Materials 12 - tinkering around with policy and expectation formation

Laura Gáti

December 9, 2019

Overview

1 Model summary

$$x_t = -\sigma i_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} \beta^{T-t}\big((1-\beta)x_{T+1} - \sigma(\beta i_{T+1} - \pi_{T+1}) + \sigma r_T^n\big) \tag{1}$$

$$\pi_t = \kappa x_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} (\alpha\beta)^{T-t}\big(\kappa\alpha\beta x_{T+1} + (1-\alpha)\beta\pi_{T+1} + u_T\big) \tag{2}$$

$$i_t = \psi_\pi \pi_t + \psi_x x_t + \rho i_{t-1} + \bar{i}_t \tag{3}$$

$$\hat{\mathbb{E}}_t z_{t+h} = \begin{bmatrix} \bar{\pi}_{t-1} \\ 0 \\ 0 \end{bmatrix} + b h_x^{h-1} s_t \quad \forall h \geq 1 \qquad b = g_x\ h_x \qquad \text{PLM} \tag{4}$$

$$\bar{\pi}_t = \bar{\pi}_{t-1} + k_t^{-1} \underbrace{\big(\pi_t - (\bar{\pi}_{t-1} + b_1 s_{t-1})\big)}_{\text{fcst error using (4)}} \qquad (b_1 \text{ is the first row of } b) \tag{5}$$

$$k_t = \begin{cases} k_{t-1} + 1 & \text{for decreasing gain learning} \\ \bar{g}^{-1} & \text{for constant gain learning.} \end{cases} \tag{6}$$

2 Some initial quick changes

1. To policy

   (a) $\mathbb{E}(\pi)$ instead of π in TR

   (b) Check the fake $\psi_\pi < 1$ exercise.

2. To expectation formation

   (a) Curiosity: check IRFs from Euler equation learning

   (b) IRFs from vector learning (meaning learn all observables)

   (c) Different implications from Bayesian learning?

Some reasoning (motivation and results):

1. $\mathbb{E}(\pi)$ instead of π in TR: indeed makes overshooting larger in magnitude b/c policy is reacting to something that moves more.

2. $\psi_\pi \leq 1$: indeed kills the overshooting, but - no surprise - makes observables unstable (IRFs don't return to steady state). Why does it work to kill the overshooting? B/c the Ball-effect of anticipated interest rate reactions no longer overweighs (less expectational feedback).

3. Townsend (1983) investigates "forecasting the forecasts of others" and finds damped oscillations $\rightarrow$ do higher-order beliefs play a role for causing oscillations in learning? If so, EE learning IRFs should exhibit no oscillations (and indeed they do not!)

4. Vector learning: are model implications different when agents learn the LOM of not only inflation but also of the other variables? $\rightarrow$ No. (Note: I'm using the same gain for all variables.)

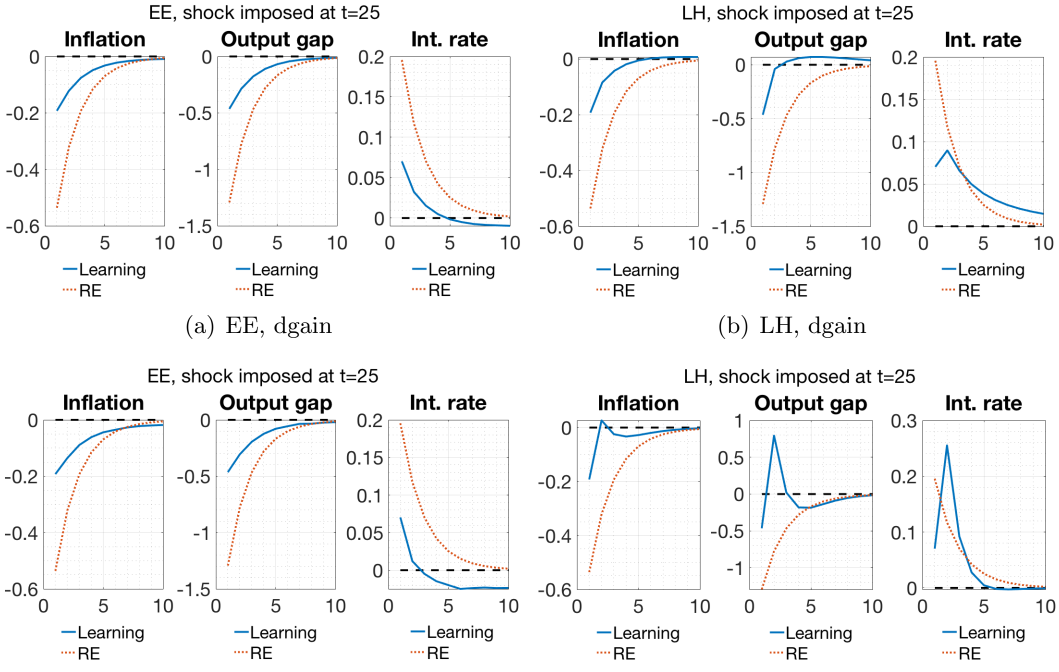
5. Does learning both slope and constant make a difference? $\rightarrow$ Yes, in particular for constant gain learning. 2 effects: 1) less foresight, so i needs to be less expansionary 2) more bumpy IRFs.

   1) I think what might be going on here is that the only thing agents now know is h_x. Therefore the Ball-type "disinflationary boom"-effect happens to a lesser extent b/c agents do not internalize movements in the interest rate in response to future inflation as much as they would otherwise (feedback from expectations is lower).

   2) More bumpy because since you're learning b, the loading on shocks, the specific sequence of shocks matters. Increasing the size of the cross-section, N, mitigates this somewhat.

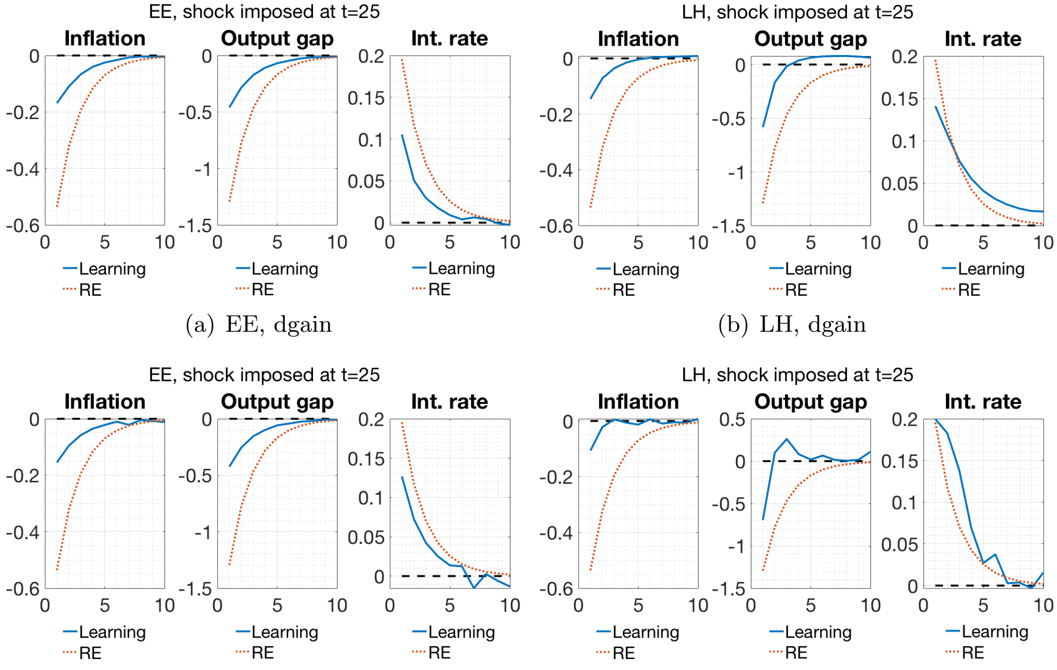
3 IRFs from vector learning: EE and LH, $T = 400, N = 100$

Figure 1: Learning constant only


(a) EE, dgain (b) LH, dgain

(c) EE, cgain (d) LH, cgain

Figure 2: Learning slope and constant


(a) EE, dgain (b) LH, dgain

(c) EE, cgain (d) LH, cgain

4 A technical note on the projection facility

Contrary to Liam Graham, I never have explosive path issues for long-horizon learning, but I do sometimes for Euler-equation learning (Graham claims this is never an issue for EE learning). Graham's solution for the projection facility is to check the eigenvalues of the learning matrix ϕ. My silly issue is that ϕ is not square. Therefore what I do is I check the eigenvalues of the following cheating matrix $(\phi\phi').^{1/2}$, where ".1/2" denotes the square root of the elements. Thoughts?

5 A note on the empirical counterpart: IRFs in the data

I note in passing that contrary to my initial priors, one possibility is that the theoretical IRFs generated by learning models actually fit empirical IRFs if the latter are properly computed. Now I base this argument not only on your oscillating IRFs from your CEE-replication, but more broadly on the discussion of computing IRFs in Valerie Ramey's handbook chapter (pointed to by Susanto):

- "iterated" forecasting vs. "direct" forecasting

  Ramey, p. 84: "[O]ne can forecast future values of a variable using either a horizon-specific regression ("direct" forecasting) or iterating on a one-period ahead estimated model ("iterated" forecasting)." Ramey suggests that calculating an IRF from a SVAR is analogous to iterated forecasting while using **local projection** à la Jordà (2005) is analogous to direct forecasting. As pointed out by Susanto as well, using local projection to compute IRFs is likely to yield more bumpy IRFs because the direct estimation of each horizon avoids the smoothness assumption embedded in an IRF coming from a SVAR.

→ One battle one could fight, which however I am reluctant to fight at this point, is to argue that the tradition of smooth, non-oscillatory IRFs in the data is wrong. Nonetheless I think it's worth keeping in mind. (And at least I can hint that I've at least looked at data at one point in my life.)

6 Changes to expectation formation

In order to dampen overshooting...

- Analysis in Section 2 suggests: learning should be about the slope as well, not just the constant

- Ball (1994) suggests: learning should work in a way to decrease the movement of expectations, not to increase it → can we get anchored "at the wrong place" so that expectations do not move?

- Or if expectations move just as much, we need to dampen the feedback from expectations.

What I've looked into:

- Bayesian learning (Gerko 2019, Collins-Dufresne et al 2016, Evans, Honkapohja & Sargent 2016)
  - I'm not sure if this is entirely correct, but I understand Bayesian learning to be the same as adaptive learning with the following two differences:
    1. Initialize using priors, and update using the Kalman filter instead of least-squares
    2. Can incorporate parameter uncertainty: can get around anticipated utility (although some confusion on whether this is simply an analogue of constant gain learning)
  - The hope here is to dampen overshooting. I see some chance for it for 2 reasons:
    1. because the Kalman gain gives us a model-specific way of choosing the value of the constant gain, dampening the movement in expectations
    2. because parameter uncertainty, if it allows me to get lower effects on future expectations than anticipated utility would, could dampen the interest rate feedback
  - A note: Evans et al make a strange side comment: when the feedback effect from expectations is negative and sufficiently strong (in the cobweb model $\alpha < -1$) "a possible problem of overshooting can emerge when agents overparametrize the PLM" - oh really?????

- Nonparametric learning (Kozlowski et al 2019)
  - In Bayesian and LS learning, agents are learning the matrix g_x which is known to be a linear, $ny \times (nx + 1)$ matrix.
  - In nonparametric learning, agents instead are learning the function g (in Koz et al, they are learning the distribution of shocks).
  - The recursion they use is a recursive kernel density estimator, giving $\hat{g}_t$ in period t.
  - So this is a frequentist approach; Orlik & Veldkamp 2014 do the same with a Bayesian recursion, imposing a trick to avoid the particle filter (since the estimated function f is potentially nonlinear).
  - Here I don't quite see how this could dampen feedback effects from expectations.

- My overall feeling: the only way to slow down learning is to decrease the gain, so what I need to do instead is to make agents know as little as possible so they can't infer things about the Taylor rule in the future

- Learn h_x too so you don't internalize how a shock will pan out
- Have them not know the Taylor rule (how to do that?)
- Put π_{t-1} instead of π_t in the Taylor rule, especially if agents don't know h_x.