Work after 26 May 2020

Trying to map VFI results to those of parametric expectations.

The VFI sol is $V(X)$, where $X = (k_{-1}, \bar{\pi}, r^n, u)$

So I'm trying to input X^{pe} at the sol of PE.

Ok, that gives me the value functions at the sol.

But I want the policy function, $\{i_t\}_{t=1}^T$.

$$V = L_t(x_t, \pi_t) + \beta V$$

So $(1-\beta)V = L_t(x_t, \pi_t)$

Σ period loss given optimal policy

Since I know the form of the function L_t ($\pi_t^2 + \lambda_x x_t^2$, and I know how π_t and x_t depend on i_t), I can solve for i_t from $(1-\beta)V = L(i)$

$$L = \pi^2 + \lambda x^2$$

$$= (\kappa x + s_3 fa + s_4 \cdot s_t)^2 + \lambda_x x^2$$

$$= (\kappa(-\sigma i + s_1 fb + s_2 s) + s_3 fa + s_4 s) + \lambda_x x^2$$

$$L = \left(k\left(-\gamma i + s_1 f_b + s_2 s\right) + s_3 f_a + s_4 s\right)^2$$
$$+ \lambda_x \left(-\gamma i + s_1 f_b + s_2 s\right)^2$$

→ I just need to solve this for i

→ The matlab sol ($\min_i L$ using fminunc) works way better than the analytical sol using Mathematica (materials31.nb) Beware!

Note that you also need to initialize the Matlab solver at i^{PE} so it finds that one.

Peter meeting 26 May 2020

- s_{t-1}: the right thing to do: 6-dim state-vector b/c s_{t-1} are separate states. → which would allow for serial corr in the shocks.

- X^{PE} in $V(X^{PE})$ on RHS in (3)

$$\rightarrow \quad V(X_t^{PE}) - \beta V(X_{t+1}^{PE}) = L$$

- Check whether a finer grid makes a diff w/o estimation, a small grid is fine. The moment you need quantitative exercises, then you need a larger grid.

$$V(X_t^{PE}) = L(i) + \beta E V(X_{t+1}^{PE})$$

If they are iid, then Peter is right to say that $E X_{t+1}^{PE} = X_t^{PE}$. To encompass the case of AR(1), we can write

$$E X_{t+1}^{PE} = hx \, X_t^{PE}$$

but that's not cool for the endog. states in X.

How to compute $E_t(k_{t+1}^{-1})$ and $E_t(\bar{\pi}_{t+1})$?

$$E_t(k_{t+1}^{-1}) = E_t\left[\rho_k k_t^{-1} + \gamma \, fe_{t+1|t}^2\right]$$

$$= \rho_k k_t^{-1} + \gamma E_t\left[\left(\pi_{t+1} - (\bar{\pi}_t + b_1 s_t)\right)^2\right]$$

$$E_t(\bar{\pi}_{t+1}) = E_t\left[\bar{\pi}_t + k_{t+1}^{-1} fe_{t+1|t}\right]$$

$$= \bar{\pi}_t + E_t\left[k_{t+1}^{-1} fe_{t+1|t}\right]$$

→ It all seems to hinge on $E_t\left[fe_{t+1|t}\right]$

$= E_t\left[\pi_{t+1} - \bar{\pi}_t - b_1 s_t\right]$

$= E_t[\pi_{t+1}] - \bar{\pi}_t - b_1 s_t$

↑ $E[ALM]_2(1,1)$

You know what: better idea: 27 May 2020

$$L = \pi^2 + \lambda x^2$$

$$= (\kappa x + \underbrace{s_3 fa + s_4 \cdot s_+}_{=: fas})^2 + \lambda_x x^2$$

$$= (\kappa(-\delta i + \underbrace{s_1 fb + s_2 s}_{=: fbs}) + s_3 fa + s_4 s) + \lambda_x x^2$$

$$\Rightarrow L = (\kappa(-\delta i + fbs) + fas)^2 + \lambda_x(-\delta i + fbs)^2$$

$$= \kappa^2(-\delta i + fbs)^2 + fas^2 + 2\kappa(-\delta i + fbs)fas$$

$$+ \lambda_x(-\delta i + fbs)^2$$

$$= (\kappa^2 + \lambda_x)(-\delta i + fbs)^2 + 2\kappa \cdot fas(-\delta i + fbs) + fas^2$$

$$= (\kappa^2 + \lambda_x)(\delta^2 i - 2\delta fbs\, i + fbs^2) - 2\delta\kappa \cdot fas \cdot i + 2\kappa\, fas\, fbs$$

$$+ fas^2$$

$$L = (\kappa^2 + \lambda_x)\delta^2 i^2 - (2\delta(\kappa^2 + \lambda_x) fbs + 2\delta\kappa \cdot fas) i$$

$$+ 2\kappa\, fas\, fbs + fas^2 + (\kappa^2 + \lambda_x) fbs^2$$

$$(\kappa^2 + \lambda_x)\delta^2 i^2 - (2\delta(\kappa^2 + \lambda_x) fbs + 2\delta\kappa \cdot fas) i$$

$$+ 2\kappa\, fas\, fbs + fas^2 + (\kappa^2 + \lambda_x) fbs^2 - L = 0$$

$$(k^2+\lambda_x)z^2 i^2 - (2z(k^2+\lambda_x)fbs + 2zk\cdot fas)\,i$$

$$+2k\,fas\,fbs + fas^2 + (k^2+\lambda_x)fbs^2 - L = 0$$

$$x_{1,2} = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

$i_{1,2}$

$$= \frac{(2z(k^2+\lambda_x)fbs + 2zk\cdot fas) \pm \Delta}{2(k^2+\lambda_x)z^2}$$

$$\Delta = (2z(k^2+\lambda_x)fbs + 2zk\,fas)^2 - 4(k^2+\lambda_x)z^2(c)$$

$$= (2z)^2((k^2+\lambda_x)fbs + k\,fas)^2 - 4z^2(k^2+\lambda_x)c$$

$$= 4z^2\left[((k^2+\lambda_x)fbs + k\,fas)^2 - (k^2+\lambda_x)c\right]$$

$$i_{1,2} = \frac{2z[(k^2+\lambda_x)fbs + k\,fas] \pm 2z\sqrt{\Delta}}{2(k^2+\lambda_x)z^2}$$

$$= \frac{\cancel{z}(k^2+\lambda_x)fbs + k\cdot fas\cancel{z} \pm \cancel{z}\sqrt{\Delta}}{(k^2+\lambda_x)z^{\cancel{2}}} \quad \checkmark$$

$$\Delta = ((k^2+\lambda_x)fbs + k\cdot fas)^2 - (k^2+\lambda_x)\left[2k\,fas\,fbs + fas^2 + (k^2+\lambda_x)fbs^2 - L\right]$$

$$\cancel{(k^2+\lambda_x)^2fbs^2} + \cancel{k^2fas^2} + \cancel{2k(k^2+\lambda_x)fas\,fbs}$$

$$\cancel{-(k^2+\lambda_x)2k\,fas\,fbs} - (\cancel{k^2}+\lambda_x)fas^2 \cancel{-(k^2+\lambda_x)^2fbs^2} + (k^2+\lambda_x)L$$

$$\Delta = z^2(k^2 + \lambda_x)L - \lambda_x f a s^2 z^2$$

So actually

$$i_{1,2} = \frac{(k^2+\lambda_x) f b s + k \cdot f a s \pm \sqrt{(k^2+\lambda_x)L - \lambda_x f a s^2}}{(k^2+\lambda_x) z}$$

Ok so what have I learned:

1. Since I know there are 2 sols, doing it analytically makes more sense than w/ fsolve.
2. Depending on the values of L, fa, fb, s, some i may be complex, in which case all i show up as complex, some w/ 0 imaginary part. Then one needs a good way to discriminate between the two sols, for every t!

Brian Dombeck macro lunch

Shadow price learning & Expectationally Driven BCs

main diff btwn econ & nat. science: expectations

→ mice!

Evans & McGough: shadow price learning

"SP learning"

SP learning: agents learn about shadow prices

→ using an RLS scheme (so it's a special case of adaptive learning)

- ↳ ask for slides!

Ulrich: ppl have been saying $\pi > 2\%$ for a long time! → hard to reconcile w/ RE.

- if news shocks can be incorporated into the info set of agents, then maybe one can trick agents into responding to promises of the gov.

Is "news shock" a proxy for adaptive learning?
Eusepi & Preston AER says that adaptive learning looks a lot like news-shock-model.

- Nick Bloom "TV volatility is exog"

- news & noise are equiv under RE
  → not obvious if they are outside RE
  Brian thinks they should be equiv outside RE b/c they are exog processes.

News shock in RE : can only generate comovement if $\sigma < 0.25$ instead of 1.

- SP-learning emphasizes the distinction btwn market clearing conditions vs. model eqs.
  → so it seems like an improvement over EE learning

SP-learners are less sophisticated than
EE-adaptive-learners.

- Is it really an RPE?

What's a state shadow price?

value to me today of an additional unit
of a state

Ryan: → $\mathcal{L}$ multipliers on <u>intertemporal</u> constraints
(multipliers on static constraints don't
matter b/c you can sub 'em out)

He lets them regress on MSV sol.

He simulated T=500.

- job market materials?

willing to read?

• These models are usually not treated in their non-linear form!

Ryan meeting

27 May 2020

Can approx the policy fct i the same way b/c you have generated values for i at every point in the state space.

Trick to save time: convex combo of VFI policy iter : faster than VFI but more stable than PI.

Update policy fct every 5^{th} iter.

Keep the policy the same for some iterations

↳ saves you the max step.

The distinguishing elements of global methods

- gridpoints
- thing you're approxing
- how you you find zeros

Value function is also a projection method, but in any case, policy iteration is a projection b/c it works directly w/ FOCs.

Brian meeting 27 May 2020

will send job market materials.

EH 2001 : misspecified models & RPE

Chap. 13

Evans & McGough Shadow Price Learning Paper

↳ sends!

End of it is interesting! NK!

fwd guidance as a news shock

Offshoot: currency crisis models untethered

targeting underpinning it

everything is fine, boom!

→ exog shift in E( )

Schmitt-Grohe & Uribe Open Econ Textbook

pdf online!

Let's talk periodically! Figure out over email!

Ok so mixed news: 29 May 2020

1) VFI = PEA...

2) ... only if you approx the policy function using a sufficiently large $\bar{\pi}$-grid: $\bar{\pi} \in [-0.2, 0.2]$ is clearly too small! $\bar{\pi} \in [-4, 4]$ is the smallest that seems approximately ok. The problem is that VFI w/ $\bar{\pi} \in [-10, 10]$ started diverging at one point. But I suppose I should use the same grid for VFI as for i-approx...?

Or not?

Saying that they should have the same grid is to say that we're approxing the value function on the same grid as the policy fct.

I don't know. The good news is that it's not the acceleration that screws up the results. It's the $\bar{\pi}$-grid. My concern is that the grids should be the same b/c we use i_t-values generated

from a grid X_1, so $i_t = \hat{V}(X_1)$
and then the coefficients that approximate
$\hat{i}(\alpha, i_t(X_1), X_2)$ shouldn't be the function of
a different grid.
Let's see if the $\bar{\pi} = [-4, 4]$ grid thus works for
VFI.

Let's try to figure out why my value function
sometimes diverges (it has a kind of spectacular
reversal). In theory (and it looks like, in practice too)
VFI is guaranteed to converge, albeit slowly.
So I'm wondering if there's something about the
learning that can screw things up: for a large
value of $|\bar{\pi}|$, some unpermitted policy is chosen
that causes say a $k_t^{-1} < 0$ or sthg. Maybe
it has to do w/ a potential incorrect (or lacking)
specification of the problem constraints? But if that's
so then only doing the i-update every jj^{th} iter

may actually be helpful b/c you don't do the max step as many times.

But $k_t^{-1} = \gamma_k k_{t-1}^{-1} + \gamma f e_t^2$

definitely keeps $k_t \geq 0$. What it can do though is that suddenly fe can become huge: in fact: the bigger & more course the grid, the more likely that is to happen b/c $\bar{x}$ changes a lot from one iteration to the next. On the other hand, since we're on a grid, the result of every iteration is $V(X)$ where X is all combination of all states. Divergence can only happen if somewhere along X, $V^{l+1}(X)$ is <u>very</u> different from $V^l(X)$

Can it be that k_t reaches 0? No b/c it's safely constrained by the grid. The only thing that can happen is that $V(X_{t+1})$ is such that an i is chosen which gives a very diff Lx and thus TV.

Ha! What I'm finding from this $\bar{\pi} = [-4, 4]$ VFI is that now I need to use a much bigger grid for $\hat{i}$ (e.g. $\bar{\pi} \in [-40, 40]$ is too small,
$\bar{\pi} \in [-100, 100]$ is fine)
for it to have the same shape as PEA.
Let's make the ~~k~~grid start at 0.0001 instead of 0.
↳ doesn't change it

Obs:

- The bigger $\bar{\pi}$ grid, the bigger are the changes in V initially. Why? B/c more "data variation?"
- For $\bar{\pi} \in [-4, 4]$, the i^* are huge: $\in [-300, 300]$. I'm wondering whether this is what makes it desirable to scale up the $\bar{\pi}$ grid afterwards for $\hat{i}$. → Seems to suggest that VFI suffers from a scaling issue somewhere, especially as the pattern is right, just the scale off.

Ok so I've set up command_vfi & command_pea to investigate the two methods; compare_value_pea_results compares the two. Now at this point I'm really wondering if the PEA is wrong.

→ Continue there!

- Initialize PEA at different rand   30 May 2020
  → gives the same
- Try 1e-10 as eps instead of 1e-6 in PEA
  → gives the same, took 20 min.
- Now input a different sequence of X and see if the scaling issues remain the same
  → it's diff of course but in the grand scheme of things it's similar in that
    i) it has big up/down oscillations initially
    ii) later in the sample it moves around 0 in the ±5% band.

⇒ the scaling issue in VFI seems to be there, no actually, no; the $\bar{x}$-grid used for VFI was $[-4, 4]$, and loading it for the $\hat{i}$-approx, I get the same thing.

value_outputs_server32-accelerated

pea_outputs_30-May-2020-10_18_28

Now try this w/ value_outputs_ w/ a $[-4, 4]$ grid!

→ it works, holy smokes! I must have used non-compatible grids. Yeah!

<u>To do for First Draft (= Draft 4)</u> <u>31 May 2020</u>

- ✓ Rewrite the analysis part fundamentally
- ✓ Once you have that, rewrite the intro in the spirit of Adam
- ✓ Once you have that, rewrite the abstract.

• Analysis

✓ – Target inflation

✓ – Implementing: optimal int. rate sequence vs. a
std Taylor rule.

✓ • policy fct as a function of X:
function of π, of k_t^{-1}

X • reaction fct wrt observables, π & X?
→ relation to TR

→ numbers to the opt. policy response!

✓ • Corr btw k & π and response of $\hat{i}$ to wander.

0.0008.8981e-04 = 0.00088981

✓ • App PEA & VFI

✓ • Conclusion

✓ • Intro: Related lit ← 2 June start here!

✓ • Go over references

✓ • Go over App. ←

✓ • visual identity for figures ← working on it

✓ • Bigger grid server

For after:

- Basu's liq. premium correction
- TR w/ smooth crit → maybe once est is done.
- TV TR
- more relation of policy function to TR
- estimation

A first read on tablet Notes: 2 June 2020

Nope.

Ryan meeting 5 June 2020

- interpretation of LoM-k: concave/convex, slope need derivative, so you only need to plot it.
- point of TR: a simple, verifiable description of mon pol. ⇒ a TV-TR has all the complications of a TL, so no need to do.

⇒ i'm at an inflection point in my work. Ryan's prior was: write up things, and then estimate, but now it seems like the prob (estimation changes the results & the pitch of the paper) is fairly high. Therefore:

→ spend up to a month estimating

if in 2 weeks I see that I won't get any quick results from estimating, then I can still revert to sending this draft.

NB. "I need 2 weeks of lead time." in the contract means "expect to have 2 weeks until he gives you feedback", <u>not</u> "give him a heads-up 2 weeks before submitting".

<u>Work after</u> Today is June 5.

+ 2 weeks is June 19.

+ 4 weeks is July 3.

Work after

Ok so the draft is done for now. The only thing it will need to do is incorporate Ryan's idea of why a time-invariant TR exercise makes sense.

Don't apologize: instead, say that the whole point of a TR is a simple, easily computable and verifiable implementation of policy.

As for the estimation, I return to the existing code command_GMM_anchoring_function.m, but replace the objective function.

- Need to redo bootstrap as a VAR
- Need to redo objective function
  - → incorporating a shape-preserving spline

→ Actually, I decided to leave that old file be. Instead, I moved stuff to materials33.m

1) First step is to make the bootstrap and the weighting matrix work

7 June 2020

2) Then the GMM will have EXACTLY the same structure as before, except LOM_k will be replaced by $B \cdot s(X)$ where B are the coeffs and $s(X)$ is the basis (hopefully a shape-preserving spline or simply a piecewise-linear approx).

3) ⇒ shape-preserving spline.

1) I'm scrapping bootstrap_resample.m b/c it's useless: it's just resampling the errors w/o building a new (VAR) sample. Instead, I'm using the old code data_boot.m from the IT-project.

So I have dataset_boot, which is the nboot × T bootstrap-resampled VAR dataset.

→ Now what I need is a procedure to estimate the ACF of a VAR. Hamilton, p. 280 MaC seems to provide such an approach.

The j^{th} autocovariance Γ_j of the original process y_t is the first n columns and n rows of

$$\Sigma_j = F\Sigma_{j-1} \quad j=1,2,\dots \quad [10.2.20]$$

which is

$$= F^j \Sigma$$

$$\Gamma_j = \Phi_1 \Gamma_{j-1} + \Phi \Gamma_{j-2} + \dots + \Phi_p \Gamma_{j-p}$$

for $j = p, p+1, p+2, \dots$ [10.2.22]

The VAR-notation Hamilton uses is

$$y_t = c + \Phi_1 y_{t-1} + \Phi_2 y_{t-2} + \dots + \Phi_p y_{t-p} + \varepsilon_t$$

VAR(p) [10.1.4]

Rewrite this in terms of deviations from the mean as

$$\mu = (I_n - \Phi_1 - \Phi_2 - \dots - \Phi_p)^{-1} c$$

$$(y_t - \mu) = \Phi_1(y_{t-1} - \mu) + \Phi_2(y_{t-2} - \mu) + \dots + \Phi_p(y_{t-p} - \mu) + \varepsilon_t$$

Use that to rewrite as

Q = VC matrix of v_t

$$\xi_t = F\xi_{t-1} + v_t$$

VAR(1) [10.1.11]

where $\xi_t = \begin{bmatrix} y_t - \mu \\ y_{t-1} - \mu \\ \text{etc} \end{bmatrix}$ (demeaned y_t) [10.1.5]

$[np \times 1]$

The j^{th} Autocovariance Matrix

$$\Gamma_j := E\left[(y_t - \mu)(y_{t-j} - \mu)'\right] \qquad [10.2.1]$$
$$= E\left[\xi_t \xi_{t-j}'\right]$$

Then $\Sigma := E\left[\xi_t \xi_t'\right]$.

Note that

$$\Sigma = \begin{bmatrix} \Gamma_0 & \Gamma_1 & \cdots & \Gamma_{p-1} \\ \Gamma_1' & \Gamma_0 & & \Gamma_{p-2} \\ \vdots & & \ddots & \vdots \\ \Gamma_{p-1}' & \Gamma_{p-2}' & \cdots & \Gamma_0 \end{bmatrix} \qquad [10.2.12]$$

Ah I see. I need to rewrite my p lag VAR(p) as a VAR(1).

Recall from [10.1.11] that

$\xi_t = F\xi_{t-1} + v_t$. Thus $\Sigma = E\left[\xi_t \xi_t'\right]$

$$= E\left[(F\xi_{t-1} + v_t)(F\xi_{t-1} + v_t)'\right]$$
$$= E\left[(F\xi_{t-1} + v_t)(\xi_{t-1}'F' + v_t')\right]$$
$$= E\left[F\xi_{t-1}\xi_{t-1}'F' + v_t v_t' + \text{crossterms}\right]$$
$$= FE\left[\xi_{t-1}\xi_{t-1}'\right]F' + E\left[v_t v_t'\right]$$

$$\Sigma = F \Sigma F' + Q \qquad [10.2.13]$$

This can be solved for Σ using vec as

$$\text{vec}(\Sigma) = \underbrace{\left[I_{r^2} - (F \otimes F) \right]}_{r^2 \times r^2}^{-1} \text{vec}(Q)$$

Then, having Σ, we find the ACF_j (Γ_j) by computing first Σ_j, the ACF of the VAR(1).

$$\Sigma_j = F^j \Sigma . \qquad [10.2.21]$$

Finally, we obtain the ACF of the VAR(p), Γ_j, as the first n rows and colums of Σ_j.

What is n, p, r?

p = order of VAR

$\underset{np \times np}{F}$ so n must be the # of variables

Maybe $r^2 := (np)^2$? Yep!

→ solved bootstrap and weighting matrix by using the estimate of the ACF from Hamilton!

A day in Prori – in honor of which <u>8 June 2020</u>
I'm using Brown ;-)
⟹ focus on shape-preserving splines
In Matlab: pchip seems to be the way to go
since, believe it or not, it actually is a one-dimensional object.
I just wanna find out in what relationship pchip stands w/ Schumaker.

Well that was it ;) <u>9 June 2020</u>
See Splines and pchips from Cleve's Corner
(saved as pdf in numerical_methods in next.
The point is that Hermite cubics and splines are both piecewise cubic polynomial interpolants.
The difference is that Pchips (Hermites) is the 1st & 2nd derivative conditions. A spline (at least in Matlab) imposes 2nd derivative conditions

at the knots and at the endpoints. The Hermite interpolating polynomial only uses 1st derivative conditions. (For this reason, the 2nd derivative may have kinks and jumps, while for a spline it's smooth & continuous. Even the 1st derivative of the Hermite exhibits kinks, while that of the spline is smooth.)

In particular, the Hermite requires that at interior points, the slope is a weighted harmonic mean of the slopes of the piecewise linear interpoland, while at the endpoints, one-sided slope conditions are imposed. That's it, and for this reason, it's less smooth but instead shape-preserving than the spline.

Harmonic mean

$$HM(x_1, \dots, x_n) = \left(\frac{x_1^{-1} + \dots + x_n^{-1}}{n} \right)^{-1}$$

Arithmetic mean = $AM(x_1, \dots, x_n) = \frac{x_1 + \dots x_n}{n}$

It looks like for ND data, the only hope is spapi, where 10 June 2020

1) you can specify <u>the order</u> of the interpolating polynomial

2) it may be that the cubic is shape-preserving

But spapi is in B-form which is, technically, a pp (piecewise polynomial), w/ some differences. The coefficients are $a_{d \times n}$ w/ breaks $t(i)_{(n+k)+1}$

Formally, the spline f is defined as

$$f = \sum_{i=1}^{n} B_{i,k} \, a(:,i)$$

↓

i^{th} B-spline
of order k
for the given knot sequence $t(i), \ldots, t(i+k)$

→ Need to use spmak to construct the B-form/pp.

Gaetano Gaballo: Learning by Shopping

Mich Survey of $\pi - E(\cdot)$:

- huge dispersion

but $\text{disp}(\pi - E(\cdot)) < \text{disp}(\text{actual } \pi)$

i.e. disp (price level at individual level)

HHs

→ . very heterogen $\pi - E(\cdot)$

. very influenced by their limited shopping experience

Ryan meeting — 10 June 2020

example for ndim_simplex.m! → He sent it.

↳ test_simplex2d.m

Use a grid for k_t, $fe_{t+1|t-1}$, then use A.6 to get initial values for b_t, $fe_{t+1|t-1}$.

Actually, this is important b/c if you simulate, You may not cover the entire domain of the function and therefore you may be underidentified (b/c a finite-element approach needs data from all elements, otherwise it's singular, vs. global interpol /spectral methods like Cheby.)

The other thing is the inputs to ndim_simplex: x and xx
x = is the whole grid: it stays constant for every evaluation of the fct.
xx = is the current input point; it changes for every evaluation.

Triangulation is about the efficient choice of triangles, once you have a grid. But for Ryan, the choice of a (uniform) grid is tied to the triangulation b/c he just picks a triangulation that he knows exists for a uniform grid. But it's not efficient, and it seems efficient methods aren't implemented for ND (> 3).

Also note that in the binning step, ndim_simplex.m stretches the triangles at the edges so that if you have a data point <u>outside the grid</u>, you still get attributed to that edge-triangle, so you still obtain a value (this is extrapolation).

<u>Work after</u>

I wonder if I should do lag selection for
1) all bootstrap samples
2) model-implied datasets individually or should I use the lag selected for the initial data?

→ I always (or very often) have singularity issues w/ my regressor and I think it might come from having too many lags

Cont here!

↳ Need to solve singularity issues w/ regressors.

→ Once that is done, maybe the coeffs will update properly. → See Peter's email, saved into notes as Peter_email_11_June_2020.pdf.

stochastic singularity

I'm still dropping that issue for now and turning to the ndim_simplex issue.

11 June 2020

I think it's happening b/c the gain goes sufficiently close to -Inf.

⇒ How can I tell my approx not to result in negative gains??

→ If I allow the α-coefficients to move too far from α_0, I get this issue.

Gaetano Gaballo meeting 11 June 2020

- don't fight rational expectation
  ↳ be specific about language
  ⇒ smoother ways of saying you are a member of the other crowd
  → produce large deviations in $\bar{\pi}$ is very hard but we need very small departures in the model

You use a lot (too much!) of the language of the adaptive learning
e.g. the understanding of RE
↳ Guesnerie → you may not end up in REE even if you know the model b/c of coordination failure

Marcet & Adam & Evans & Preston are the ones successful in selling the argument: how large is the departure that can explain → we want the smallest departure that can still explain the stuff.

" W/ a small departure in modelling I can achieve deviations from $\bar{\pi}$ from target and nice intuitions about behavior"

There should be a mapping from k_t to a Kalman gain.

Graph to explain the target criterion
→ relevant graph here: utility indifference curve of planner
Maybe simplify into 2 periods

Ppl will not be surprised that w/o RE, discretion = commitment. So the contrib. is how the learning changes the problem.

should sell to CB b/c it's "instructions to a non-Lucas world"

→ emphasize not only what is opt pol, but what's the mistake I make when I follow a RE monpol.

- RE is great in LR
- but crisis has shown that things move quickly
- so how costly is it to ass plain vanilla RE

Recall that discretion = commitment

But the sol differs depending on the model

So I do 3 things

1. Est. model
2. solve opt. policy
3. Assess the mistake I do by ass RE

Add numbers.

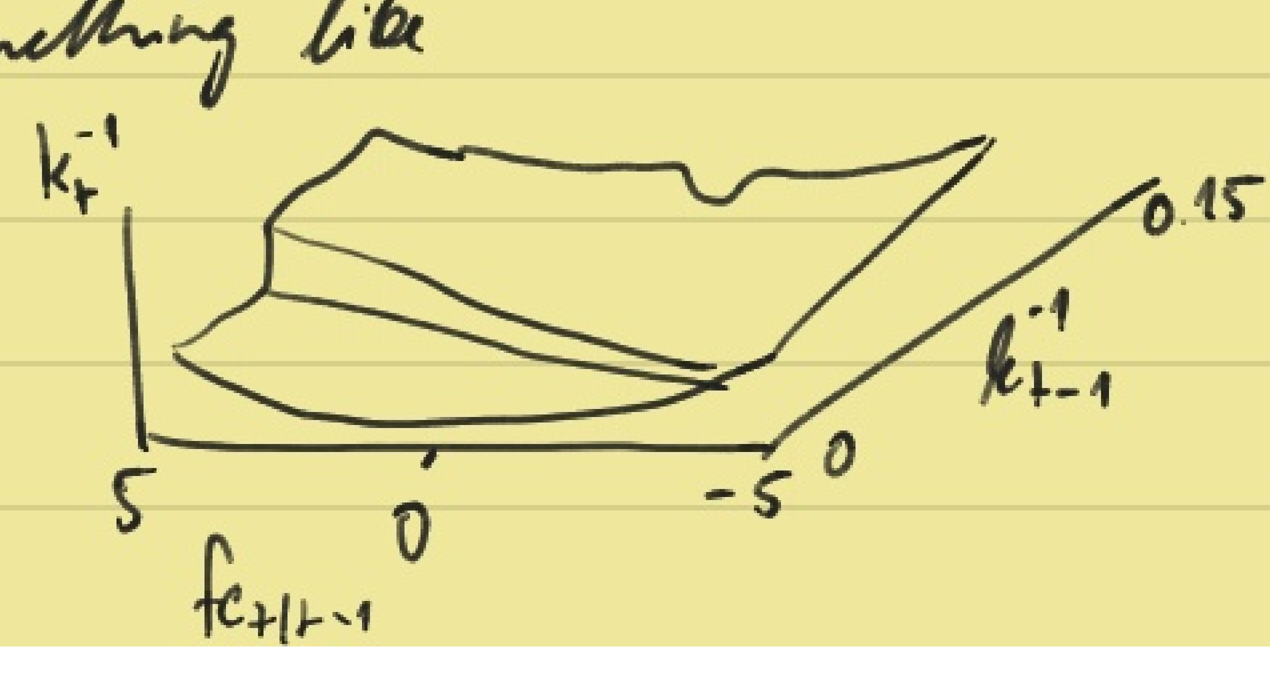
A two-period problem w/ an intertemporal price ppl can learn about, you can already show the mistake ppl make by using the RE ass.

He said it's fine to talk at the end of summer.

<u>Work after</u> <u>12 June 2020</u>

I needed to make sure the simulation doesn't explode for certain values of α, so whenever an explosion happens, I set the GMM loss to 1e+10. (It's otherwise on an order of 3e+02.)

The estimated relationship (when $\kappa > 0$ constrained) is something like



1) For a small k_{t-1}^{-1}, a given k_{t-1}^{-1} is associated w/ a larger k_t^{-1} if the $fe_{t|t-1} < 0$ than if it is $fe_{t|t-1} > 0$.

NB. $fe_{t|t-1} < 0 \rightarrow \pi_t - fcst < 0 \Rightarrow \pi_t < fcst$

$\Rightarrow$ surprise deflation / disinflation

"negative inflation surprise"

"overestimated inflation"

2) This is no longer true for high k_{t-1}^{-1}: then a given k_{t-1}^{-1} is associated w/ higher k_t^{-1} if $|fe|$ higher except for when fe is small and negative.

$\rightarrow$ Let's rephrase in words

0) E(·) are always more likely to unanchor for larger FE.

1) When expectations are anchored, they are more likely to unanchor for negative inflation surprises.

2) When E(·) are unanchored, then small negative inflation surprises are associated w/ faster anchoring.

$\Rightarrow$ This may be driven by my sample which spans

1959 - 2020 (Jan 1 to Jan 1), so the Great Inflation is in there, under which $E(\cdot)$ were unanchored, and seeing π come in lower than expected was stabilizing. Once $E(\cdot)$ became anchored in the 90s and especially in the 2010s (after 2008), inflation shortfalls are more likely to unanchor $E(\cdot)$.

⇒ Neat!

Redo PEA w/ this?