Materials 12 - tinkering around with policy and expectation formation
An informational assumption question

Laura Gáti

January 14, 2020

Overview

1 Model summary **2**

2 Four quick changes **3**

2.1 Quick overview ◀ . . . 3

2.2 IRFs from vector learning: EE and LH, $T = 400, N = 100$ . . . 4

2.3 A technical note on the projection facility ◀ . . . 5

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

3 Bigger changes to expectation formation - so far only thoughts **5**

4 Changes to policy - a challenge for informational assumptions ◀ ◀ **7**

4.1 The big-picture issue . . . 7

4.2 How the issue arises in the extensions . . . 8

5 IRFs for baseline and 3 extensions, for 3 info assumptions **9**

5.1 Baseline and $\hat{\mathbb{E}}_t\pi_{t+1}$ in TR figures . . . 11

5.2 π_{t-1} in TR figures . . . 12

5.3 ρi_{t-1} in TR figures . . . 13

5.4 Baseline with agents NOT knowing the Taylor-rule . . . 14

1 Model summary

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

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

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

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

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

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

2 Four quick changes

2.1 Quick overview ◀

1. To policy
   (a) 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) Learning slope too, not just constant

What do they do:

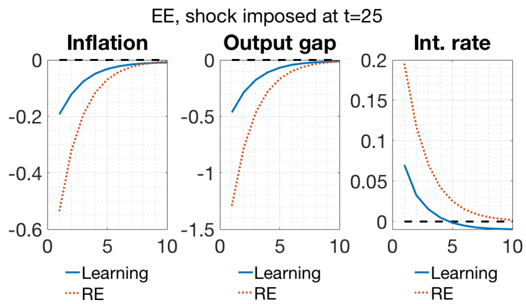
1. $\psi_\pi \leq 1$: indeed kills the overshooting, but - no surprise - makes observables unstable (IRFs don't return to steady state). Why does it work to kill the overshooting? B/c the Ball-effect of anticipated interest rate reactions no longer overweighs (less expectational feedback).
2. Townsend (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!)
3. 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.)
4. 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).

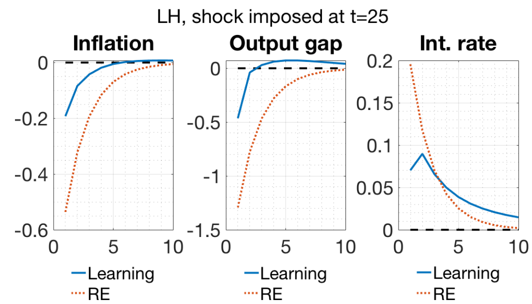
   2) More bumpy because since you're learning b, the loading on shocks, the specific sequence of shocks matters. Increasing the size of the cross-section, N, mitigates this somewhat.

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

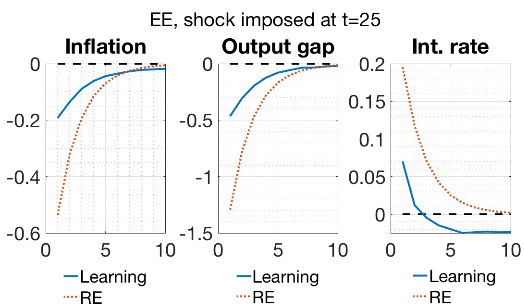
Figure 1: Learning constant only



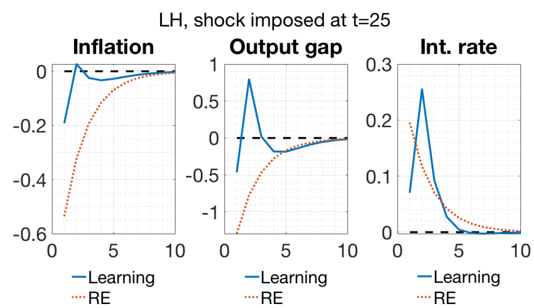
(a) EE, dgain



(b) LH, dgain

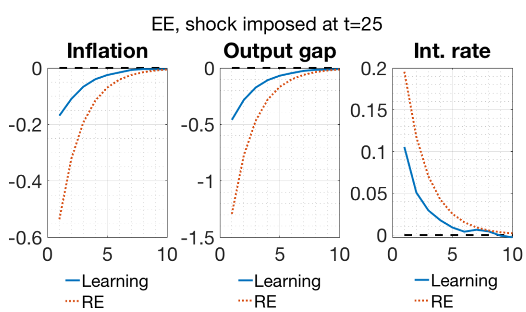


(c) EE, cgain

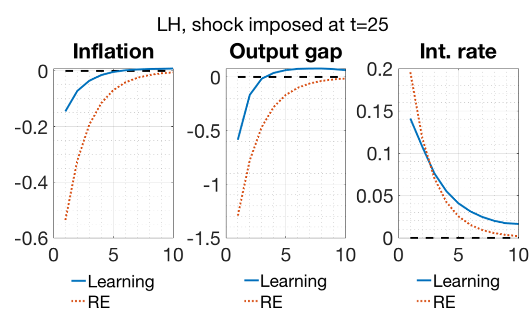


(d) LH, cgain

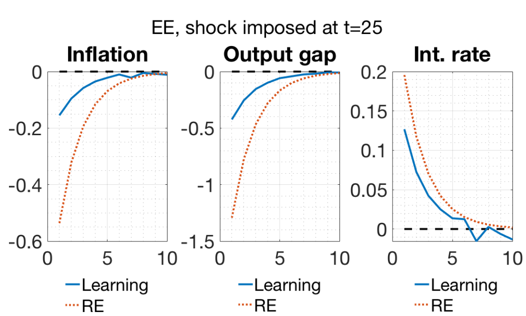
Figure 2: Learning slope and constant



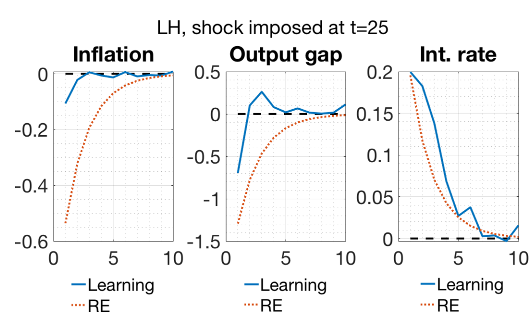
(a) EE, dgain



(b) LH, dgain



(c) EE, cgain



(d) LH, cgain

2.3 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?

2.4 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.)

3 Bigger changes to expectation formation - so far only thoughts

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.

4 Changes to policy - a challenge for informational assumptions ◀ ◀

4.1 The big-picture issue

The three equations of the model reproduced:

$$
\begin{aligned}
x_t &= -\sigma i_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} \beta^{T-t}\big((1-\beta)x_{T+1} - \sigma(\beta i_{T+1} - \pi_{T+1}) + \sigma r_T^n\big) \\
\pi_t &= \kappa x_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} (\alpha\beta)^{T-t}\big(\kappa\alpha\beta x_{T+1} + (1-\alpha)\beta\pi_{T+1} + u_T\big) \\
i_t &= \psi_\pi \pi_t + \psi_x x_t + \bar{i}_t
\end{aligned}
$$

Three extensions to the policy rule of the baseline model:

- $\hat{\mathbb{E}}_t \pi_{t+1}$ in TR instead of π_t ("expected-inflation")
- π_{t-1} in TR instead of π_t ("lagged inflation")
- Add ρi_{t-1} to TR ("interest rate smoothing")

It will turn out that these extensions highlight some subtle informational assumptions that will all be a variation on the following big-picture informational question:

The solution of the model takes the state-space form:

$$X_t = h_x X_{t-1} + \eta \varepsilon_t \tag{7}$$

$$Y_t = g_x X_t \tag{8}$$

Agents know h_x and estimate $\hat{g}_x$. In terms of the model equations NKIS, NKPC and TR, what does this informational assumption imply? In particular, do agents know the relationships between the jumps, that is, do they know the three model equations, and are they thus able to impute

$$Y_t = A_a f_a + A_b f_b + A_s X_t ? \tag{9}$$

I tend to think that they do *not* know the NKIS and NKPC and thus cannot impute (9) b/c in that case they would internalize that they are identical, which we've assumed away. But they can still know the TR, in fact I think they do know the TR b/c the CB might announce it publicly.

4.2 How the issue arises in the extensions

The question reemerges when the extension involves introducing an endogenous state. (So the $\hat{\mathbb{E}}_t\pi_{t+1}$-extension is not problematic in this regard. That extension just requires rewriting the compact notation as $Y_t = A_a f_a + A_b f_b + A_s X_t + A_e \hat{\mathbb{E}}_t \pi_{t+1}$, where $A_e = \psi_\pi$, and the learning code needs to be altered to take this into account, but otherwise no drastic changes are required.) When you introduce an endogenous state, however, like lagged inflation or lagged interest rate, then the problem emerges.

Let me illustrate on the example of interest rate smoothing. Model equations are:

$$
\begin{aligned}
x_t &= -\sigma i_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} \beta^{T-t}\big((1-\beta)x_{T+1} - \sigma(\beta i_{T+1} - \pi_{T+1}) + \sigma r^n_T\big) \\
\pi_t &= \kappa x_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} (\alpha\beta)^{T-t}\big(\kappa\alpha\beta x_{T+1} + (1-\alpha)\beta\pi_{T+1} + u_T\big) \\
i_t &= \psi_\pi \pi_t + \psi_x x_t + \bar{i}_t + \rho i l_t \\
il_{t+1} &= i_t
\end{aligned}
$$

The big-picture question arises in the following particular form: do agents internalize the last, linking equation between i and il ? The reason this becomes important is because it has consequences for how they forecast. 2 options:

1. No: they treat i and il as two separate variables and use h_x (which they know) to forecast il.
   - Advantage: Easy to solve because this means that agents do not realize that `gx(end,:)  = hx(end,:)`.
   - Disadvantage: Unrealistic and also has undesirable implications for dynamics b/c now you've just added another exogenous shock to the Taylor-rule that happens to have the same law of motion as i; no wonder I found that it didn't make a difference! (This is the approach I have taken so far from materials6 onwards for interest rate smoothing w/o realizing it.)
2. Yes: Now you're in for a curious modeling difficulty: What should agents now use to forecast il and in particular i?
   - If they use h_x for il and g_x for i they are inconsistent b/c they know that the variables are truly equal.
   - If they use g_x for both they are forecasting suboptimally because they are not using all the information available to them. (And unfortunately, materials12f all do this!)

- If they use h_x for both then `gx(end,:)` is revealed! This is however only a problem if they are learning both constant *and slope* b/c if they are only learning the constant, then `gx(2:end,:)` is known anyway. But I just concluded in above sections that it's desirable for them to learn the slope too because it dampens shock impacts.

5 IRFs for baseline and 3 extensions, for 3 info assumptions

Recap of model versions

1. Baseline
2. $\hat{\mathbb{E}}_t \pi_{t+1}$ in TR instead of π_t ("expected-inflation")
3. π_{t-1} in TR instead of π_t ("lagged inflation")
4. Add ρi_{t-1} to TR ("interest rate smoothing")

Recap of informational assumptions:

1. Forecast jump using g_x, endogenous state using h_x ("myopic forecasters")
2. Forecast both using g_x ("suboptimal forecasters")
3. Forecast both using h_x ("optimal forecasters")

Recap of which info assumption makes a difference for which extension

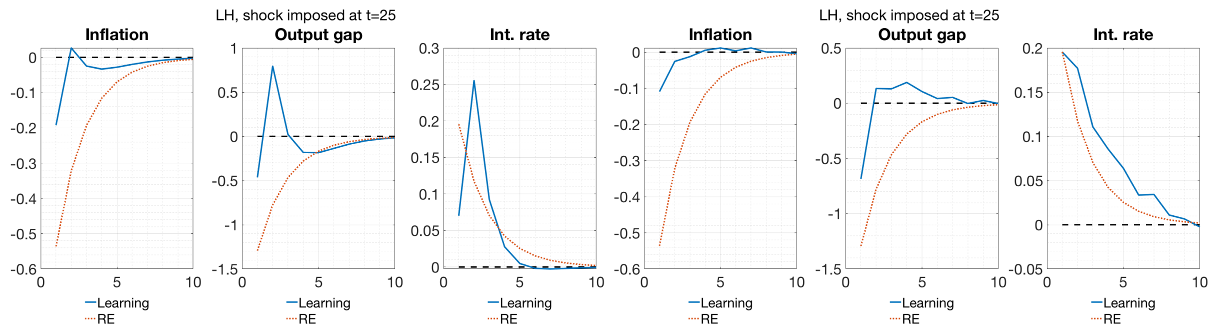
1. Baseline $\rightarrow$ same for all
2. $\hat{\mathbb{E}}_t \pi_{t+1}$ in TR instead of π_t ("expected-inflation") $\rightarrow$ same for all
3. π_{t-1} in TR instead of π_t ("lagged inflation") $\rightarrow$ info assumptions matter, have implemented only (2) and (3)
4. Add ρi_{t-1} to TR ("interest rate smoothing") $\rightarrow$ info assumptions matter, have implemented all

Overview of plots - all plots LH, constant gain vector learning, monetary policy shock imposed at $t = 25$.

1. Baseline
   (a) Learning constant only
   (b) Learning slope and constant
2. $\hat{\mathbb{E}}_t \pi_{t+1}$ in TR instead of π_t ("expected-inflation")
   (a) Learning constant only
   (b) Learning slope and constant
3. π_{t-1} in TR instead of π_t ("lagged inflation")
   (a) Forecast both using g_x ("suboptimal forecasters")
      i. Learning constant only
      ii. Learning slope and constant
   (b) Forecast both using h_x ("optimal forecasters")
      i. Learning constant only
      ii. Learning slope and constant
4. ρi_{t-1} in TR ("interest rate smoothing") with $\rho = 0.6$
   (a) Forecast jump using g_x, endogenous state using h_x ("myopic forecasters")
      i. Learning constant only
      ii. Learning slope and constant
   (b) Forecast both using g_x ("suboptimal forecasters")
      i. Learning constant only
      ii. Learning slope and constant
   (c) Forecast both using h_x ("optimal forecasters")
      i. Learning constant only
      ii. Learning slope and constant
5. Baseline with agents not knowing the Taylor-rule

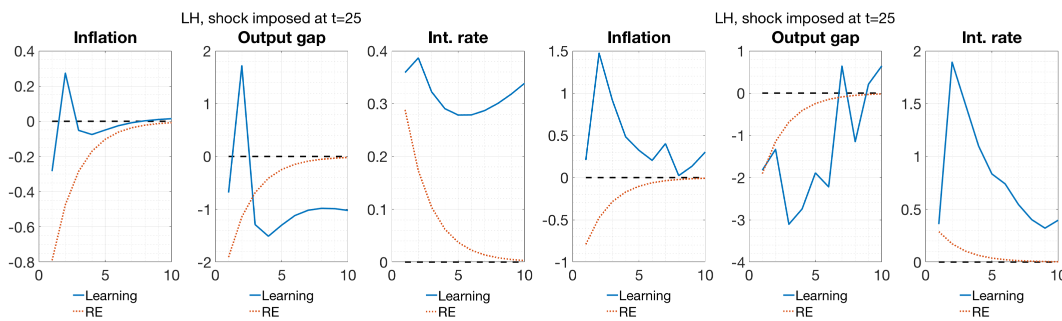
5.1 Baseline and $\hat{\mathbb{E}}_t\pi_{t+1}$ in TR figures

Figure 3: Baseline



(a) Learning constant only (b) Learning slope and constant

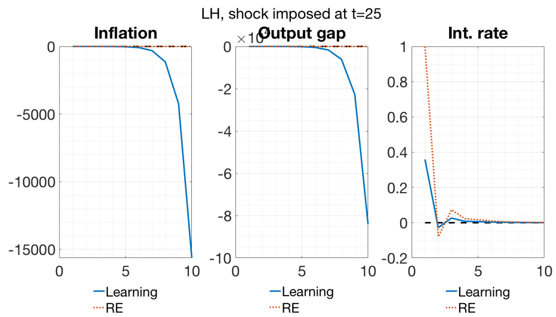
Figure 4: $\hat{\mathbb{E}}_t\pi_{t+1}$ in TR



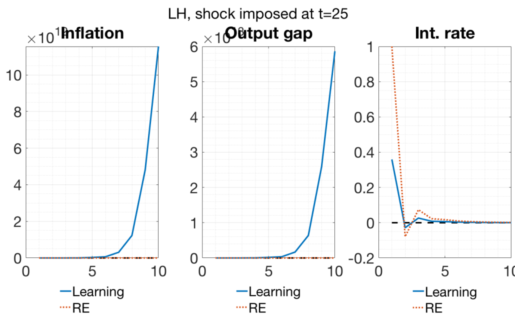
(a) Learning constant only (b) Learning slope and constant

5.2 π_{t-1} in TR figures

Figure 5: π_{t-1} in TR, forecasting using g_x for jump, h_x for state

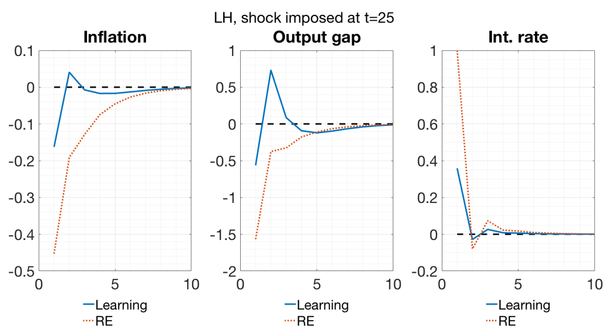


(a) Learning constant only

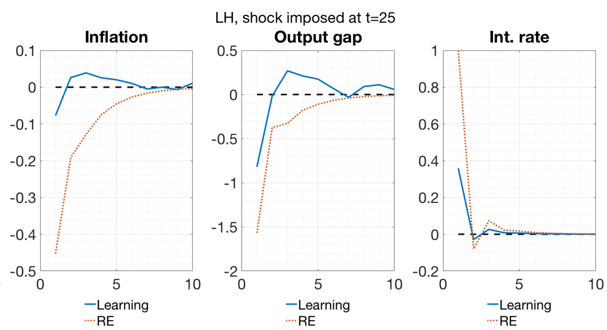


(b) Learning slope and constant

Figure 6: π_{t-1} in TR, forecasting using g_x for both

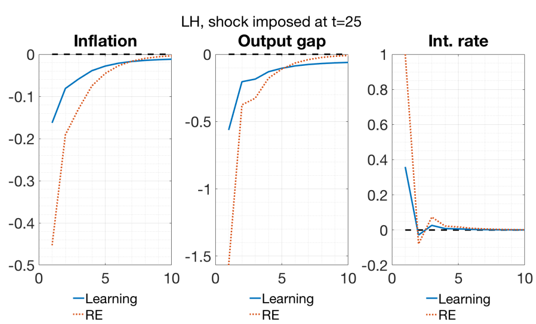


(a) Learning constant only

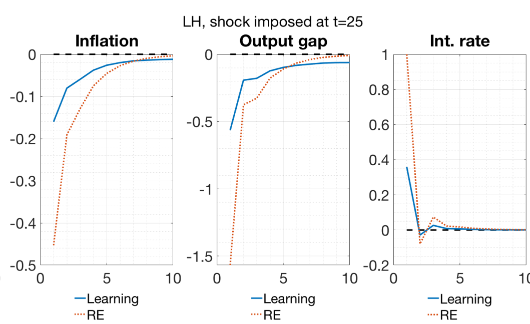


(b) Learning slope and constant

Figure 7: π_{t-1} in TR, forecasting using h_x for both



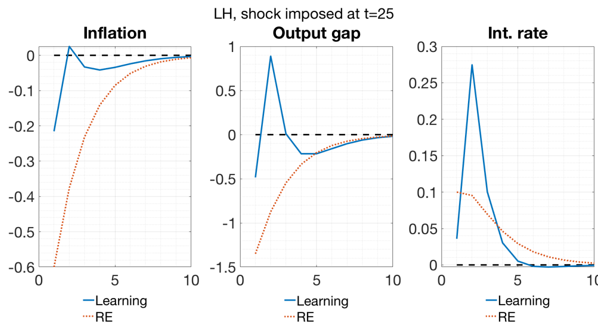
(a) Learning constant only



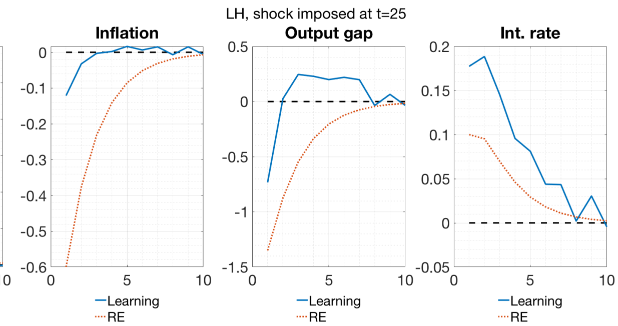
(b) Learning slope and constant

5.3 ρi_{t-1} in TR figures

Figure 8: ρi_{t-1} in TR, forecasting using g_x for jump, h_x for state

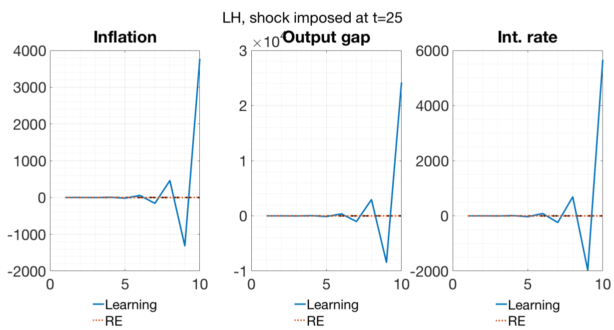


(a) Learning constant only

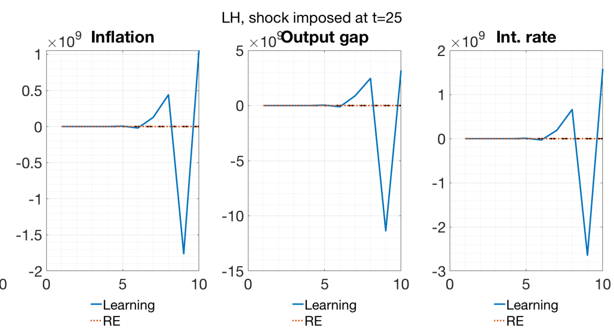


(b) Learning slope and constant

Figure 9: ρi_{t-1} in TR, forecasting using g_x for both

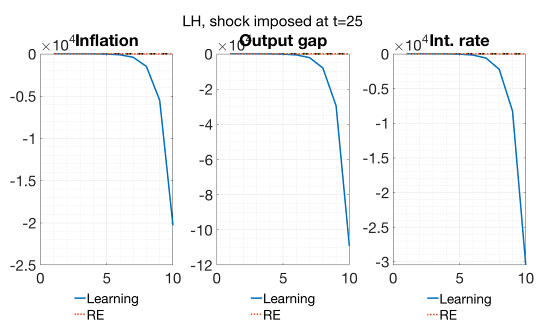


(a) Learning constant only

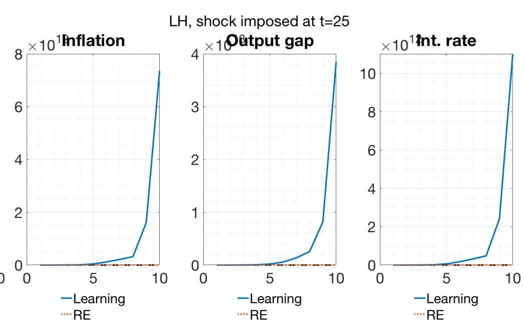


(b) Learning slope and constant

Figure 10: ρi_{t-1} in TR, forecasting using h_x for both



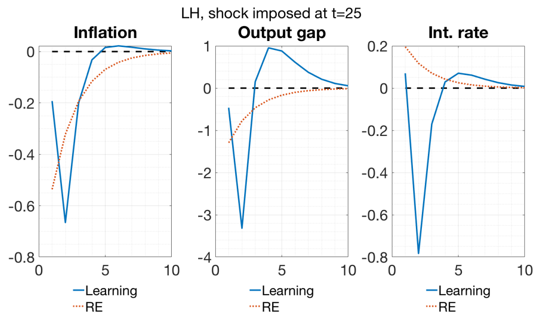
(a) Learning constant only



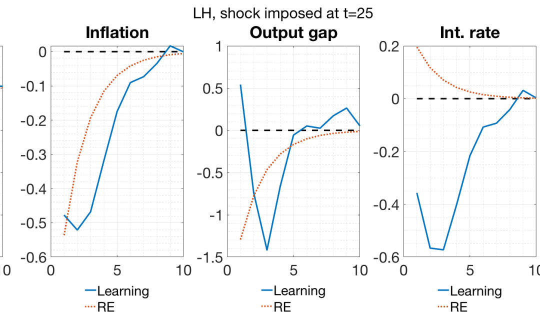
(b) Learning slope and constant

5.4 Baseline with agents NOT knowing the Taylor-rule

Figure 11: Baseline, agents don't know TR



(a) Learning constant only



(b) Learning slope and constant