Nested Functions Exercises 3

1

Outline

- Introduction
- smod_t2.gms with different data
  - Adjust model data load
  - Rerun previous experiment
- smod_t3.gms
  - Assign equations
  - Check
  - Rerun previous experiment
  - Compare results
- Change nesting structure in smod_t3.gms
  - Data
  - Rerun previous experiment
- Comparing Results

2

smod_t2 with different data

3

Data

Well-designed models should be able to run with different data when

- Behavioural relationships are **unchanged**
- SAM structure is **unchanged**

Review `smod_t_mod06_t3.xlsx`

Same SAM structure as in *smod_mod06_t2.xlsx*

10	commodities/activities
5	'natural' factors
2	households
5	Tax instruments

ALL elasticity values are 'place holders'

4

cgemod

Change Data for smod_t2

1. Open `smod_t2.gms` in Studio
2. In the file `smod_t2.gms` open the file `smod_t2_load.inc` (use `Ctrl+click` on the `$INCLUDE` statement)
3. Change the `$GDXXRW` instruction to load the data file `smod_t_mod06_t3.xlsx`
4. Run the programme (`F10`) and check the model is running correctly

The ability to change data used by a 'standard' model is valuable

- When making changes to behaviour in a model starting with an aggregated SAM is useful
- Sometimes a model can be used for different SAMs

Do NOT change data to match a model

5

cgemod

Experiment

Trade policy experiment from Module O5

Experiment File

1. Use the file `smod_t2_exp2.inc` used with `smod_t2`,
2. Make any necessary changes to the code for saving results
   i. Extend the set scal to include the additional scalar results and extend the results reported in *levSCAL* and *pcSCAL*.
   ii. Declare parameters results for the new variables, i.e., *lev*** and *pc***, and assign values to these parameters.
3. Add the instruction `gdx=[unique file name]` in the command line
4. Now run the programme (`F10`) with the same experiment file used for the `smod_t2` exercises.
5. Review the results

For each experiment add the instruction `gdx=****` in the command line, where `****` identifies different experiments.

6

cgemod

cgemod

Equation Coding for smod_t3.gms

7

cgemod

smod_t3.gms coding

- Complete data file provided
- All sets declared and assigned
- All parameters declared and assigned
- All variables declared and initial values assigned
- All equations declared

Your tasks

1. Assign equations
2. Revise model definition
3. Revise factor clearing
4. Run trade policy experiment
5. Compare results

8

cgemod

cgemod

Data

Review `smod_t_mod06_t3b.xlsx`

Same SAM structure as in *smod_mod06_t2.xlsx and smod_t_mod06_t3.xlsx*

1. What mapping sets are declared?
2. What parameters are declared?
3. How many commodities, activities, factor and households are declared and assigned?

ALL elasticity values are 'place holders'

9

cgemod

Using $STOP

GAMS Configuration: set to produce *smod***.gdx* and *smod***.ref* files with *F10*

Progress One 'Baby' Step at a Time

- Use **$stop** to set end of code run

Step 1:

- Data Entry
- Data adjustments and scaling
- Data Diagnostics
- Additional set assignments
- Parameter assignments
- Variable initialisation

Checks for

- syntax errors
- data errors
- (some) coding errors

Gets to starting point for model coding

10

cgemod

cgemod

Equation Assignment – smod_t3

Equations needed are already declared

Transform the algebraic expressions, in the `smod_t` technical document (Appendix 3), into GAMS code.

Tricky parts

- use of LHS and RHS `$` control at different levels of the production nest
- use of `$` control in parameter calibration

'new' variables already initialised

11

cgemod

Some Questions

- Why are the sets *ff(sac)* and *f(ff)* necessary? Note the following
  - What is the difference between the members of *fag* and *f*?
  - Factor incomes (*YF*) are only received for natural factors, *f*. Why?
  - Do the aggregate factors, members of *fag*, have 'real world' identities? If not, what are their roles in the system?
- What are the roles of the $ controls on equations for *QVAFOC(ff,a)* and *FDPRODFN(ff,a)* and *FDFOC(ff,ffp,a)* (mapping sets run TO FROM)
  - *map_va_ff(ff,a)*
  - *map_fagg_ff(ff,ffp,a)*
- What are the roles of the RHS $ controls in *QVAPRODN(a)*, *QVAFOC(ff,a)* and *FDPRODFN(ff,a)* and *FDFOC(ff,ffp,a)*
  - *map_va_ff(ff,a)* and *map_fagg_ff(ff,ffp,a)*
  - *deltava(ffp,a)* and *deltafd(ff,ff,a)*
- Why are the following statements that fix variable values needed
  - `FD.FX(f,a)$(NOT SAM(f,a)) = 0.0 ;`
  - `WF.FX(ff)$fag(ff) = WF0(ff) * CPI.L ;`

12

cgemod

Model Definition & Closure Conditions

Defining the Model

The new equations need to be added to the model definition. This has been done for you in the template.

The model closure file – `smod_t3_cl_base.inc` – updated

- Compare this closure file with the closure file `smod_t2_cl_base.inc`
- Explain why the changes are needed

Practical CGE, 2024 © cgemod 13

13

Check the Model

1. Check the data in the model are the intended data.
2. Check that the value for `VAR WALRAS` is zero.
3. Check that the basic prices (`PE, PD, PM`) are equal to one.
4. Check that all entries in `ASAM1CHK` are equal to zero
5. Check that all entries in `ASAM2CHK` are equal to one
6. Check the `LHS` values.
7. Numéraire check.

Only after these checks have been passed should you move on to using the model.

Practical CGE, 2024 © cgemod 14

14

cgemod

A Simple Experiment

15

cgemod

Experiment

Trade policy experiment from Module O5

Experiment File

1. Use the file `smod_t2_exp2.inc` used with `smod_t2`,
2. Make any necessary changes to the code for saving results
   i. Extend the set scal to include the additional scalar results and extend the results reported in *levSCAL* and *pcSCAL*.
   ii. Declare parameters results for the new variables, i.e., *lev*** and *pc***, and assign values to these parameters.
3. Add the instruction `gdx=[unique file name]` in the command line
4. Now run the programme (`F10`) with the same experiment file used for the `smod_t2` exercises.
5. Review the results

For each experiment add the instruction `gdx=****` in the command line, where `****` identifies different experiments.

16

cgemod

cgemod

Comparing Results

17

cgemod

Comparing Results

Comparison of `smod_t2.gms` with `smod_t3.gms`

Comparison of the Results

1. To compare the results across the 2 different assumptions about factor market clearing, use the GDX MERGE utility. Open the file `compare.gms` and save it as `compare**.gms`.
2. Edit the call statement to refer to the gdx files generate by each run of the model.
3. Run `compare**.gms` (make sure `compare**.gms` is the Main File before running the model). Review the merged file, MERGE.GDX.

Analyse the results

Emphasise explanation not simple reporting of the results.

1. Factor demands (*FD*)
2. Factor prices (*WF* and *WFDIST*) and factor incomes (*YH*)
3. Value added (*QVA*) and intermediate inputs (*QINT*)
4. Production structure (*QX* and *QXC*)
5. Domestic quantities (*QD* and *QQ*)
6. Household incomes (*YH*) consumption (*HEXP* & *QCD*) & welfare (*EV*)
7. etc.

18

cgemod

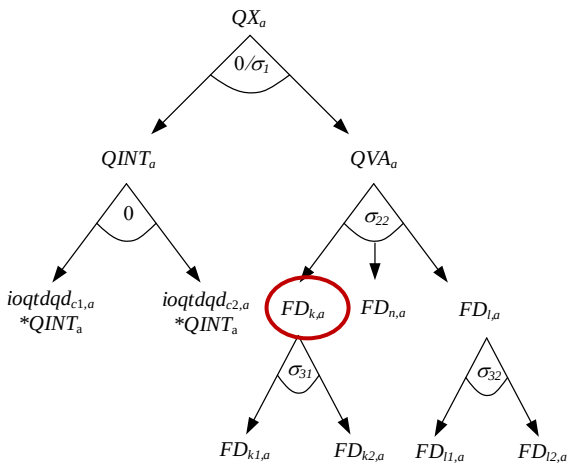
cgemod

Changing the Production Nest

19

cgemod

Possible Three Level Production Nest



Two types of capital
- 'Fixed' capital
- 'Flexible' capital

Elasticity – σ_{31}

Two types of labour
- 'Skilled' labour
- 'Unskilled' labour

Elasticity – σ_{32}

20

cgemod

cgemod

Tasks

- Extend the sets *ff* and *fag* to include aggregate labour
  - Make changes in an Excel file
  - Extend the set *sac*
- Modify the maps that control the nesting
  - *map_va_ff(ff,a)*
  - *map_fagg_ff(ff,ff,a)*
- Checks (use `$stop`?)
  - data intended are being loaded
  - parameters are correctly assigned
  - model runs correctly
- Run the experiment (different model)
- Compare the results across different models with the same data

21

cgemod

Comparing Results

22

cgemod

cgemod

Comparing Results

`smod_t2.gms` with variants of `smod_t3.gms`

Comparison of the Results

1. To compare the results across the 2 different assumptions about factor market clearing, use the GDX MERGE utility. Open the file `compare.gms` and save it as `compare**.gms`.
2. Edit the call statement to refer to the gdx files generate by each run of the model.
3. Run `compare**.gms` (make sure `compare**.gms` is the Main File before running the model). Review the merged file, MERGE.GDX.

Analyse the results

Emphasise explanation not simple reporting of the results.

1. Factor demands (*FD*)
2. Factor prices (*WF* and *WFDIST*) and factor incomes (*YH*)
3. Value added (*QVA*) and intermediate inputs (*QINT*)
4. Production structure (*QX* and *QXC*)
5. Domestic quantities (*QD* and *QQ*)
6. Household incomes (*YH*) consumption (*HEXP* & *QCD*) & welfare (*EV*)
7. etc.

23

cgemod

The End

Nested Functions Exercises 2

24

cgemod