The Sequence Space

Juan Herreño   Johannes Wieland

UCSD, Spring 2024

Outline

Outline

Solving Models

- Hard when there is rich heterogeneity.
- Infinitely dimensional state space from the distribution of agent's endogenous state variables—how to handle it?
- In macro, we are often interested what happens when aggregate shock X hits the economy (monetary, fiscal, etc).
- State space: must forecast infinite-dimensional distribution to estimate prices / quantities relevant to agents' problem.
- Sequence space: must find price / quantity sequences that solve market clearing conditions.

$\Rightarrow$ Sequence space is of dimension $T \times$ number of variables, versus infinite dimensional state space.

- Cost: linearization / perfect foresight.

Sequence Space Methods

- Sequence Space methods mean we use linear algebra to solve models with rich heterogeneity.
- But how do we get matrices like these:

$$\nabla_{\boldsymbol{Y}} \mathbf{C} = \begin{pmatrix} \frac{\partial C_0}{\partial Y_0} & \frac{\partial C_0}{\partial Y_1} & \cdots \\ \frac{\partial C_1}{\partial Y_0} & \frac{\partial C_1}{\partial Y_1} & \cdots \\ \frac{\partial C_2}{\partial Y_0} & \frac{\partial C_2}{\partial Y_1} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

- Need to solve the household consumption problem: hard.
- Auclert, Bardoczy, Rognlie, Straub (2021): efficient algorithm that exploits model structure and linearity to get the Jacobians of the model really fast!

OUTLINE

Household

- Preferences:

$$V_t(e, a_-) = \max_{c,k} u(c) + \beta \sum_{e'} V_{t+1}(e', k) P(e, e')$$

- Budget constraint

$$c + k = (1 + r_t)k_- + w_t en$$

- Borrowing constraint

$$k \geq 0$$

- Optimal policy:

$$c_t^*(e, k_-),\ k_t^*(e, k_-)$$

- Functions of $\{r_t, w_t\}_{t \geq 0}$.

Aggregating Consumer Problem

- Distribution of capital and productivity:

$$D_{t+1}(e', K) = \sum_e D_t(e, k_t^{*-1}(e, K))P(e, e')$$

- Given D_0, $D_{t+1}(e', K)$ are also functions of $\{r_t, w_t\}_{t\geq 0}$.
- Aggregate capital holdings are therefore also a function of $\{r_t, w_t\}_{t\geq 0}$.:

$$A_t(\{r_s, w_s\}_{s\geq 0}) = \sum_e \int_{k_-} k_t^*(e, k_-)D_t(e, dk_-)$$

- The function A_t maps an aggregate sequence $\{r_t, w_t\}_{t\geq 0}$ into another aggregate sequence $\{K_t\}_{t\geq 0}$.
- The dimensionality of this problem is the length of the sequence T.

Firm and Market Clearing

- Keep this simple:

$$
\begin{aligned}
Y_t &= Z_t K_{t-1}^{\alpha} N_t^{1-\alpha} \\
r_t &= \alpha Z_t K_{t-1}^{\alpha-1} N_t^{1-\alpha} - \delta \\
w_t &= (1-\alpha) Z_t K_{t-1}^{\alpha} N_t^{-\alpha}
\end{aligned}
$$

- Labor supply fixed in this problem so

$$N_t = \sum \pi(e) en$$

- Market for capital has to clear

$$H_t(\boldsymbol{K}, \boldsymbol{Z}) \equiv A_t(\mathbf{r}, \mathbf{w}) - K_t = 0$$

DAG Representation

IRFs

- You know how to compute $\nabla_{\mathbf{K}}\mathbf{r}, \nabla_{\mathbf{U}}\mathbf{r}, \nabla_{K}\mathbf{w}, \nabla_{U}\mathbf{w}$ from firm block.

- How do we compute $\nabla_{\mathbf{r}}\mathbf{A}$ and $\nabla_{\mathbf{w}}\mathbf{A}$?

Outline

Fake News Algorithm

- Goal: efficient computation of Jacobians.
- Framework:

$$
\begin{aligned}
\boldsymbol{v}_t &= v(\boldsymbol{v}_{t+1}, \boldsymbol{X}_t) \\
\boldsymbol{D}_{t+1} &= \Lambda(\boldsymbol{v}_{t+1}, \boldsymbol{X}_t)' \boldsymbol{D}_t \\
\boldsymbol{Y}_t &= y(\boldsymbol{v}_{t+1}, \boldsymbol{X}_t)' \boldsymbol{D}_t
\end{aligned}
$$

  - $\boldsymbol{X}_t$ are inputs (e.g., Z_t).
  - $\boldsymbol{Y}_t$ are outputs (e.g., C_t, K_t).
  - $\boldsymbol{D}_t$ is the discretized distribution.
- In sequence space:

$$
\boldsymbol{Y} = h(\boldsymbol{X})
$$

- Want: $\mathscr{J} = \nabla h_{\boldsymbol{Y}}(\boldsymbol{X}^{SS})$

Brute Force

- Column s of $\mathscr{J}_X^Y$ is IRF of outcome (e.g., capital) to a one-time shock (e.g. productivity) at s, $\boldsymbol{X}^s = \boldsymbol{X}_{ss} + \boldsymbol{e}^s dx \equiv \boldsymbol{X}_{ss} + d\boldsymbol{X}^s$.
- Solve problem backward given the known sequence of $\boldsymbol{X}$ to get value functions $\boldsymbol{v}_t$, policy functions $\boldsymbol{y}_t^s$, and transition matrices $\Lambda(\boldsymbol{v}_{t+1}, \boldsymbol{X}_t)'$.
- Given optimal policy functions iterate distribution of agents forward starting from steady state,

$$\boldsymbol{D}_{t+1}^s = (\Lambda_t^s)\boldsymbol{D}_t^s$$

- Aggregate policy functions using the distribution to get outcome

$$\mathbf{Y}_t^s = (\boldsymbol{y}_t^s)'\boldsymbol{D}_t^s$$

- Repeat for $s = 0, ..., T$ to get all columns of $\mathscr{J}$.
- This "brute force" method works but is very slow.

Tasks

- Solve for steady state in consumption problem.
- Get the first column of $\mathscr{J}_{\mathbf{r}}^{\mathscr{K}}$.
- Get the second column of $\mathscr{J}_{\mathbf{r}}^{\mathscr{K}}$.

Algorithm: Efficient backward step

- Lemma 1: for any $s \geq 1$, $t \geq 1$:

$$\boldsymbol{y}_t^s = \begin{cases} \boldsymbol{y}_{ss} & s < t \\ \boldsymbol{y}_{T-1-(s-t)}^{T-1} & s \geq t \end{cases}, \qquad \Lambda_t^s = \begin{cases} \Lambda_{ss} & s < t \\ \Lambda_{T-1-(s-t)}^{T-1} & s \geq t \end{cases}$$

- Policy functions at t for shock at $t+s$ the same as policy function at 0 for shock at s.

$\Rightarrow$ Policy functions need to be solved backwards only once starting with a shock at $T-1$. (vs T times in brute force.)

- For any time after the shock, the policy functions are the same as in steady state.

- Key: policy function cannot directly depend on distribution of agents

Fake News

- Denote the sequence $\boldsymbol{\varepsilon}^s$ in which $\varepsilon_s = 1$ and zero otherwise:

$$\varepsilon_t = \begin{cases} 1 & t = s \\ 0 & t \geq 0, t \neq s \end{cases}$$

- Let η_t^s be a fake news shock for ε_t:
  - At t learn that $\varepsilon_s = 1$ with certainty.
  - At $t+1$ lean that $\varepsilon_s = 0$.
  - $\mathbb{E}_k \eta_t^s = 0$ for all $k < t$.
- Lemma: The sequence $\boldsymbol{v}^s$

$$v_t = \begin{cases} \eta_t^{s-t} & t < s \\ \boldsymbol{\varepsilon}^0 & t \geq s \end{cases}$$

  takes on the same expected values and realized values as $\boldsymbol{\varepsilon}^s$.
- Next step: build Jacobian based on $\boldsymbol{v}^s$ rather than $\boldsymbol{\varepsilon}^s$.

Algorithm: Efficient forward step

- If we return to steady state and the shock is infinitesimal,

$$
\begin{aligned}
F_{t,s}dx &= dY_t^s - dY_{t-1}^{s-1} \\
&= \boldsymbol{y}_{ss}'(\Lambda_{ss}')^{t-1} d\boldsymbol{D}_1^s
\end{aligned}
$$

- Thought experiment: shock happens at $s-1$ and we are interested in outcomes $t = 0, ..., T-1$. How is this different from a shock that happens at s on outcomes $t = 1, ..., T$?
- From earlier we know the policy functions are the same.
- The only difference is the distribution of agents: $\boldsymbol{D}_s s$ in the first case and $\boldsymbol{D}_1^s$ in the second.
- To a first order, this only affects outcomes as if all agents followed their steady state policy functions.

Fake News Interpretation

- For a given s, $F_{t,s}$ can be interpreted as the impulse response to a "date-s fake news shock": a shock to date s announced at date 0, and retracted at date 1.

- At date 0, agents react to the announcement, which leads to the distribution $\boldsymbol{D}_1^s$.

- After the announcement is retracted, they revert to steady-state policies, so the effect on output at all dates $t \geq 1$ is $\boldsymbol{y}_{ss}'(\Lambda_{ss}')^{t-1} d\boldsymbol{D}_1^s$.

Fake News Matrix

- Define the expectation vector as

$$\mathscr{E}_t = (\Lambda_{ss})^t \boldsymbol{y}_{ss}$$

- Define the fake news matrix as

$$\mathscr{F}_{t,s} dx \equiv \begin{cases} d\mathscr{Y}_0^s & t = 0 \\ \mathscr{E}'_{t-1} d\boldsymbol{D}_1^s & t \geq 1 \end{cases}$$

- Then the Jacobian of h is given by

$$\mathscr{J}_{t,s} = \sum_{k=0}^{\min\{s,t\}} \mathscr{F}_{t-k,s-k}$$

- $t = 0$ follows by definition.
- Why does this make sense?

Fake News Matrix Examples

- $t > 0, s = 0$:

$$\mathscr{J}_{t,0} = \boldsymbol{y}_{ss}'(\Lambda_{ss}')^{t-1} d\boldsymbol{D}_1^0$$

  - Shock at 0 affected distribution at 1. Now trace out this effect.

- $t = 1, s = 1$:

$$\mathscr{J}_{1,1} = d\mathscr{Y}_0^0 + \boldsymbol{y}_{ss}' d\boldsymbol{D}_1^1$$

  - Sum of surprise contemporaneous shock and fake news shock.

- $t = 1, s = 2$:

$$\mathscr{J}_{1,2} = d\mathscr{Y}_0^1 + \boldsymbol{y}_{ss}' d\boldsymbol{D}_1^2$$

  - As if we got news today about shock tomorrow, but taking into account change in distribution given that news was known earlier.

$$\mathscr{J}_{3,4} = d\mathscr{Y}_0^1 + \boldsymbol{y}_{ss}' d\boldsymbol{D}_1^2 + \boldsymbol{y}_{ss}'(\Lambda_{ss}') d\boldsymbol{D}_1^3 + \boldsymbol{y}_{ss}'(\Lambda_{ss}')^2 d\boldsymbol{D}_1^4$$

Bottom Line

- $t = 1, s = 0$:

$$\mathscr{J}_{t,s} = \mathscr{J}_{t-1,s-1} + \mathscr{F}_{t,s}$$

  - A shock at s from time t looks very similar to a shock at $s-1$ from time $t-1$.
  - Policy functions are exactly the same.
  - Only difference is that the distribution at t is different from $t-1$, which is captured by the fake news term.

$\Rightarrow$ Very small number of steps / computations to get all of the Jacobian and therefore the IRFs.

Tasks

1. Compute backward step once.
2. Compute the first row of the Jacobian / FNM $\{d\mathscr{Y}_0^s\}_{s=0}^T$.
3. Compute $d\boldsymbol{D}_1^s$ using steady state distribution and $s = 0, ..., T$ policy functions.
4. Compose the fake news matrix.
5. Compose the Jacobian using using the fake news matrix.
6. Combine matrices to get $\mathbf{H_U}$ and $\mathbf{H_K}$
7. Compute IRF to AR(1) productivity shocks with $\rho_a = 0.9$ at $t = 0$, $t = 10$.