

Dynare

Všechno o programu Dynare se dá najít na www.dynare.org.

Stáhnout toolbox z webu nebo ve škole je na disku `Q:\Matlab_Toolbox\dynare\`. V adresáři `doc` najdete dokumentaci `guide.pdf` nebo `UserGuide.pdf`.

Po stažení je třeba nastavit cestu (File – Set Path – Add with Subfolders) nebo napsat do příkazového řádku `addpath q:\Matlab_Toolbox\dynare\4.1.3\matlab`

Pro naše účely je ještě potřeba mít soubor `disp_dr.m` v adresáři, kde pracujeme, případně k němu nastavit cestu. Soubor si stáhněte z mého webu nebo stránky Woutera DenHaana.

Dynare kód

Dynare pracuje s vlastním typem souboru `.mod` file. Zvláštnosti

- komentář se píše za `//`
- konec řádku (bloku kódu) musí být označen pomocí `;`
- Preamble
 - definování *endogenních* proměnných `var` ¹
 - definování *exogenních* proměnných `varexo`
 - definování parametrů a steady statových hodnoty `parameters`
- Tělo modelu mezi `model;` a `end;`
 - proměnná v čase t bez označení, `c`
 - proměnná v čase $t + 1$ se označí, `c(+1)`
 - proměnná v čase $t - 1$ obdobně, `c(-1)`
- ALE! v Dynaru je důležité, kdy je proměnná *vybraná*. Proměnná *vybraná* v čase t mít tomu odpovídající označení (tedy t).
 - naše k_{t+1} vybíráme v čase t , v Dynare kódu je bráno jako k_t tedy `k`
 - naše k_t vybíráme v čase $t - 1$, v Dynare kódu je bráno jako k_{t-1} tedy `k(-1)`
- `steady;` vypočítá steady state
- `check;` zkontroluje, zda existuje (jediné) řešení modelu
- `shocks;` a `end;` definování exogenních proměnných a jejich rozptylu (směrodatné odchylky)
- příkaz `stoch_simul(...)`; udělá, co se mu řekne. Simulace, impulsní odezvy, spočítá momenty, varianční dekompozice atd. My si necháme najít pouze *rozhodovací pravidlo* – decision rule (Dynare mu říká *policy function*). Zbytek si uděláme sami.

¹Šok v technologii (TFP) byl v našem modelu exogenní stavovou proměnnou. Zde je endogenní proměnnou, exogenní proměnnou je pouze inovace (disturbance) ϵ_t

Linearizované rozhodovací pravidlo

$$k_t - \bar{k} = a_{k,k}(k_{t-1} - \bar{k}) + a_{k,z}(z_{t-1} - \bar{z}) + a_{k,\epsilon}\epsilon_t$$

případně v odchylkách

$$\hat{k}_t = a_{k,k}\hat{k}_{t-1} + a_{k,z}\hat{z}_{t-1} + a_{k,\epsilon}\epsilon_t$$

Dynare ale používá tento zápis:

$$k_t = \bar{k} + a_{k,k}(k_{t-1} - \bar{k}) + a_{k,z}(z_{t-1} - \bar{z}) + a_{k,\epsilon}\epsilon_t$$

Aby rozhodovací pravidlo vypadalo stejně jak Dynare vypíše na obrazovce, použijeme soubor `disp_dr.m`.

Spuštění Dynare souboru, stačí napsat do příkazového řádku `dynare nizev_souboru.mod`. (Dynare sám vytvoří několik m.file, který pak Matlab spouští ... `nizev_souboru.m`, `nizev_souboru_static.m`, `nizev_souboru_dynamic.m`)

`Const` odpovídá steady statové hodnotě, `k(-1)` značí devakto odchylku od steady statu $k_{t-1} - \bar{k}$. Čísla v maticích odpovídají parametrům $a_{k,k}$, $a_{k,z}$ atd.

Využijeme jen rozhodovací pravidlo (vše ostatní si uděláme sami). Funkce `disp_dr.m` uloží rozhodovací pravidlo (v podobě jako je na obrazovce) do matice `decision.m`, která je uložena v souboru `dynarerocks.mat`

Pozor! Pořadí proměnných v rozhodovacím pravidle je podle definice v preambuli `.mod` souboru nebo podle pořadí za příkazem `stoch_simul(...)` `promenne;`. Rozhodovací pravidlo pro každou proměnnou odpovídá sloupci v matici `decision.m`

Vlastní kód

Vytvoříme si vlastní m.file. Struktura:

- indexování proměnných dle matice `decision`
- předdefinování proměnných a šoků
- cyklus pro šok a další proměnné (impulsní odezva, šok jen ve jednom období
- vykreslení

Dynare umí všechno možné. Ale někdy je lepší přístup *udělej si sám*

Příkaz `stoch_simul(periods=1000, irf=100)` nasimuluje chování modelu v reakci na 1000 šoků a udělá impulsní odezvy pro délku 100. Statistiky vypíše na obrazovku a vše uloží do strukturované matice `oo_`. Umí i odhadovat pomocí metody maximální věrohodnosti nebo Bayesovskými metodami.

Co dál?

Na webu existuje databáze modelů z různých článků a studií, které jsou přepsané do Dynare kódu. <http://www.macromodelbase.com/>. Stačí se zaregistrovat, stáhnout si je a můžete s nimi pracovat dle libosti.