# Materials 3 - Special cases

Laura Gáti

September 17, 2019

## Overview

**1 The models to be simulated** 2

1.1 RE . . . 2

1.2 EE . . . 2

1.3 LR . . . 3

**2 Compact notation** 3

**3 Learning** 5

**4 ALMs** 7

4.1 RE . . . 7

4.2 EE . . . 7

4.3 LR . . . 7

**5 Timeline in the learning models** 8

**6 Special cases towards general case: procedure** 8

**7 Simulations** 9

7.1 Learning slope and constant . . . 9

7.2 Learning constant only . . . 13

# 1 The models to be simulated

1. Rational expectations NK model (RE)
2. Euler equation approach learning NK model à la Bullard & Mitra (2002) (EE)
3. LR expectations learning NK model à la Preston (2005) (LR)
4. (Eventually: LR expectations learning NK model à la Preston with anchoring à la CEMP)

The difference between these models is 1) in the expectations (rational or not), 2) in the number of horizons of expectations that need to be summed (1 vs. infinite). So what I'm going to do consists of 2 steps:

1. Write a learning rule that takes the form of Preston's, but that nests CEMP, and has a decreasing gain.
2. Write out $f_a$ and $f_b$ as truncated sums of $h$-period ahead forecasts. When $h = 1$, EE and LR (models (6) and (7)) should coincide.

## 1.1 RE

$$x_t = \mathbb{E}_t\, x_{t+1} - \sigma(i_t - \mathbb{E}_t\, \pi_{t+1}) + \sigma r_t^n \tag{1}$$

$$\pi_t = \kappa x_t + \beta\, \mathbb{E}_t\, \pi_{t+1} + u_t \tag{2}$$

$$i_t = \bar{i}_t + \psi_\pi \pi_t + \psi_x x_t \tag{3}$$

## 1.2 EE

$$x_t = \hat{\mathbb{E}}_t x_{t+1} - \sigma(i_t - \hat{\mathbb{E}}_t \pi_{t+1}) + \sigma r_t^n \qquad \text{(Preston, eq. (13))}$$

$$\pi_t = \kappa x_t + \beta \hat{\mathbb{E}}_t \pi_{t+1} + u_t \qquad \text{(Preston, eq. (14))}$$

$$i_t = \bar{i}_t + \psi_\pi \pi_t + \psi_x x_t \qquad \text{(Preston, eq. (27))}$$

### 1.3 LR

$$x_t = -\sigma i_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} \beta^{T-t}\big((1-\beta)x_{T+1} - \sigma(\beta i_{T+1} - \pi_{T+1}) + \sigma r_T^n\big) \quad \text{(Preston, eq. (18))}$$

$$\pi_t = \kappa x_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} (\alpha\beta)^{T-t}\big(\kappa\alpha\beta x_{T+1} + (1-\alpha)\beta\pi_{T+1} + u_T\big) \quad \text{(Preston, eq. (19))}$$

$$i_t = \psi_\pi \pi_t + \psi_x x_t + \bar{i}_t \quad \text{(Preston, eq. (27))}$$

One issue is that if I set $T = t$, I don't think Preston, eq. (18) reduces to Preston, eq. (13), nor does Preston, eq. (19) reduce to Preston, eq. (14).

## 2 Compact notation

Exogenous states are summarized as:

$$s_t = Ps_{t-1} + \epsilon_t \quad \text{where} \quad s_t \equiv \begin{pmatrix} r_t^n \\ \bar{i}_t \\ u_t \end{pmatrix} \quad P \equiv \begin{pmatrix} \rho_r & 0 & 0 \\ 0 & \rho_i & 0 \\ 0 & 0 & \rho_u \end{pmatrix} \quad \epsilon_t \equiv \begin{pmatrix} \varepsilon_t^r \\ \varepsilon_t^i \\ \varepsilon_t^u \end{pmatrix} \quad \text{and} \quad \Sigma = \begin{pmatrix} \sigma_r & 0 & 0 \\ 0 & \sigma_i & 0 \\ 0 & 0 & \sigma_u \end{pmatrix}$$

Let $z_t$ summarize the endogenous variables as

$$z_t \equiv \begin{pmatrix} \pi_t \\ x_t \\ i_t \end{pmatrix} \tag{4}$$

Then I can write the models compactly as

$$z_t = A_p^{RE}\, \mathbb{E}_t\, z_{t+1} + A_s^{RE} s_t \tag{5}$$

$$z_t = A_p^{RE} \hat{\mathbb{E}}_t z_{t+1} + A_s^{RE} s_t \tag{6}$$

$$z_t = A_a^{LR} f_a + A_b^{LR} f_b + A_s^{LR} s_t \tag{7}$$

$$s_t = Ps_{t-1} + \epsilon_t \tag{8}$$

where $f_a$ and $f_b$ capture discounted sums of expectations at all horizons of the endogenous states $z$ (following Preston, I refer to these objects as "long-run expectations"):

$$f_a \equiv \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} (\alpha\beta)^{T-t} z_{T+1} \qquad f_b \equiv \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} (\beta)^{T-t} z_{T+1} \tag{9}$$

and the coefficient matrices are given by:

$$A_p^{RE} = \begin{pmatrix} \beta + \frac{\kappa\sigma}{w}(1-\psi_\pi\beta) & \frac{\kappa}{w} & 0 \\ \frac{\sigma}{w}(1-\psi_\pi\beta) & \frac{1}{w} & 0 \\ \psi_\pi\big(\beta + \frac{\kappa\sigma}{w}(1-\psi_\pi\beta)\big) + \psi_x\frac{\sigma}{w}(1-\psi_\pi\beta) & \psi_x(\frac{1}{w}) + \psi_\pi(\frac{\kappa}{w}) & 0 \end{pmatrix} \tag{10}$$

$$A_s^{RE} = \begin{pmatrix} \frac{\kappa\sigma}{w} & -\frac{\kappa\sigma}{w} & 1-\frac{\kappa\sigma\psi_\pi}{w} \\ \frac{\sigma}{w} & -\frac{\sigma}{w} & -\frac{\sigma\psi_\pi}{w} \\ \psi_x(\frac{\sigma}{w}) + \psi_\pi(\frac{\kappa\sigma}{w}) & \psi_x(-\frac{\sigma}{w}) + \psi_\pi(-\frac{\kappa\sigma}{w}) + 1 & \psi_x(-\frac{\sigma\psi_\pi}{w}) + \psi_\pi(1-\frac{\kappa\sigma\psi_\pi}{w}) \end{pmatrix} \tag{11}$$

$$A_a^{LR} = \begin{pmatrix} g_{\pi a} \\ g_{xa} \\ \psi_\pi g_{\pi a} + \psi_x g_{xa} \end{pmatrix} \quad A_b^{LR} = \begin{pmatrix} g_{\pi b} \\ g_{xb} \\ \psi_\pi g_{\pi b} + \psi_x g_{xb} \end{pmatrix} \quad A_s^{LR} = \begin{pmatrix} g_{\pi s} \\ g_{xs} \\ \psi_\pi g_{\pi s} + \psi_x g_{xs} + \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \end{pmatrix} \tag{12}$$

$$g_{\pi a} = (1 - \frac{\kappa\sigma\psi_\pi}{w})\left[(1-\alpha)\beta, \kappa\alpha\beta, 0\right] \tag{13}$$

$$g_{xa} = \frac{-\sigma\psi_\pi}{w}\left[(1-\alpha)\beta, \kappa\alpha\beta, 0\right] \tag{14}$$

$$g_{\pi b} = \frac{\kappa}{w}\left[\sigma(1-\beta\psi_\pi), (1-\beta-\beta\sigma\psi_x, 0\right] \tag{15}$$

$$g_{xb} = \frac{1}{w}\left[\sigma(1-\beta\psi_\pi), (1-\beta-\beta\sigma\psi_x, 0\right] \tag{16}$$

$$g_{\pi s} = (1 - \frac{\kappa\sigma\psi_\pi}{w})\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}(I_3 - \alpha\beta P)^{-1} - \frac{\kappa\sigma}{w}\begin{bmatrix} -1 & 1 & 0 \end{bmatrix}(I_3 - \beta P)^{-1} \tag{17}$$

$$g_{xs} = \frac{-\sigma\psi_\pi}{w}\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}(I_3 - \alpha\beta P)^{-1} - \frac{\sigma}{w}\begin{bmatrix} -1 & 1 & 0 \end{bmatrix}(I_3 - \beta P)^{-1} \tag{18}$$

$$w = 1 + \sigma\psi_x + \kappa\sigma\psi_\pi \tag{19}$$

# 3 Learning

In Preston (2005), agents forecast the endogenous variables using the exogenous ones as

$$z_t = a_t + b_t s_t + \epsilon_t \tag{Preston, p. 101}$$

which I suspect isn't precise about the timing. Therefore, I write a general PLM of the form

$$z_t = a_{t-2} + b_{t-2} s_{t-1} + \epsilon_t \tag{20}$$

and then $\phi_{t-2} = (a_{t-2}, b_{t-2})$, here $3 \times 4$, so that agents learn both a constant and a slope term. This means $\hat{\mathbb{E}}_t z_{t+1} = \phi_{t-1} \begin{bmatrix} 1 \\ s_t \end{bmatrix}$. Later, I will simplify here so that agents only learn about the constant, i.e. about CEMP's drift term, but forecast exogenous states rationally:

$$z_t = \bar{z}_{t-2} + s_{t-1} + \epsilon_t \tag{21}$$

so that $\phi_{t-2} = \bar{z}_{t-2}$. I'm actually quite worried about the assumption that agents only learn about the constant because it seems like a permanent deviation from RE: might it screw up E-stability?

Anticipated utility implies that

$$\hat{\mathbb{E}}_{t-1} \phi_{t+h} = \hat{\mathbb{E}}_{t-1} \phi_t \equiv \phi_{t-1} \quad \forall\ h \geq 0 \tag{22}$$

This is a little tricky. It doesn't only mean that agents today mistakenly believe that they will not update the forecasting rule. It also implies that the belief about $\phi_t$ was formed at $t-1$. Assuming RE about the exogenous process and anticipated utility, $h$-horizon forecasts are constructed as:

$$\hat{\mathbb{E}}_t z_{t+h} = a_{t-1} + b_{t-1} P^{h-1} s_t \quad \forall h \geq 1 \tag{23}$$

Or, if I assume that agents don't learn the slope,

$$\hat{\mathbb{E}}_t z_{t+h} = \bar{z}_{t-1} + P^{h-1} s_t \quad \forall h \geq 1 \tag{24}$$

and the regression coefficients are updated using (for now) a decreasing gain RLS algorithm:

$$\phi_t = \left(\phi'_{t-1} + t^{-1}\mathbf{R_t^{-1}}\begin{bmatrix}\mathbf{1}\\ \mathbf{s_{t-1}}\end{bmatrix}\left(z_t - \phi_{t-1}\begin{bmatrix}1\\ s_{t-1}\end{bmatrix}\right)'\right)' \tag{25}$$

$$R_t = R_{t-1} + t^{-1}\left(\begin{bmatrix}1\\ s_{t-1}\end{bmatrix}\begin{bmatrix}1 & s_{t-1}\end{bmatrix} - R_{t-1}\right) \tag{26}$$

$R_t$ is $4 \times 4$ and $\phi_t$ is $3 \times 4$. Three questions:

1. The forecast error $z_t - \phi_{t-1}\begin{bmatrix}1\\ s_{t-1}\end{bmatrix}$ has weird timing: I don't think agents ever carried out this forecast, because at time $t-1$, their forecast was $\hat{\mathbb{E}}_{t-1}z_t = \phi_{t-2}\begin{bmatrix}1\\ s_{t-1}\end{bmatrix}$. So the fcst $\phi_{t-1}\begin{bmatrix}1\\ s_{t-1}\end{bmatrix}$ seems to be $\hat{\mathbb{E}}_t z_t$, i.e. a forecast that they are just using to assess $\phi$, but never actually relied upon previously (let's refer to this forecast as the "assessment forecast"; really it's actually a nowcast). Or?

2. The bold $\mathbf{R_t^{-1}}\begin{bmatrix}\mathbf{1}\\ \mathbf{s_{t-1}}\end{bmatrix}$ indicates a difference to CEMP's learning algorithm: these terms are missing in CEMP. Am I right in thinking that that's because in CEMP, agents only learn the constant, and so the data they use is 1 instead of $\begin{bmatrix}1\\ s_t\end{bmatrix}$, making $R_t = 1 \; \forall \; t$?

3. Can this formulation capture the special case that agents only learn about the constant? $\Leftrightarrow$ Following up on the previous point, it seems to me that when agents learn only the constant, then $\phi_t = \bar{z}_{t-1}$ and the learning algorithm boils down to

$$\bar{z}_t = \bar{z}_{t-1} + t^{-1} \underbrace{\left(z_t - (\bar{z}_{t-1} + s_{t-1})\right)}_{\text{fcst error given assessment fcst using (24)}} \tag{27}$$

And a note: CEMP is a special case of this model, with the gain switching between decreasing and constant according to the anchoring mechanism. I'm leaving that out for the time being.

# 4 ALMs

## 4.1 RE

With some abuse of terminology, call the state-space representation the ALM of the RE model:

$$x_t = hx\; x_{t-1} + \eta s_t \tag{28}$$

$$z_t = gx\; x_t \tag{29}$$

Then I can write the "ALM" as

$$z_t = gx\; hx\; x_{t-1} + gx\; \eta s_t \tag{30}$$

Since this ALM implies no constant, I initialize $\bar{z}_0$ as a $3 \times 1$ zero vector, and thus $\phi_0 = \begin{bmatrix} \bar{z}_0 & gx\; hx \end{bmatrix}$ (and $hx = P$ for the NK model). Analogously, I initialize $R$ as a $R_0 = \begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \Sigma_x \end{bmatrix}$, where $\Sigma_x$ is the VC matrix of the states from the RE solution. For the case where agents only learn the constant, I still initialize $\bar{z}_0$ as a $3 \times 1$ zero vector (and $R$ drops).

## 4.2 EE

I just need to use (23) to evaluate one-period ahead forecasts (for constant-learning only, (24)), and plug those into (6).

## 4.3 LR

Evaluate analytical "LR expectations" (9) using the PLM (23),

$$f_a = \frac{1}{1-\alpha\beta} a_{t-1} + b_{t-1}(I_3 - \alpha\beta P)^{-1} s_t \qquad f_b = \frac{1}{1-\beta} a_{t-1} + b_{t-1}(I_3 - \beta P)^{-1} s_t \tag{31}$$

and plug them into (7). In the case where agents only learn the constant I use (24):

$$f_a = \frac{1}{1-\alpha\beta} \bar{z}_{t-1} + (I_3 - \alpha\beta P)^{-1} s_t \qquad f_b = \frac{1}{1-\beta} \bar{z}_{t-1} + (I_3 - \beta P)^{-1} s_t \tag{32}$$

Alternatively I can evaluate each $h$-period ahead forecast individually using (23), and then sum $H$ of these terms, discounting appropriately. Earlier, it seemed that already a $H = 100$ is not a bad approximation of $\infty$-horizons, but now that only holds for $f_a$. For $f_b$ to be accurate, I need at least $H = 10000$. Why? Does the fact that $\alpha\beta < \beta$ matter so much?

# 5 Timeline in the learning models

$\underline{t=0}$: Initialize $\phi_{t-1} = \phi_0$ at the RE solution.

For each $t$:

1. Evaluate expectations $t+s$ (the one-period ahead, $(s=1)$ or the full 1 to $\infty$-period ahead $(s=1,\ldots,\infty)$) given $\phi_{t-1}$ and states dated $t$
2. Evaluate ALM given expectations: "today's observables are a function of expectations and today's state"
3. Update learning: $\phi_t =$ RLS of $\phi_{t-1}$ and fcst error between today's data and yesterday's forecast

# 6 Special cases towards general case: procedure

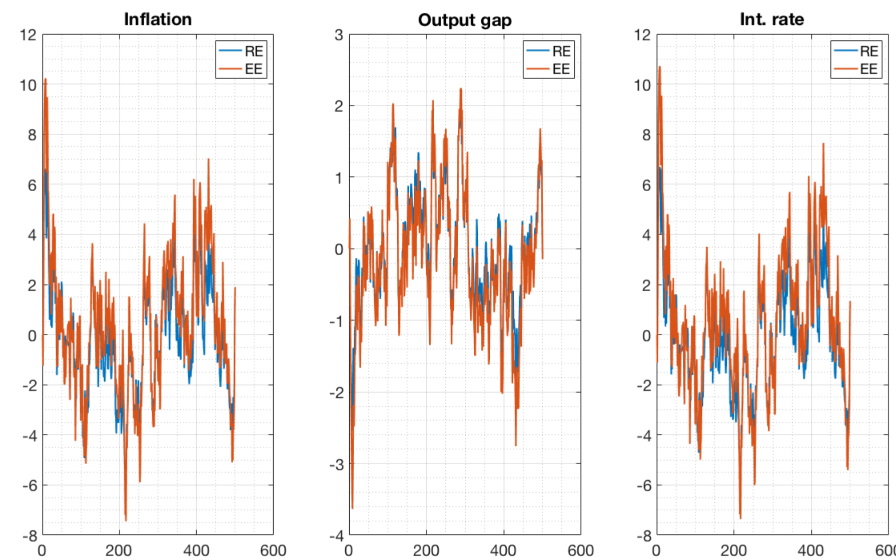
1. Simulate RE model ✓
2. Simulate EE model where agents learn both slope and constant ✓
   - Simulate using the "implicit ALM": rearranging the expectational matrix equation that underlies the solution to the model, you obtain the simulated observables $z_t$ without explicitly writing out the ALM ✓
   - Simulate using the "explicit ALM", equation (6), plugging in expectations evaluated separately. ✓

   The cool thing is: when I do the above two steps, I obtain the same simulated observables, so I know I'm doing it correctly.
3. Simulate LR model where agents learn both slope and constant, extend horizons from 1 to $\infty$ ✓
4. Simulate EE model where agents learn only the constant ✓
5. Simulate LR model where agents learn only the constant, extend horizons from 1 to infinity ✓

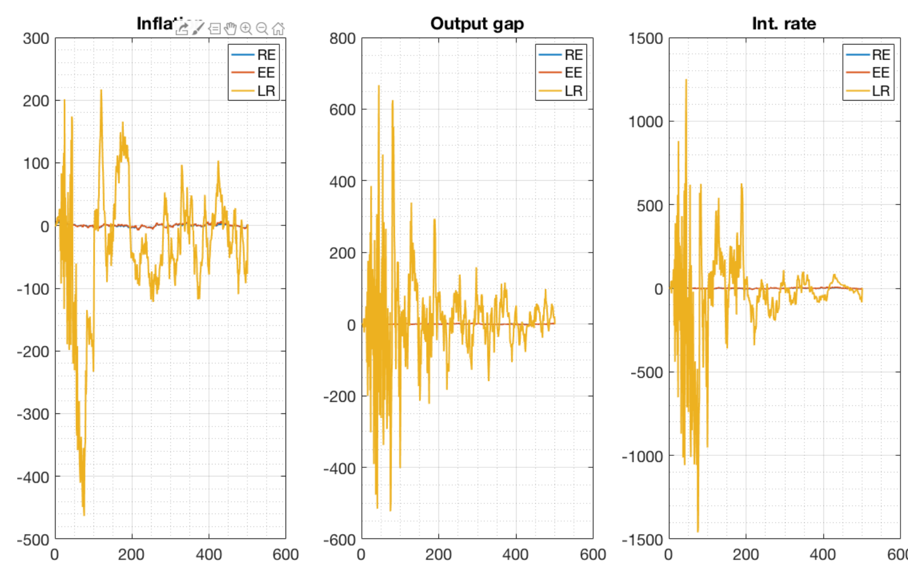
# 7 Simulations

## 7.1 Learning slope and constant

**Figure 1:** Comparing models

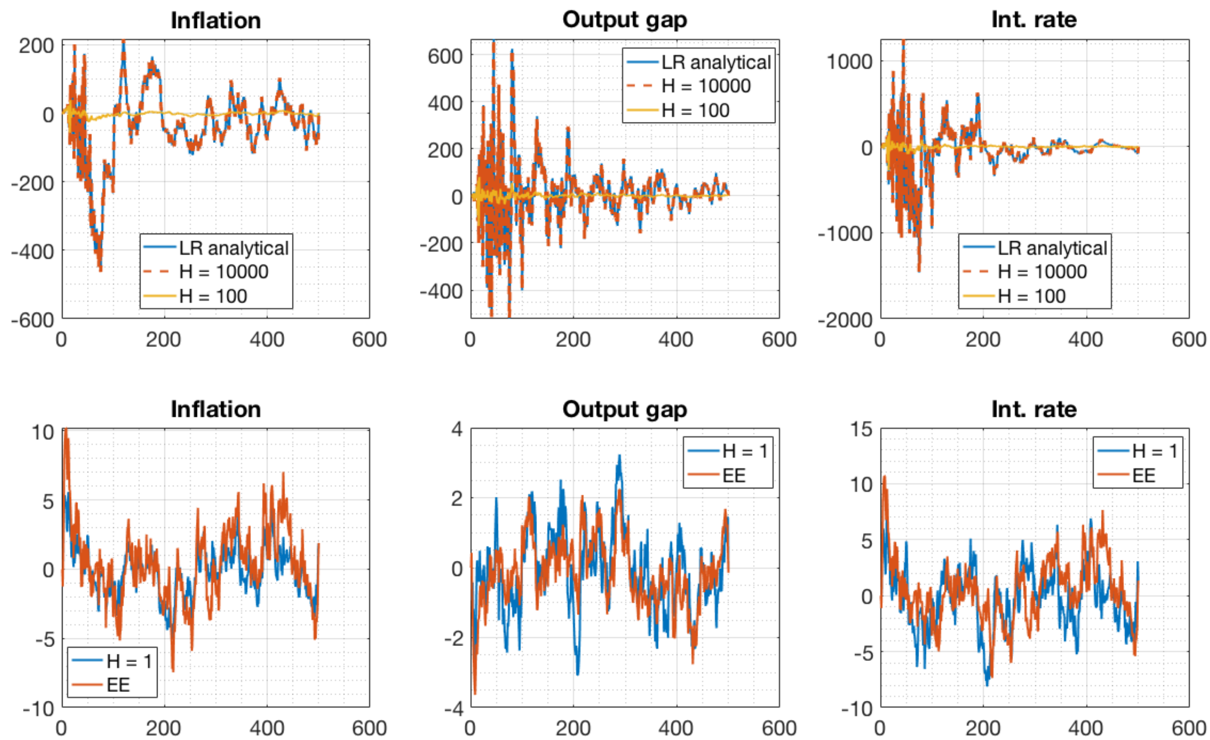



(a) RE and EE models only, learning both slope and constant




(b) RE, EE and LR models, learning both slope and constant, analytical expectations

**Figure 2:** Comparing horizons




Takeaways:

1. EE learning converges to RE over time, confirming that it's correct. Does LR? It doesn't seem like it (at $T = 100000$, it hasn't converged).

2. LR analytical and truncated expectations coincide for a large enough horizon ($H \sim 10000$)

3. Even for $H = 1$, LR and EE don't coincide; I think this is because the equations do not map onto one another.

So is it the case that learning isn't converging in the LR model?

→ No! It's clearly converging, albeit slowly, see next fig!

**Figure 3:** Convergence LR learning

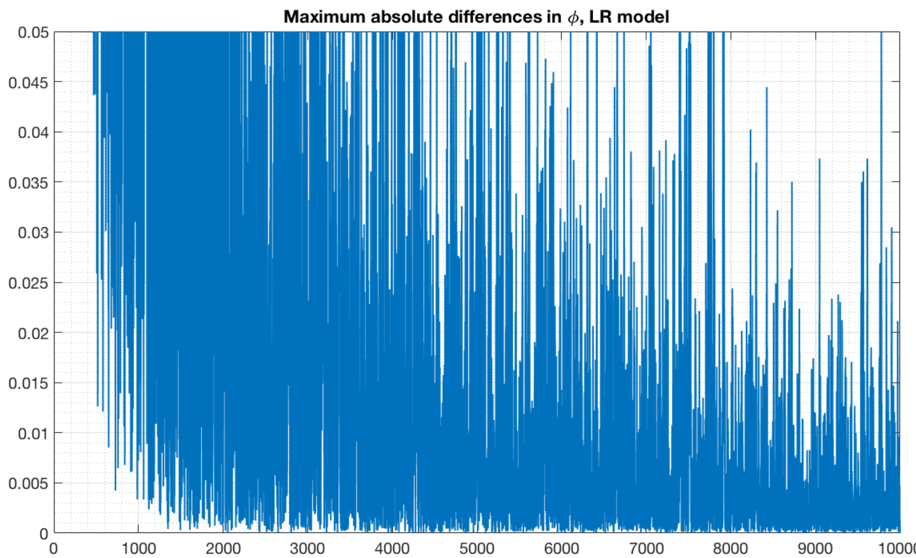
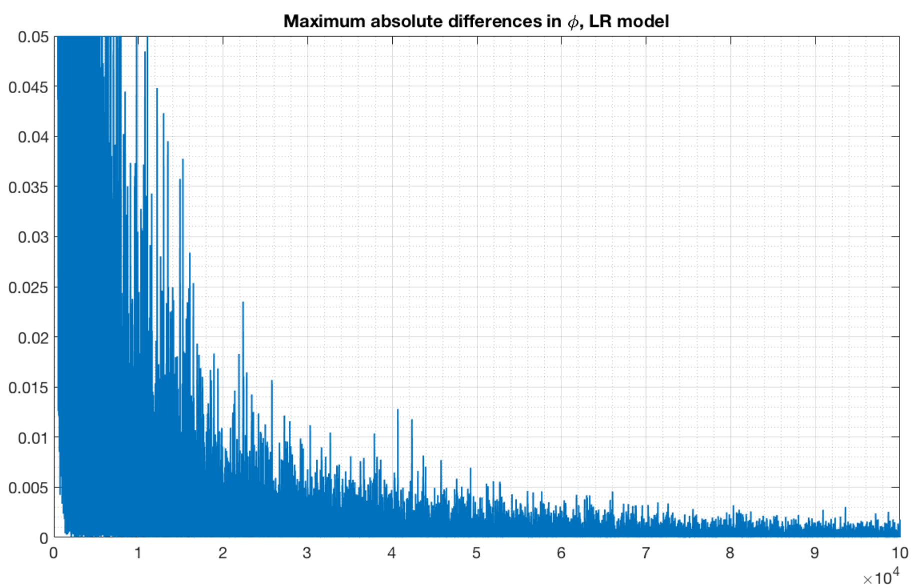


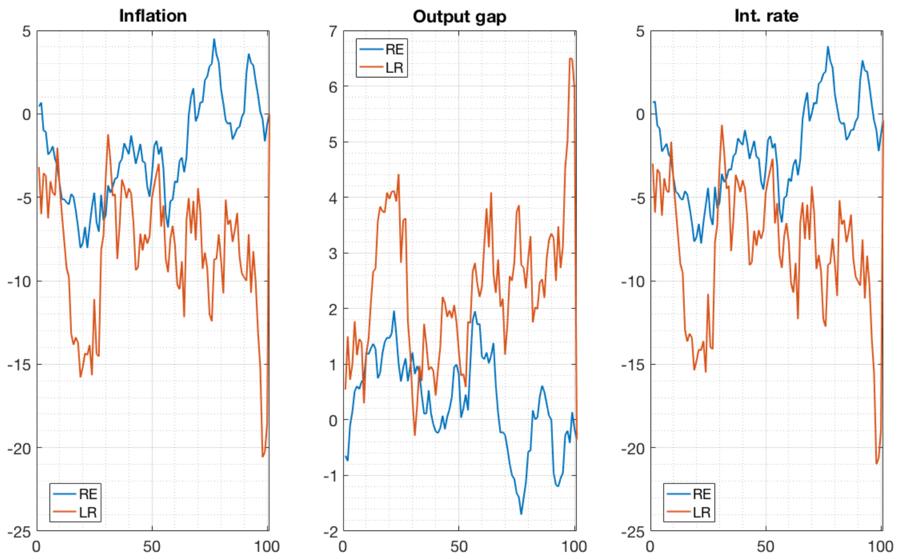

**Figure 4:** Convergence LR learning, $T = 100000$




See: clearly converging!

... but, problematically...

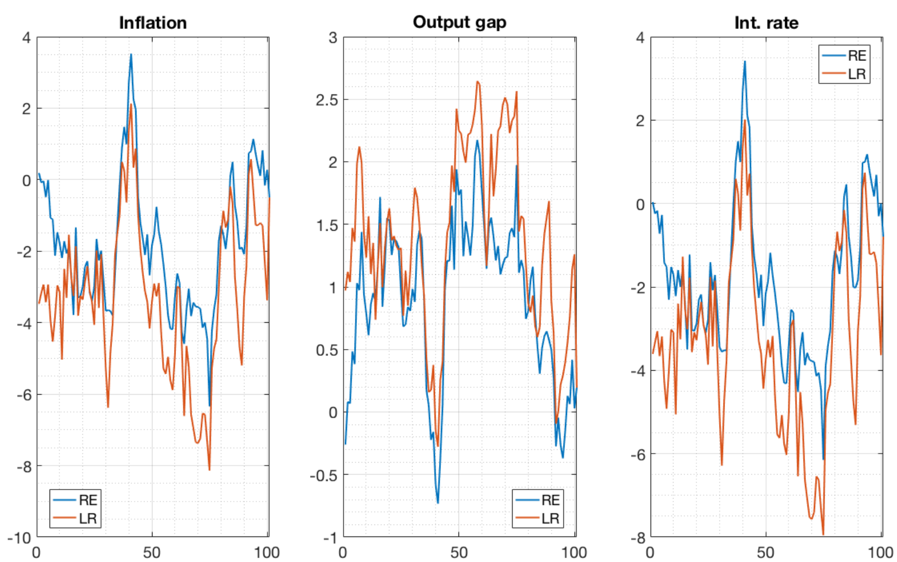
**Figure 5:** RE and LR models, $T = 100000$, last 100 periods




... the LR model observables clearly aren't converging to the RE model, not even after 100,000 simulated periods!

But wait a second, look at what they're doing after a million periods...

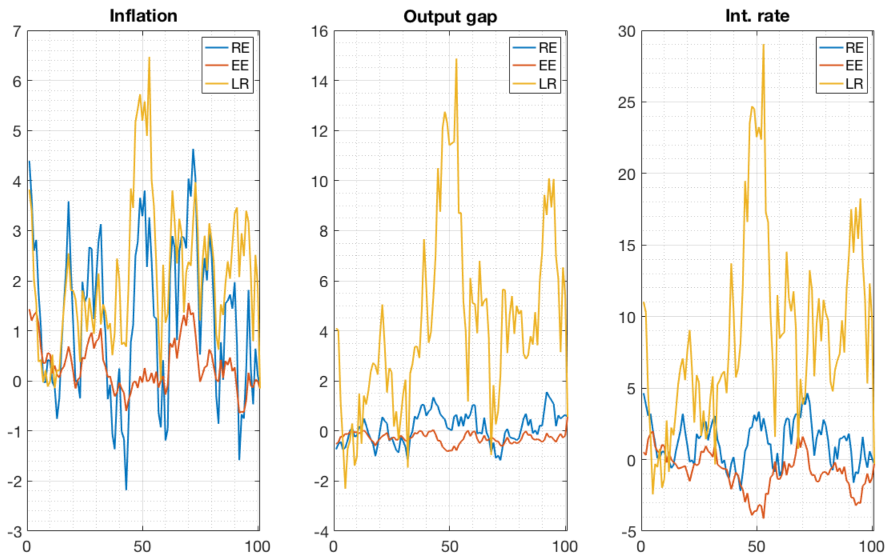
**Figure 6:** RE and LR models, $T = 100000$, last 100 periods




I don't believe it! They are converging ...! (Side note: here, max abs differences in $\phi$ are on the order of magnitude of $10^{-5}$.) After 2 million periods, they nearly overlap, but still not quite. (Diffs are at $10^{-6}$ now. That takes 4 min to run though!)

## 7.2 Learning constant only

**Figure 7:** All models, $T = 200000$, last 100 periods




As I feared, this doesn't look E-stable: even after 1 million periods, neither EE nor LR converges to the RE solution. And this is despite learning clearly converging: max abs differences in the constant are $< 10^{-5}$. Or are they converging, just *much* slower?

Maybe I need to change the PLM such that it nests the REE. If under RE CHECK wheth