ACR-LQ

license BSD 3-clause   version v1.0.0

ACR-LQ is a package of codes to generate a quadratic approximation to the welfare function and solve for the correct first order approximation solution to the optimal policy problem. It also checks the second order conditions.

The codes were developed for

Altissimo, F., Cúrdia, V., and Rodriguez-Palenzuela, D. (2005) Linear-Quadratic Approximation to Optimal Policy: An Algorithm and Two Applications Unpublished, European Central Bank and Princeton University.

The codes implement the linear-quadratic approximation method described in

Benigno, P. and Woodford, M. (2012) Linear-quadratic approximation of optimal policy problems *Journal of Economic Theory* 147(1), pp. 1-42.

These replication codes are available online at:
https://github.com/vcurdia/ACR-LQ

Requirements

Matlab (R)

These codes have been tested on Matlab (R) R1016b with following toolboxes

- Symbolic Toolbox

Additional Matlab (R) codes

- gensys by Chris Sims
- optimize by Chris Sims

Description of main files

`LQGenSymVar.m`

Creates symbolic variables needed in other routines. See example in order to see when to call this script.

`LQ.m`

Main LQ code. It computes synmbolic derivatives and generates all the symbolic matrices needed to implement optimal policy, including the quadratic approximation to the welfare function and the first order approximation to the laws of motion of the economy.

`LQSolveREE.m`

After replacing the output matrices from LQ with steady state values, this code solves for the REE equilibrium using gensys routine created by Chris Sims and available through his website.

`LQCheckSOC.m` and `LQCheckSOCAlt.m`

Two alternative codes to check for verification of the second order conditions.

`LQAltRule.m`

Computes the REE under an alternative policy rule submitted in non-linear form.

`LQAltRuleLinear.m` Computes the REE under an alternative policy rule submitted in linear form.

`LQWEval`

Evaluates the welfare value of a given policy rule. Current limitation: assumes that policies are stationary.

Example:

The package also includes an example (in folder Example) showing how to use the codes. The example's main script is the file `MonFrictions.m` which sets up the model framework and calls other functions and scripts.

The codes in the examples are the following:

`MonFrictions.m`

Example's main script. Sets the required variables and equations. Solves for the optimal steady state using the csolve numerical solution to systems of non-linear equations created by Chris Sims and available through his website. then calls the `LQ` function. Replaces the symbolic matrices with numerical steady state values and then solves for the REE solution under optimal policy using `LQSolveREE`. It also computes the IRF for all variables in response to all of the shocks. It also computes the value of welfare under optimal and simple rule. Finally it saves all the output in a mat file.

`MonFrictionsIRFPlot.m`

Plots the IRFs. This is an example of how to make simple plots of the IRF.

`MonFrictionsIRFPlotAltRule.m`

Plots the optimal policy against the alternative rule.

`MonFrictionsIRFPlotComp.m`

Generates comparison plots for alternative calibrations of the model.

Additional Information

Each of the functions and scripts contains help at the beginning of the codes, which can be accessed using the help or doc commands in matlab. E.g.:

```
help LQ
```

or

```
doc LQ
```