Maximum Likelihood Estimation

Maximum Likelihood in Dynare

Tools for Macroeconomists: The essentials

Petr Sedláček

Maximum Likelihood in Dynare

Maximum Likelihood in Dynare

- essentially the same program as when solving a model
- only instead of solving, we tell Dynare to estimate
- several additional options and requirements

Maximum Likelihood in Dynare

Dynare and its Structure

We've seen essentially all of this before...

Dynare produces perturbation approximation to policy rules

We've seen essentially all of this before...

Dynare produces perturbation approximation to policy rules

- in neoclassical growth model
- Dynare generates following policy rules

$$k_t = \overline{k} + a_{kk}(k_{t-1} - \overline{k}) + a_{kz}(z_{t-1} - \overline{z}) + a_{k\epsilon}\epsilon_t$$

We've seen essentially all of this before...

Dynare produces perturbation approximation to policy rules

- in neoclassical growth model
- Dynare generates following policy rules

$$k_t = \overline{k} + a_{kk}(k_{t-1} - \overline{k}) + a_{kz}(z_{t-1} - \overline{z}) + a_{k\epsilon}\epsilon_t$$

- i.e. it splits structural shocks into
  - past value and
  - innovation
  - i.e. if $z_t = 1 - \rho + \rho z_{t-1} + \epsilon_t$ then $a_{kz} = \rho a_{k\epsilon}$

Dynare blocks

A Dynare file has several blocks:

1. list of variables
2. list of exogenous shocks
3. list of model parameters and their values
4. model block (optimality conditions)
5. shock properties
6. initial values
7. solution (and other) commands

Definitions and parametrization

1. Specify variables
   - specified by typing “var” and then listing variables
2. Specify exogenous shocks
   - specified by typing “varexo” and then listing shocks
3. Specify parameters and their values
   - specified by typing “parameters” and then listing parameters
   - each parameter must then be assigned a value
     - either directly in Dynare file
     - or by loading it from outside Dynare file
     - the latter is more convenient for calibration

4. Model block contains equilibrium conditions

Model block

4. Model block contains equilibrium conditions

- initialize block by typing “model;”
- end it by typing “end;”
- in between simply write your model equations

Specifics

- Dynare figures out there are expectations when you write $t+1$
- e.g. the Euler equation:

```
c^(-gamma)=beta*c(+1)^(-gamma)*(alpha*Z(+1)*k^(alpha-1)+1-delta)
```

Shock properties

5. Shock properties
   - initialize the block by typing “shocks;”
   - end it by typing “end;”
   - in between specify shock properties
     - e.g. “var e; stderr sigZ;”
     - can specify more, like correlations etc.

Initial values

6. Initial values
   - initialize block by typing “initval;”
   - end it by typing “end;”
   - inbetween list the initial values of all variables
     - ideally give Dynare the steady state
     - often difficult to compute, so supply it yourself

Solution

7. Give Dynare the green light to solve the model
   - "`stoch_simul(options)`"
   - options include
     - order of perturbation: e.g. "order=1" for linear
     - length of IRFs: e.g. IRF=20
     - many, many more

To actually run Dynare type `dynare filename.mod`

Other useful features

- "resid" command shows equation errors
  - it plugs initial values into model equations
  - they should all be zero in steady state
  - useful for finding out typos

Maximum Likelihood in Dynare

Estimating DSGE models in Dynare

Neoclassical growth model again

$$c_t^{-\nu} = \mathbb{E}_t \left[\beta c_{t+1}^{-\upsilon} (\alpha z_{t+1} k_t^{\alpha-1} + 1 - \delta) \right]$$

$$c_t + k_t = z_t k_{t-1}^{\alpha} + (1-\delta) k_{t-1}$$

$$z_t = 1 - \rho + \rho z_{t-1} + \epsilon_t$$

$$\epsilon_t \sim N(0, \sigma^2)$$

ML estimation in Dynare: initialization

- initialize as usual

```
var c, k, z, y;
varexo e;
parameters beta, rho, alpha, nu, delta, sigma;
```

ML estimation in Dynare: initialization

- initialize as usual

```
var c, k, z, y;
varexo e;
parameters beta, rho, alpha, nu, delta, sigma;
```

- set values for all parameters (even those that are estimated)

```
alpha = 0.36;
rho = 0.95;
beta = 0.99;
nu = 1;
delta = 0.025;
sigma = 0.01;
```

ML estimation in Dynare: setting it up

- after model part, and specification of steady state
- tell Dynare which parameters he should estimate

ML estimation in Dynare: setting it up

- after model part, and specification of steady state
- tell Dynare which parameters he should estimate

```
estimated_params;
stderr e, 0.01, 0, 0.2;
end;
```

- the above tells Dynare to
  - estimate σ, the st. error of the productivity disturbance
  - 0.01 is the initial value (starting point for minimization routine)
  - 0 is the lower and 0.2 is the upper bound (optional)

ML estimation in Dynare: steady state

- steady state calculated for many different values of Ψ!
- solve for the steady state yourself (linearizing makes it easier)
- give the exact steady state to Dynare for the initial values
- option to provide own function that calculates steady state!
  - `modfilename_steadystate.m` or
  - `steady_state_model;` block
- have not specified it in the above, why?

ML estimation in Dynare: estimation command

- then also tell Dynare which are the observable variables

```
varobs y;
estimation(options)[VARIABLE_NAME];
```

- options include
  - specify data file for estimation: `datafile=data`
  - assess convergence to max: `mode_check`
  - optimization options: `optim(options)`
  - many more!

Maximum Likelihood in Dynare

Decomposition

ML estimation in Dynare: decomposition

- decompose endogenous variables into contribution of shocks
- possible also after `stoch_simul`

```
shock_decomposition(options) [VARIABLE_NAME];
```

- options include e.g. `parameter_set`
  - use calibrated values: `=calibration`
  - use prior/posterior mode: `=prior_mode`/`=posterior_mode`
- variables specifies for which variables to run the decomposition

Maximum Likelihood in Dynare

Example

ML estimation of neoclassical growth model

- use neoclassical growth model as data generating process
- 265 observations of output
- use ML to estimate
  - σ
  - σ, ρ, δ, α

ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

Estimating only σ:

ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

Estimating only σ:

```
estimated_params;
stderr e, 0.01, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

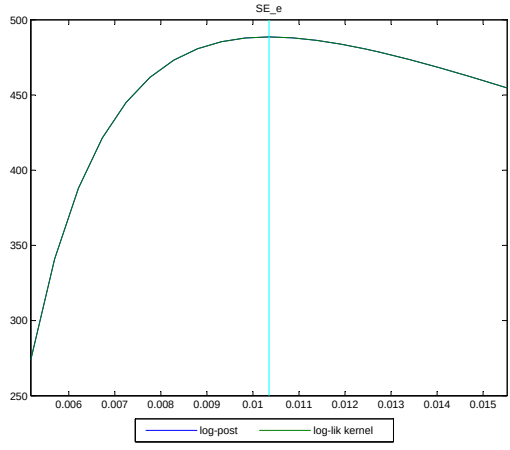
Estimating only σ:

```
estimated_params;
stderr e, 0.01, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

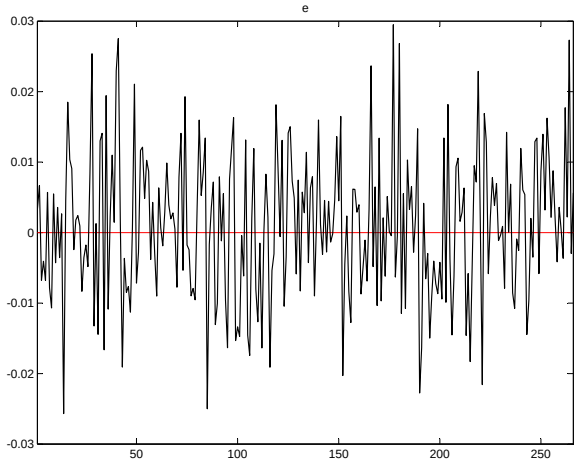
Results:

σ	0.0103 (0.0004)

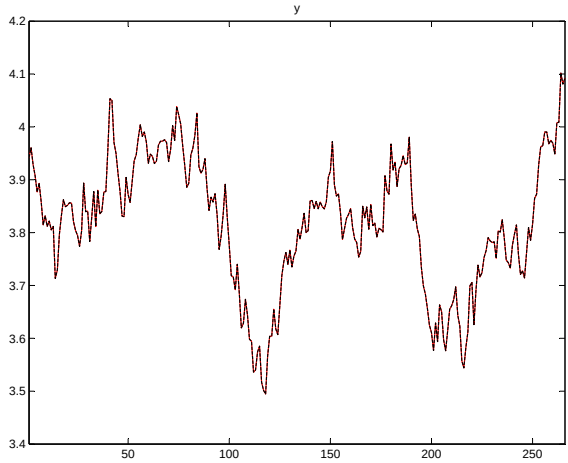
ML estimation of easy case: mode check



ML estimation of easy case: shocks



ML estimation of easy case: fitted values



ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

Estimating only σ:

Estimating only σ:

```
estimated_params;
stderr e, 0.1, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

Estimating only σ:

```
estimated_params;
stderr e, 0.1, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

Results:

σ	0.0103 (0.0004)

ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

Estimating more than just σ:

ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

Estimating more than just σ:

```
estimated_params;
stderr e, 0.01, 0, 0.2;
rho, 0.95, 0, 1;
alpha, 0.36, 0, 1;
delta, 0.025, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

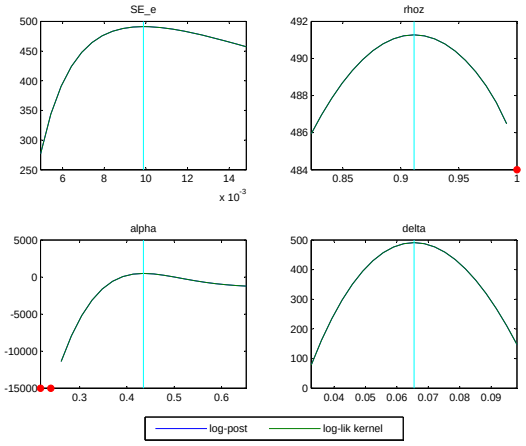
Estimating more than just σ:

```
estimated_params;
stderr e, 0.01, 0, 0.2;
rho, 0.95, 0, 1;
alpha, 0.36, 0, 1;
delta, 0.025, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

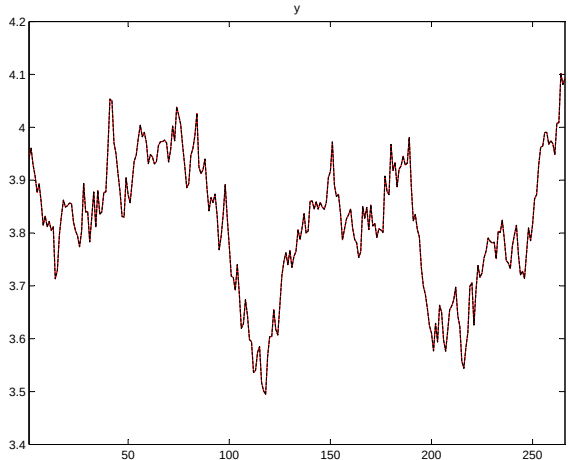
Results:

σ	0.0099 (0.0004)	ρ	0.9114 (0.0676)
α	0.4349 (0.1290)	δ	0.0655 (0.0921)

ML estimation of tough case: mode check



ML estimation of tough case: fitted values



ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

Estimating more than just σ:

ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

Estimating more than just σ:

```
estimated_params;
stderr e, 0.1, 0, 0.2;
rho, 0.95, 0, 1;
alpha, 0.36, 0, 1;
delta, 0.025, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

ML ESTIMATION OF NEOCLASSICAL GROWTH MODEL

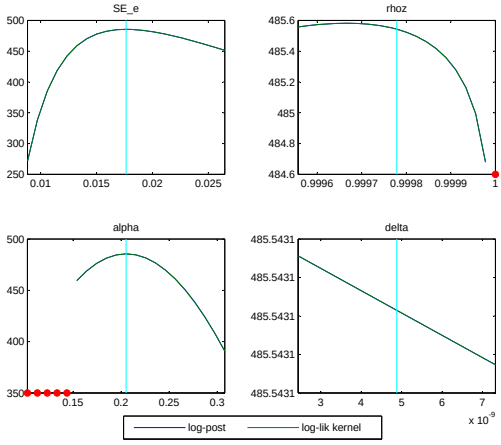
Estimating more than just σ:

```
estimated_params;
stderr e, 0.1, 0, 0.2;
rho, 0.95, 0, 1;
alpha, 0.36, 0, 1;
delta, 0.025, 0, 0.2;
end;
varobs y; estimation(datafile=y,mode_check) c, k, y;
```

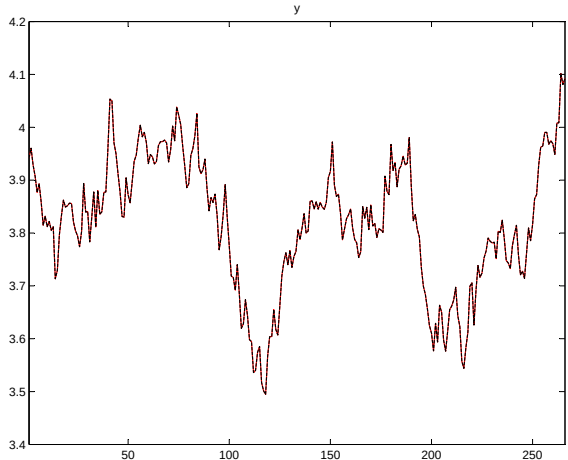
Results:

σ	0.0177 (0.0046)	ρ	0.9998 (0.0005)
α	0.2053 (0.0444)	δ	0.0000 (0.0003)

ML estimation of tough case: mode check



ML estimation of tough case: fitted values



ML estimation of neoclassical growth model

- observed data consistent with many parameter combinations
- the likelihood is typically quite flat
- $\rightarrow$ Maximum Likelihood has trouble converging
- solutions:
  - sometimes alternative optimization algorithms work
  - use of extra (prior) information $\rightarrow$ Bayesian estimation

Maximum Likelihood in Dynare

Taking Stock

Estimating DSGE models with Maximum Likelihood in Dynare

- same structure of program as when solving a model
  - beware of same pitfalls (timing, notation)

Estimating DSGE models with Maximum Likelihood in Dynare

- same structure of program as when solving a model
  - beware of same pitfalls (timing, notation)
- estimation command straightforward
  - specify data, which variables are observed and initial values


That's all Folks!