages on the road stock.

Three adaptation measures are considered. First, elevating affected roadbeds by a certain amount can tackle rises in sea level. If this is done before the sea level rises, the potential damage will be reduced to zero. Second, replacing conventional asphalt concrete with polymer asphalt concrete can protect the road from high temperatures. It is assumed that as soon as the implementation of this measure has begun, the damage posed by heatwaves is zero. Third, measures against road erosion are used to mitigate the impact of landslides on the road stock and take only a year to be implemented.

6 Practical session

6.1 Practical Session 1

Task 1: Create an Excel input file with 12 subsectors and 1 region.

- Use the ***CreateRawExcelInputFile.m*** file, located in the Functions/Miscellaneous folder. You need to run it in the main directory of the model *'.../DGE_CRED_Model/'* and select "add path", as opposed to "change path."
- *Subsector names:* Rice, Agriculture excluding rice, Aquaculture, Forestry, Water, Energy, Manufacturing, Construction, TransportWater, TransportLand, Health, Services.
- *Region name:* Vietnam.
- *regional climate variables names:* surface temperature (Celsius), surface windspeed (m/s), surface precipitation flux (mm), sunshine (hour per day), surface relative humidity (percent), heatwaves per year, maximal consecutive dry days, maximal consecutive wet days, number of storms (equivalent to average historic storm), number of floods (equivalent to historic floods), number of forest fires, landslides.
- *regional climate variables symbols*: tas, SfcWind, pr, sunshine, hurs, heatwave, maxdrydays, maxwetdays, storms, floods, fires, landslides.
- *national climate variables names:* sea level (cm).
- *national climate variables symbols:* SL.

The solution to task 1:

1. Initiate *MATLAB* and open the ***CreateRawExcelImputFile.m***, located under *Functions/Miscellaneous*. Make sure that the optimisation toolbox is installed in MATLAB.
2. We will include 12 subsectors, for which the variable "*casSectors*" must include each of the 12 subsectors' names within curly brackets ({...}). Be sure to type each subsector name within quotation marks ('...') and separate by commas, as shown below.

```
%% Define subsectors
casSubSectors = {'Rice'; 'Agriculture excluding rice'; 'Aquaculture'; 'Forestry';...
                'Water'; 'Energy'; 'Manufacturing'; 'Construction';
                'TransportWater'; 'TransportLand';  'Health'; 'Services'...
                };
```

3. The task prompts us to work with a single region, which we will call Vietnam. As a result, the variable "*casRegions*" must be set equal to "Vietnam", in quotation marks and within curly brackets, as follows.

```
%% Define regions
casRegions = {'Vietnam'; 'RoW'};
```

4. Next, define the regional climate variable names under the variable "*casClimateVarsRegionalName*", as well as the climate variable symbols under the variable "*casClimateVarsRegiona.l*" Make sure to type every name and symbol within quotation marks, separated by commas and in between curly brackets. Similarly, define national climate variable names and symbols within "*CasClimateVarsNationalName*" and "*CasClimateVarsNational*", respectively. Your input should look like the one below.

```
%% Define regional climate variables
casClimateVarsRegionalName = {'surface temperature (Celsius)', 'surface windspeed (m/s)',...
    'surface precipitation flux (mm)', 'sunshine (hour per day)', 'surface relative humidity (percent)',...
    'heatwaves per year', 'maximal consecutive dry days', 'maximal consecutive wet days',...
    'number of storms (equivalent to average historic storm)', ...
    'number of floods (equivalent to historic floods)', 'number of forest fire', ' land slides'...
    };
casClimateVarsRegional = {'tas', 'SfcWind', 'pr', 'sunshine', 'hurs', 'heatwave', ...
    'maxdrydays', 'maxwetdays', 'storms', 'floods', 'fire', 'landslide'};

%% Define national climate variables
casClimateVarsNationalName = {'Sea level'};
casClimateVarsNational = {'SL'};
casClimateVars = [casClimateVarsRegional casClimateVarsNational];
```

IMPORTANT: Before executing the file, make sure that none of the previously defined variables is included twice. In order to avoid defining variables twice, you can write "%" in front of the lines with extra definitions or delete these lines altogether.

5. Now that all the relevant variables are defined navigate back in the file browser such that the current directory is the main DBE_CRED_Model file. Run the ***CeateRawExcelImputFile.m***. A window will pop up informing the user that the file to be executed is not included in the current directory; select "add path" and wait for the program to finish executing.

6. To see the output of the program, go to the "ExcelFiles" folder and select the newly created ***ModelSimulationandCalibration12Sectorsand1Regions.xlsx*** file. Notice that while the raw outline of the Excel file that feeds the model has been created, the data needs to be imputed by the user.

   The "Data" sheet looks as follows.

	A	B	C	D	E	F	G	H	I	J
1	Sector	Region	Initial Value Added Shares (phiY0)	Initial Employment Shares (phiN0)	Labour Cost Shares (phiW)		Sector	export share (phiX)	import share (phiM)	intermediate products (phi(
2	Rice	Vietnam	enter value here	enter value here	enter value here		Rice	enter value here	enter value here	enter value here
3	Agriculture excluding rice	Vietnam	enter value here	enter value here	enter value here		Agriculture excluding rice	enter value here	enter value here	enter value here
4	Aquaculture	Vietnam	enter value here	enter value here	enter value here		Aquaculture	enter value here	enter value here	enter value here
5	Forestry	Vietnam	enter value here	enter value here	enter value here		Forestry	enter value here	enter value here	enter value here
6	Water	Vietnam	enter value here	enter value here	enter value here		Water	enter value here	enter value here	enter value here
7	Energy	Vietnam	enter value here	enter value here	enter value here		Energy	enter value here	enter value here	enter value here
8	Manufacturing	Vietnam	enter value here	enter value here	enter value here		Manufacturing	enter value here	enter value here	enter value here
9	Construction	Vietnam	enter value here	enter value here	enter value here		Construction	enter value here	enter value here	enter value here
10	Transport Water	Vietnam	enter value here	enter value here	enter value here		Transport Water	enter value here	enter value here	enter value here
11	Transport Land	Vietnam	enter value here	enter value here	enter value here		Transport Land	enter value here	enter value here	enter value here
12	Health	Vietnam	enter value here	enter value here	enter value here		Health	enter value here	enter value here	enter value here
13	Services	Vietnam	enter value here	enter value here	enter value here		Services	enter value here	enter value here	enter value here
14										

The “Start” sheet looks as follows.

	W	X	Y	Z	AA	AB	AC	AD	AE
1	initial number of forest fire (fire)	initial land slide (landslide)		Name	Value		Name	Value	
2	enter value here	enter value here		initial Sea level	enter value here		initial population	enter value here	
3							initial value added	enter value here	
4							import share	enter value here	
5							housing to population ratio	enter value here	
6							investmetns in residential building relative to GDP	enter value here	
7									

The “Baseline” sheet looks as follows.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	Time	exo_PoP	gY_1_1	gY_2_1	gY_3_1	gY_4_1	gY_5_1	gY_6_1	gY_7_1	gY_8_1	gY_9_1	gY_10_1	gY_11_1	gY_12_1	gN_1_1	gN_2_1	gN_3_1	gN_4_1	gN_5_1	gN_6_1
2	2	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	3	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	4	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	5	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	6	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	7	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	8	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	9	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	10	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
11	11	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
12	12	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
13	13	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
14	14	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
15	15	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
16	16	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
17	17	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
18	18	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
19	19	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
20	20	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
21	21	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
22	22	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
23	23	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
24	24	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
25	25	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
26	26	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
27	27	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
28	28	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
29	29	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Content | Data | Start | Structural Parameters | **Baseline** | Scenario

NOTE: In the “Start” and “Structural Parameters” sheets, the shares should add up to 1. In order to test this, type “1-SUM(share sector 1; share sector 2;…; share sector 12)” in an empty cell, as shown in the picture below. If the result is not equal to 0, a message will occur when you simulate the model in MATLAB.

55	Parameter values for import	11Z x 1S	
56	phiM_1_p	0.083333	import shares in sector 1
57	phiM_2_p	0.083333	import shares in sector 2
58	phiM_3_p	0.083333	import shares in sector 3
59	phiM_4_p	0.083333	import shares in sector 4
60	phiM_5_p	0.083333	import shares in sector 5
61	phiM_6_p	0.083333	import shares in sector 6
62	phiM_7_p	0.083333	import shares in sector 7
63	phiM_8_p	0.083333	import shares in sector 8
64	phiM_9_p	0.083333	import shares in sector 9
65	phiM_10_p	0.083333	import shares in sector 10
66	phiM_11_p	0.083333	import shares in sector 11
67	=1-SUMME(B56:B66		import shares in sector 12

Task 2: Run the ***RunSimulation.m*** file.

- Test whether the created input file is a valid input to conduct simulations.
  - Run the model simulation in the baseline scenario.
  - Execute the program with 12 subsectors and 1 region.

The solution to task 2:

1. Initiate *MATLAB* and open the ***RunSimulations.m***, located in the main *DGE_CRED_Model* file. Since we want to simulate the baseline scenario, the variable “*casScenatioNames*” should be set equal to “Baseline”, in quotation marks and within curly brackets, as follows.

```
%% Specify scenario names
casScenarioNames = {'Baseline'};

```

2. Next, define the distribution of subsectors using the variables “sSubsecstart” and “sSubsecend.” For a more detailed description of how to do this, refer to Section 4.3 of the manual. Furthermore, the number of regions should be set to 1 through the variable “sRegions.” Your input should roughly look like the one below.

```
%% Define sector strucutre
sSubsecstart = '[1, 7, 9, 10, 11]';
sSubsecend =   '[6, 8, 9, 10, 12]';

%% Define number of regions
sRegions = '1';

```

3. Finally, click on the “Run” icon. Throughout a successful execution of the simulation, your output should be the following.

```
Starting preprocessing of the model file ...
Found 82 equation(s).
Evaluating expressions...done
Computing static model derivatives (order 1).
Computing dynamic model derivatives (order 2).
Processing outputs ...
done
Preprocessing completed.
============================================
=== Step 1 of 20 for Baseline ===
.....
Iter: 1,          err. = 7.54952e-14,      time = 0.117

Total time of simulation: 0.119.
```

```
Perfect foresight solution found.

Total computing time : 0h00m10s
```

Task 3: Calibrate the Model.

- Fill the Excel file ***ModelSimulationandCalibration12Sectorsand1Regions.xlsx*** with the following data on
  - Subsectoral exports, imports, and intermediate products

Sector	Export Share	Import Share	Intermediate Products
Rice	0,0121	0,0003	0,5592
Agriculture excluding rice	0,1410	0,0407	0,6669
Aquaculture	0,7670	0,0002	0,7111
Forestry	0,3750	0,0446	0,5527
Water	0,0010	0,0001	0,6108
Energy	0,0082	0,0202	0,5776
Manufacturing	0,3368	0,8331	0,8198
Construction	0,0010	0,0001	0,7585
Transport Water	0,1872	0,0001	0,7508
Transport Land	0,1081	0,0049	0,5507
Health	0,0116	0,0030	0,5953
Services	0,1226	0,0526	0,5637

  - Subsectoral value added, employment and labour cost

Sector	VA Shares	Employment Shares	LC Shares
Rice	0,031810	0,100255	0,494986
Agriculture excluding rice	0,072820	0,229508	0,501080
Aquaculture	0,026922	0,084851	0,526193
Forestry	0,014400	0,045386	0,583953
Water	0,006530	0,002700	0,199205
Energy	0,079610	0,002600	0,234692
Manufacturing	0,292172	0,140000	0,481682
Construction	0,044659	0,062000	0,724319
Transport Water	0,008473	0,002291	0,564652
Transport Land	0,098791	0,026709	0,585345
Health	0,013103	0,009300	0,691108
Services	0,310710	0,294400	0,550248

  - Climate variables

Climate Variable	Value
Initial surface temperature, Celsius (tas)	22.4754892
Initial surface windspeed, m/s (SfcWind)	2.71182685
Initial surface precipitation flux, mm (pr)	2036.516996
Initial sunshine, hours per day (sunshine)	4.985255811
Initial surface relative humidity, % (hurs)	85.95630137
Initial heatwaves per year (heatwave)	0.009803922
Initial maximal consecutive dry days, (maxdrydays)	21.20634921
Initial maximal consecutive wet days, (maxwetdays)	30.38095238
Initial number of storms (equivalent to avg. historic storm) (storms)	0
Initial number of floods (equivalent to historic floods) (floods)	0
Initial number of forest fire (fire)	0
Initial landslides (landslide)	0

- Initial values

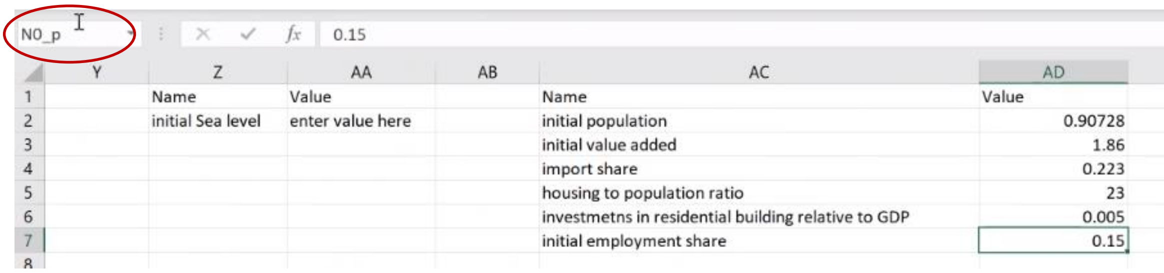
Variable	Value
Initial population	0.907280
Initial value added/GDP	1.86
Import share	0.223
Initial housing area	23
Initial employment level	0.15
Initial investment into housing	0.005

- Once the values have been added to the Excel file, use the program ***UpdateDataExcel.m*** to fill for structural parameters and baseline scenario values.

The solution to task 3

1. Open the ***ModelSimulationandCalibration12Sectorsand1Regions.xlsx*** file, located in the *ExcelFiles* folder. Copy each of the columns in the tables above and paste them in the appropriate Excel columns in the "Data" sheet.
2. Now go to the "Start" sheet and introduce the initial values for GDP, price level, population, employment level and investments into housing. Furthermore, add the initial climate parameter values, as well as initial VA and employment shares for each sector.

NOTE: If you want to add additional parameters defined for the model, or copy them to other positions in the file, make sure that cells are appropriate names. For example, the variable "initial employment level" is defined in cell C5 of the sheet "Start." However, we can move it to the empty cell AD7. In order to do this properly, make sure to type the name of the variable in cell AC7. Additionally, include the variable abbreviation, in this case, N0_p, in the upper left field. This is shown in the picture below.



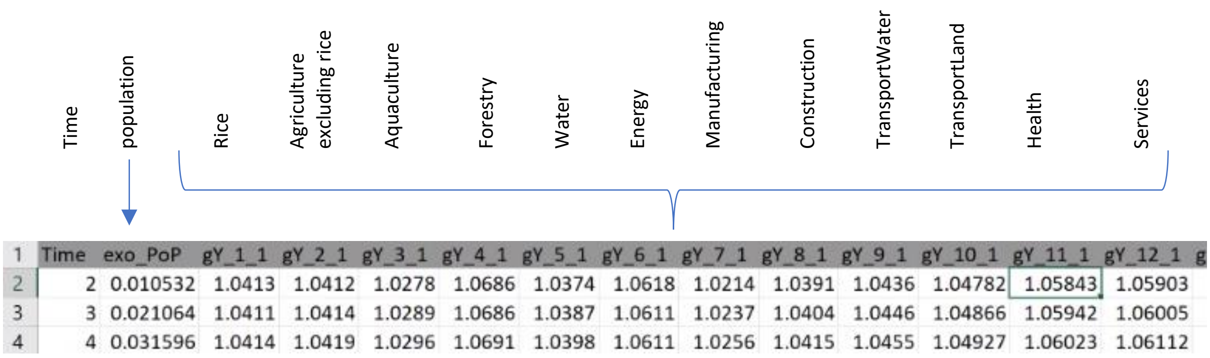
3. Once all the data has been inserted into the "Data" and "Start" sheets of the Excel file, run the ***UpdateDataExcel.m***. This program will fill the "Structural Parameters" and "Baseline" sheets automatically. However, the "Baseline" values are the same for years 1 through 100. We will deal with this issue in the following task.

Task 4: Define the Baseline scenario

- Define the subsectoral value added and employment growth rates, as well as population proyections
- Use data located in the ***BaselineProjections.xlsx*** file, under the ExcelFiles/Data directory.

The solution to task 4

1. Copy the values for each sector and parameter from the **BaselineProjections.xlsx** into the appropriate columns of the "Baseline" sheet of the **ModelSimulationandCalibration12Sectorsand1Regions.xlsx**. The mapping of the variables is the following.



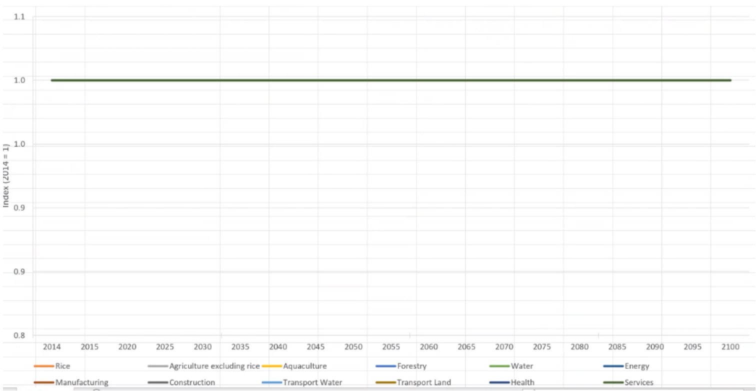
1	Time	exo_PoP	gY_1_1	gY_2_1	gY_3_1	gY_4_1	gY_5_1	gY_6_1	gY_7_1	gY_8_1	gY_9_1	gY_10_1	gY_11_1	gY_12_1	g
2	2	0.010532	1.0413	1.0412	1.0278	1.0686	1.0374	1.0618	1.0214	1.0391	1.0436	1.04782	1.05843	1.05903	
3	3	0.021064	1.0411	1.0414	1.0289	1.0686	1.0387	1.0611	1.0237	1.0404	1.0446	1.04866	1.05942	1.06005	
4	4	0.031596	1.0414	1.0419	1.0296	1.0691	1.0398	1.0611	1.0256	1.0415	1.0455	1.04927	1.06023	1.06112	

Task 5: Plot the Value Added for all sectors

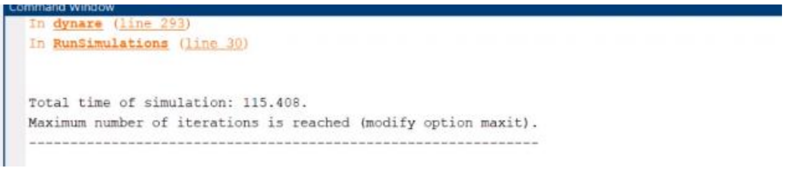
- Use the ***BaselineKeyindicators.xlsx*** Excel file in the *ExcelFiles/Figures* directory

The solution to task 5:

1. Execute the RunSimulations.m file on MATLAB and wait until the simulation has concluded.
2. Open the BaselineKeyIndicators.xlsx file. Define cell A1 as "Baseline" and cell A3 as "ResultsScenarios12Sectorsand1Regions." The figure should update and show that the growth rate in all sectors remains almost the same as in 2014.



If errors occur, such as:



Try to type in the command window

- CTRL+c → that will stop the program
- type sScenario → provides feedback whether the scenario is available

In sheet "Scenario" also add the population growth in column B (same as in baseline)

6.2 Practical Session 2

6.3 Practical Session 3

6.4 Practical session 4

6.5 Practical session 5

7 Appendix

7.1 MATLAB Functions

The different MATLAB functions needed to simulate the model are described in this section.

LoadExogenous.m

This function is included in the ***DGE_CRED_Model_Simulations.mod*** file, which is executed for each scenario specified in the ***DGE_CRED_Model.mod*** file. The inputs used by this program are the following:

- sWorkbookNameInput: Variable which contains the excel sheet *ModelSimulationandCalibrationKSectorsandRRegions.xslx*, where the exogenous variables for each scenario are specified.
- sScenario: Variable containing the specific scenario of the simulation
- oo_: Structure with various simulation results.
- M_: Structure with various information about the model.

The program reads the 'sScenario' sheet in 'sWorkBookNameInput' and determines which variables of the scenario are included in the model variables vector, 'M.exo_names.' This information is used to add values to the vector 'oo_.exo_simul', which contains the paths for the exogenous variables to be used in the simulation. Thus, the program produces the following output:

- oo_: *Dynare* structure containing various simulation results. More specifically, an updated version of oo_.exo_simul.

Simulation_Model.m

This Matlab script calls the perfect foresight solver of Dynare. It divides the original perfect foresight problem into small steps to find the solution to the problem. It calls the function ***FindK.m***.

SteadyState_Model.m

This Matlab script calls the respective excel input file and calibrates the model. It calls the ***DGE_CRED_Model_steadystate.m*** file and the ***Calibration.m***.

DGE_CRED_Model_steadystate. m

This function is called by the ***DGE_CRED_Model.mod*** file. The inputs used by this program are the following:

- ys: Vector containing initial values for the steady-state of the endogenous variables.
- Exo: Vector containing initial values for the steady-state of the exogenous variables.
- M_: *Dynare* structure for various information about the model.
- options_: Dynare structure for the values of various options used during the computation.

First, the program retrieves all the model parameters, as well as endogenous and exogenous variables, assigns them a unique name, and stores them into 'M_.params', 'ys' and 'exo', respectively. The next step is determined by the value of the 'lCalibration_p' variable, which can be set in the ***DGE_CRED_Model_Simulations.mod*** file.

- o If lCalibration_p = 0 or 3, the program uses the function ***FindK.m*** to find vectors with endogenous and exogenous variables that fulfil the static equations of the model such that the vector with residuals does not contain any elements larger than $1 \cdot 10^{-8}$.
- o If lCalibration_p = 2, the program calculates sectoral and regional output and labor values. These values are used by the ***FindK.m*** function to find vectors with endogenous and exogenous variables that fulfil the static equations of the model such that the vector with residuals does not contain any elements larger than $1 \cdot 10^{-8}$.
- o For any other values of lCalibration_p, the program uses the function ***Calibration.m*** to find the initial conditions to reflect a specific year.

Once the program has executed one of these tasks, the values in the vectors' M_.params', 'ys,' and 'exo' are updated according to the newly found steady state. Thus, the program produces the following output:

- ys: Updated vector containing initial values for the steady-state of the endogenous variables.
- exo: Updated vector containing initial values for the steady-state of the exogenous variables.
- M_: *Dynare* uses a MatLab structure to store various information about the model. In particular, an updated version of M_.params.

-

FindK.m

This function is stored in the **steadystate.m** and used for lCalibration_p values of 0 or 3, and 2. The inputs used by the function are the following:

- x: Vector of initial values for the steady-state of the regional and sectoral capital stock.
- strys: Structure containing all endogenous variables of the model.
- strexo: Structure containing all exogenous variables of the model.
- strpar: Structure containing all parameters of the model.

The program begins by calculating the initial value of the exogenous variables and storing them in 'strexo.' These include effective exchange rate with the rest of the world, sea level, population, price level, and regional and sectoral capital stock. The following block is responsible for computing the initial values for different production factors and output at the sectoral and regional level, including tax rates, productivity, and investment, and storing them in 'strys'. Different damage functions are also computed and stored in 'strys' depending on whether the scenario under consideration is 'SeaLevel' or not.

Next, sectoral and regional price indices and sectoral aggregates, such as investment, wages, and capital stock, are calculated and included in 'strys.' Once these calculations are performed, national aggregates are computed for investment, capital, output, labour, as well as wage and capital taxes, and adaptation costs and stored in 'strys.' The final block of this function determines the residual of the Household's first-order conditions with respect to labour and records the result in 'fval_vec.' As a result, the program produces the following outputs:

- fval_vec: vector with residuals of regional and sectoral first-order conditions of households with respect to labour.
- strys: Structure containing updated endogenous variables of the model.
- strexo: Structure containing updated exogenous variables of the model.

Calibration.m

This function is stored in the **steadystate.m** and used for lCalibration_p values different than 0, 3, or 2. The inputs used by the function are the following:

- x: Vector of initial values for the steady-state of the regional and sectoral capital stock.
- strys: Structure containing all endogenous variables of the model.
- strexo: Structure containing all exogenous variables of the model.

The program begins by calculating the initial value of the exogenous variables and storing them in 'strexo.' These include effective exchange rate with the rest of the world, sea level, population, price level, and regional and sectoral capital stock. The following block is responsible for computing the initial values for different production factors and output at the sectoral and regional level, including tax rates, productivity, and investment, and storing them in 'strys'. Different damage functions are also computed and stored in 'strys' depending on whether the scenario under consideration is 'SeaLevel' or not.

Next, sectoral and regional price indices and sectoral aggregates, such as investment, wages, and capital stock, are calculated and included in 'strys.' Once these calculations are performed, national aggregates are computed for investment, capital, output, labour, as well as wage and capital taxes, and adaptation costs and stored in 'strys.' Once these calculations are performed, national aggregates are computed for investment, capital, output, labour, as well as wage and capital taxes, and adaptation costs and stored in 'strys.'

The final section of this function consists of two parts. The first one computes the labour disutility parameters and stores them into 'strys.' Additionally, the right-hand side of the Household's first-order condition with respect to labour is calculated and fitted into the left-hand side to find a value of the coefficient of disutility to work that satisfies the condition. The second part of the function uses the formula for sectoral SEC aggregation to obtain the implicit output and compares it with the initial output value. The output of this program is the following:

- fval_vec: Vector containing the percentage difference between the implicit and initial output for every region.
- strys: Structure containing all endogenous variables of the model.
- strpar: Structure containing all parameters of the model.