Lecture 5: Putting Algorithms to Work: Solving an Income Fluctuations Problem

Fatih Guvenen
University of Minnesota

November 2023

Let's Circle Back to the Beginning

- We started with the goal of solving dynamic programs.
- One of the simplest problems you will encounter is the income fluctuation problem:

$$V(k_i, z_j) = \max_{c,k'} \left\{ \frac{c^{1-\gamma}}{1-\gamma} + \beta \mathbb{E}\left(V(k', z') \mid z_j\right) \right\} \tag{1}$$

$$\text{s.t} \quad c + k' = (1+R)k + z$$

$$\ln z' = \rho \ln z_j + \eta', \qquad k' \geq k_{\text{min}}.$$

Choosing Parameters

	Model Parameters		
Parameter	Description	Value(s)	Comment
γ	RRA	2 & 10	
ρ	Persist. of earnings	0.90 & 0.98	
σ_{z}	Uncond. std. dev. of z	0.20	
σ_{η}	Innovation std. dev.	$0.20 \times \sqrt{1-\rho^2}$	Gaussian
β	Time discount factor	0.95 & 0.99	
R	Interest rate	Calibrated for $\bar{\mathrm{k}}/\bar{\mathrm{z}} = 5$	
ψ	Borr. limit (% of natural limit)	0.6 & 0.1	

Choosing Parameters

Parameter	Description	Value(s)	Comment
	Model Parameters		
γ	RRA	2 & 10	
ρ	Persist. of earnings	0.90 & 0.98	
σ_z	Uncond. std. dev. of z	0.20	
σ_η	Innovation std. dev.	$0.20 \times \sqrt{1-\rho^2}$	Gaussian
β	Time discount factor	0.95 & 0.99	
R	Interest rate	Calibrated for $\overline{k}/\overline{z} = 5$	
ψ	Borr. limit (% of natural limit)	0.6 & 0.1	
	Choices for Numerical Solution		
N	# grid points	Experiment from 20 up	
k_0	Lower bound	$k_{\text{min}} = -(0.6/R) z_{\text{min}}$	
k_N	Upper bound	500	
Support of z	Discrete grid	11-states	Rouwenhorst
θ	Expansion exponent	3, 1.3, 1	
	Optimization method	Brent	
	Interpolation	3-spline or pcws-linear	
V_0	Initial guess	$V_0(k_i, z_j) = U((1+R)k_i + z_j)$	
Stop. criteria	$\max_i\left(\left(V_i^{n-1} - V_i^n\right)/(1+\lvert V_i^n \rvert)\right)$	$< 10^{-7}$	i indexes grid pts

Setting Up

- Capital grid: $G_k \equiv \{k_0, k_1, \ldots, k_N\}$

Setting Up

- Capital grid: $G_k \equiv \{k_0, k_1, \ldots, k_N\}$
- How to choose k_N? As discussed, there are two ways:
  1. Choose such that $g(k_N, z) < k_N$ for sure. Need a very wide grid and some side calculations.
     - Inefficiently large grid but safer choice to start
     - Cannot be used when changing parameters during estimation/calibration
  2. Choose a large k_N but be ready to extrapolate.. with lots of trepidation!
     - If you extrapolate more than $k_N - k_{N-1}$, your grid is too small.

Setting Up

- Capital grid: $G_k \equiv \{k_0, k_1, \ldots, k_N\}$
- How to choose k_N? As discussed, there are two ways:
  1. Choose such that $g(k_N, z) < k_N$ for sure. Need a very wide grid and some side calculations.
     - Inefficiently large grid but safer choice to start
     - Cannot be used when changing parameters during estimation/calibration
  2. Choose a large k_N but be ready to extrapolate.. with lots of trepidation!
     - If you extrapolate more than $k_N - k_{N-1}$, your grid is too small.
- Reformulate the utility function so that the Bellman objective reads (recursive CES trick from Lecture 2):

$$\mathcal{V}(k, z) = \max_c \left[(1-\beta) c^{1-\gamma} + \beta \mathbb{E}\left(\mathcal{V}(k', z')^{1-\gamma} \right) \right]^{1/(1-\gamma)}.$$

Implied Parameters

- R is calibrated to get $\bar{k}/\bar{z} = 5$. As β is varied, so will R.
- Since natural borrowing limit depends on R, it will have to be adjusted too.

Table 1: Interest rates and Borrowing Constraint

		Interest rate		k_{min}		min(c)/mean(z)
ρ	β	2	10	2	10	2 & 10
0.9	0.95	0.050	0.030	−6.34	−10.48	0.212
	0.99	0.010	0.008	−32.56	−42.00	0.212
0.98	0.95	0.051	0.027	−6.29	−11.76	0.212
	0.99	0.009	0.005	−33.91	−61.42	0.212

Caution: Pay Attention to the Lower End

- Check if the imposed k_{min}, together with z_{min} imply a consumption level that is "too low".
- Define $\underline{c}_{max}$ as the maximum consumption feasible in the worst state: $\underline{c}_{max} = (Rk_{min} + z_{min})$.

Caution: Pay Attention to the Lower End

- Check if the imposed k_{min}, together with z_{min} imply a consumption level that is "too low".
- Define $\underline{c}_{\text{max}}$ as the maximum consumption feasible in the worst state: $\underline{c}_{\text{max}} = (Rk_{\text{min}} + z_{\text{min}})$.
- So when maximizing Bellman objective, we will be evaluating $U(c)$ at $c < \underline{c}_{\text{max}}$!
- For example, above $\underline{c}_{\text{max}}/z_{\text{min}} = 0.21$. If we normalize $\bar{z}$ to 1, and $U(c)$ is a CRRA with $\gamma = 10$, we have $U'(\underline{c}_{\text{max}})$ $-(0.21)^{-9}/9 \approx 140,000$.

Caution: Pay Attention to the Lower End

- Check if the imposed k_{min}, together with z_{min} imply a consumption level that is “too low”.
- Define $\underline{c}_{\text{max}}$ as the maximum consumption feasible in the worst state: $\underline{c}_{\text{max}} = (Rk_{\text{min}} + z_{\text{min}})$.
- So when maximizing Bellman objective, we will be evaluating $U(c)$ at $c < \underline{c}_{\text{max}}$!
- For example, above $\underline{c}_{\text{max}}/z_{\text{min}} = 0.21$. If we normalize $\bar{z}$ to 1, and $U(c)$ is a CRRA with $\gamma = 10$, we have $U'(\underline{c}_{\text{max}})$ $-(0.21)^{-9}/9 \approx 140,000$.
- Suppose we are not careful and search down to $c = 0.05$. Then $U'(0.05) \sim 10^{13}$. Every numerical computation as well as interpolation, etc., will be a challenge.
- So, before we test any lower c, check if $c = \underline{c}_{\text{max}}$ is a solution. Very often it will be.

Stopping Criterion: Issues to Consider

1. Using a tolerance based on levels is not a good idea, because:
    1. The numerical value of utility has no meaning (10^{-5},10^{-8}, etc) since it's an ordinal measure.
    2. Because U has wide range of variation, this imposes an "uneven" stringency condition across different parts of value function.

Stopping Criterion: Issues to Consider

1. Using a tolerance based on levels is not a good idea, because:
    1. The numerical value of utility has no meaning (10^{-5},10^{-8}, etc) since it's an ordinal measure.
    2. Because U has wide range of variation, this imposes an "uneven" stringency condition across different parts of value function.

2. Better to use a percentage-like deviation tolerance.
    1. Caution: when V is close to zero, this creates same problem as levels: values near zero imply huge percentage deviations.

Stopping Criterion: Issues to Consider

1. Using a tolerance based on levels is not a good idea, because:
   1. The numerical value of utility has no meaning (10^{-5},10^{-8}, etc) since it's an ordinal measure.
   2. Because U has wide range of variation, this imposes an "uneven" stringency condition across different parts of value function.

2. Better to use a percentage-like deviation tolerance.
   1. Caution: when V is close to zero, this creates same problem as levels: values near zero imply huge percentage deviations.
   2. Modified criterion:

$$\max_{i,j} \frac{\left|V^n(k_i, z_j) - V^{n-1}(k_i, z_j)\right|}{1 + |V^n(k_i, z_j)|} < 4 \times 10^{-7}, \quad (2)$$

where the $+1$ in the denominator avoids ratio blowing up.

Stopping Criterion: Issues to Consider

1. Using a tolerance based on levels is not a good idea, because:
    1. The numerical value of utility has no meaning (10^{-5},10^{-8}, etc) since it's an ordinal measure.
    2. Because U has wide range of variation, this imposes an "uneven" stringency condition across different parts of value function.

2. Better to use a percentage-like deviation tolerance.
    1. Caution: when V is close to zero, this creates same problem as levels: values near zero imply huge percentage deviations.
    2. Modified criterion:

$$\max_{i,j} \frac{\left|V^n(k_i, z_j) - V^{n-1}(k_i, z_j)\right|}{1 + \left|V^n(k_i, z_j)\right|} < 4 \times 10^{-7}, \tag{2}$$

where the $+1$ in the denominator avoids ratio blowing up.

3. Ideally, set criterion based on decision rule deviation. More on this later.

How To Ensure the Accuracy of Solution

Checking for Accuracy

1. Question: Is it sufficient to check the criterion at (i, j) grid points as written above?
   - No. We will see this clearly below.

Checking for Accuracy

1. Question: Is it sufficient to check the criterion at (i, j) grid points as written above?
   - No. We will see this clearly below.
2. Notice that the CES formulation avoids all of these problems at once: Always positive with small slope.

Checking for Accuracy

1. Question: Is it sufficient to check the criterion at (i, j) grid points as written above?
   - No. We will see this clearly below.
2. Notice that the CES formulation avoids all of these problems at once: Always positive with small slope.

- Preview of Main Lesson: *There is no silver bullet.* Every test has significant false positives and negatives (or Type I and Type II errors so to speak).
- That said: there is an essential checklist to go through. Collectively they will be informative.

Minimum Checklist

1. **Check for convergence:** Define a finer grid than the one used to solve the problem. Check for stopping criteria at all points (using same interpolation used in the solution).
    - We will see how critical this will be in a moment.

Minimum Checklist

1. **Check for convergence:** Define a finer grid than the one used to solve the problem. Check for stopping criteria at all points (using same interpolation used in the solution).
   - We will see how critical this will be in a moment.
2. **Plot, plot, plot:**

Minimum Checklist

1. **Check for convergence:** Define a finer grid than the one used to solve the problem. Check for stopping criteria at all points (using same interpolation used in the solution).
   - We will see how critical this will be in a moment.
2. **Plot, plot, plot:**
   1. Start with value function and decision rule at grid points. Some problems are immediately visible here (nonconcavity, wiggles, crossing into wrong sign, etc.)

Minimum Checklist

1. **Check for convergence:** Define a finer grid than the one used to solve the problem. Check for stopping criteria at all points (using same interpolation used in the solution).
   - We will see how critical this will be in a moment.
2. **Plot, plot, plot:**
   1. Start with value function and decision rule at grid points. Some problems are immediately visible here (nonconcavity, wiggles, crossing into wrong sign, etc.)
   2. Repeat (a) but plot over a much finer grid using the same interpolation method used in solution.

Minimum Checklist

1. **Check for convergence:** Define a finer grid than the one used to solve the problem. Check for stopping criteria at all points (using same interpolation used in the solution).
   - We will see how critical this will be in a moment.
2. **Plot, plot, plot:**
   1. Start with value function and decision rule at grid points. Some problems are immediately visible here (nonconcavity, wiggles, crossing into wrong sign, etc.)
   2. Repeat (a) but plot over a much finer grid using the same interpolation method used in solution.
   3. Scaling can obscure many faults. So (a) and (b) may not give any hints even when there are problems.
      1. Change scaling (very much underrated!): plot $\log(V)$ vs. k, and V vs. $\log(k)$, and $\log(V)$ vs. $\log(k)$ (when sign +).

Minimum Checklist

1. **Check for convergence:** Define a finer grid than the one used to solve the problem. Check for stopping criteria at all points (using same interpolation used in the solution).
   - We will see how critical this will be in a moment.
2. **Plot, plot, plot:**
   1. Start with value function and decision rule at grid points. Some problems are immediately visible here (nonconcavity, wiggles, crossing into wrong sign, etc.)
   2. Repeat (a) but plot over a much finer grid using the same interpolation method used in solution.
   3. Scaling can obscure many faults. So (a) and (b) may not give any hints even when there are problems.
      1. Change scaling (very much underrated!): plot $\log(V)$ vs. k, and V vs. $\log(k)$, and $\log(V)$ vs. $\log(k)$ (when sign +).
      2. Plot the derivative of the functions, which makes problems more visible.

Finally...

3. **Tighten all the screws** and solve the problem again:
   - Increase number of grid points by 3X, 5X, etc.
   - Tighten tolerance criteria (remember: more in inner loops than outer)
   - If you used a less accurate method, like linear interpolation, switch to spline and redo.

Finally...

3. **Tighten all the screws** and solve the problem again:
   - Increase number of grid points by 3X, 5X, etc.
   - Tighten tolerance criteria (remember: more in inner loops than outer)
   - If you used a less accurate method, like linear interpolation, switch to spline and redo.

- Then **simulate the more refined solution** and **check all statistics** you care about.

Finally...

3. **Tighten all the screws** and solve the problem again:
   - Increase number of grid points by 3X, 5X, etc.
   - Tighten tolerance criteria (remember: more in inner loops than outer)
   - If you used a less accurate method, like linear interpolation, switch to spline and redo.

- Then **simulate the more refined solution** and **check all statistics** you care about.
- If they haven't changed, then you can breathe a sigh of relief.

Finally...

3. **Tighten all the screws** and solve the problem again:
    - Increase number of grid points by 3X, 5X, etc.
    - Tighten tolerance criteria (remember: more in inner loops than outer)
    - If you used a less accurate method, like linear interpolation, switch to spline and redo.

- Then **simulate the more refined solution** and **check all statistics** you care about.
- If they haven't changed, then you can breathe a sigh of relief.
- N.B. In general, getting prices right requires much higher accuracy than quantities.

Algorithms: Comparing 4 Methods

Algorithms:

1. Plain-vanilla VFI
2. VFI + Modified Policy Iteration Algorithm (MPIA)
3. VFI + MacQueen-Porteus Bounds (MQP)
4. Endogenous Grid Method (EGM)

Checking for Accuracy: Example

- Use following parameters:
  - $\rho = 0.98$, $\beta = 0.99$.
  - Choose R as before. Set $k_{max} = 500$ (times average income).

Checking for Accuracy: Example

- Use following parameters:
  - $\rho = 0.98$, $\beta = 0.99$.
  - Choose R as before. Set $\mathrm{k}_{\text{max}} = 500$ (times average income).
- Test four sets of methods for solving this problem:
  - (i) CRRA specification, spline interpolation;
  - (ii) CRRA specification, linear interpolation,
  - (iii) CES specification, spline interpolation; and
  - (iv) CES specification, linear interpolation.

Checking for Accuracy: Example

- Use following parameters:
  - $\rho = 0.98$, $\beta = 0.99$.
  - Choose R as before. Set $\mathrm{k}_{\mathrm{max}} = 500$ (times average income).
- Test four sets of methods for solving this problem:
  - (i) CRRA specification, spline interpolation;
  - (ii) CRRA specification, linear interpolation,
  - (iii) CES specification, spline interpolation; and
  - (iv) CES specification, linear interpolation.
- Take a 20-point grid for k and $\theta = 3$. If convergence fails on this grid $\rightarrow$ increase grid points by 20 until convergence.

Results

Table 2: Convergence Time of VFI Algorithm, Four Methods

Method:	RRA	
	2	10
CES-spline	8.6	15.9
CES-linear	7.8	14.2
CRRA-spline	12.7	55.1*
CRRA-linear	14.8	20.4

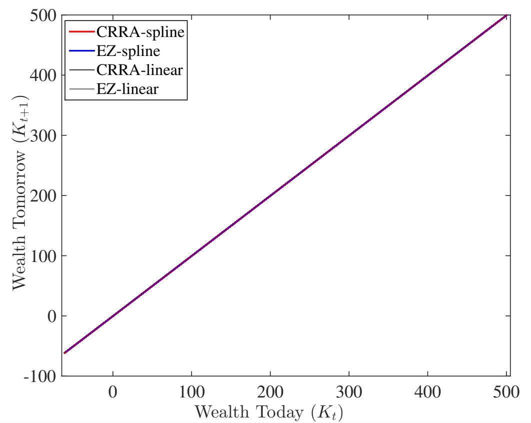
*Note: The baseline number of grid points is 20. This looks extremely low and it is. But we will see how it yields very accurate solutions if the right methods are applied. If convergence is not obtained for this specification we increase grid points by 20 up to 100. *60 point grid was needed. RRA is coded as integer (not real) for efficiency.*

How did CRRA-linear pull off this feat?

- If it sounds too good to be true, it usually is.
- But, the convergence criteria was satisfied to 10^{-7}??
  - Yes, but on the 20 grid points. We didn't check off grid points!
- Suppose, we don't notice that. Let us plot the decision rule.

Plot decision rule: Version 1, RRA=10

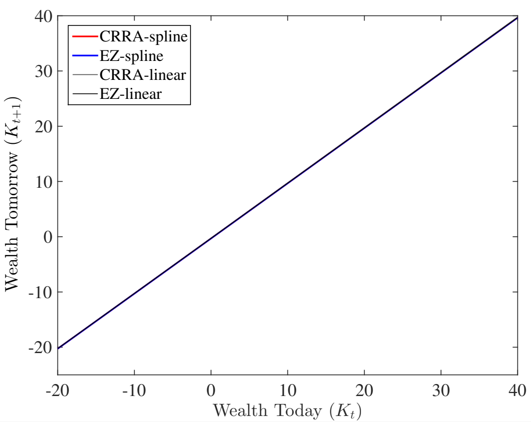
Figure 1: Savings Decision Generated by Four Solutions of the Same Model



- The first reaction to this figure is often that they all look the same. Nothing interesting.. or is there?

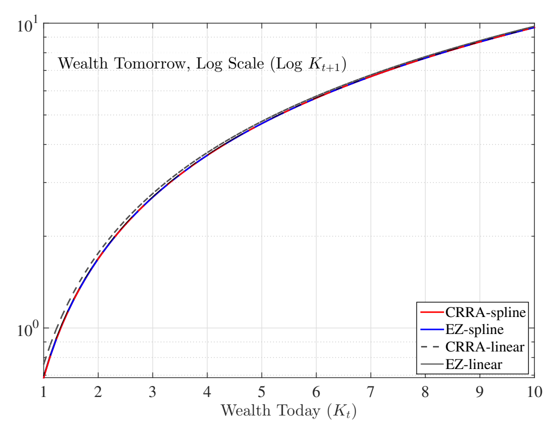
Plot: Version 2, zoom in, RRA=2

Figure 2: Savings Decision Generated by Four Solutions of the Same Model



- Let's look at the easier case, RRA=2. Zoom in by shrinking x-axis range. Still nothing..

Plot: Version 3, Zoom in Again, log(K')



- Zoom further in to $k_t \in [1, 10]$ and plot log of savings.. A little gap appears at the low end. This gap will increase further at lower k values near limit.

Does this deviation matter? Statistics from Simulation

	Mean	Std. Dev.	Max	Min
	Capital			
CES-S	11.26	28.81	100.24	–33.912
CRRA-S	11.18	28.77	99.76	–33.912
CES-L	13.34	28.90	101.86	–33.912
CRRA-L	**31.83**	*28.66*	**80.78**	–33.912

Does this deviation matter? Statistics from Simulation

	Mean	Std. Dev.	Max	Min
	Capital			
CES-S	11.26	28.81	100.24	−33.912
CRRA-S	11.18	28.77	99.76	−33.912
CES-L	13.34	28.90	101.86	−33.912
CRRA-L	**31.83**	*28.66*	**80.78**	−33.912
	Consumption			
CES-S	1.14	0.325	2.16	0.213
CRRA-S	1.13	0.325	2.16	0.213
CES-L	1.15	0.326	2.17	0.213
CRRA-L	**1.33**	*0.326*	*2.14*	0.213

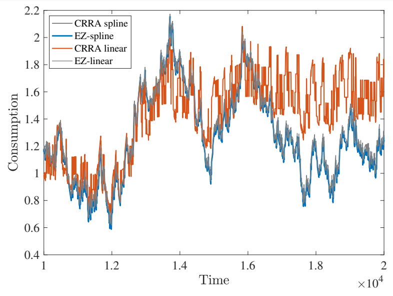
Does this deviation matter? Statistics from Simulation

	Mean	Std. Dev.	Max	Min
	Capital			
CES-S	11.26	28.81	100.24	−33.912
CRRA-S	11.18	28.77	99.76	−33.912
CES-L	13.34	28.90	101.86	−33.912
CRRA-L	**31.83**	*28.66*	**80.78**	−33.912
	Consumption			
CES-S	1.14	0.325	2.16	0.213
CRRA-S	1.13	0.325	2.16	0.213
CES-L	1.15	0.326	2.17	0.213
CRRA-L	**1.33**	*0.326*	*2.14*	0.213
	$K_{t+1} - K_t$			
CES-S	0.001	0.132	0.611	−0.324
CRRA-S	0.001	0.132	0.613	−0.324
CES-L	0.001	0.130	0.609	−0.318
CRRA-L	*0.001*	**0.075**	**0.566**	**−0.233**

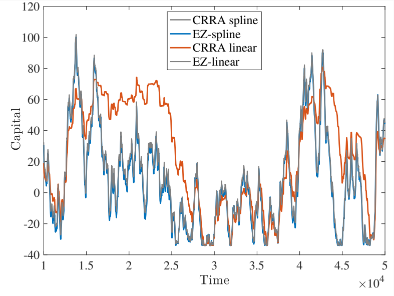
That Doesn't Look Good!

Figure 3: Simulated Paths Generated by Four Solutions Methods, RRA=2

(a) Consumption Path



(b) Capital Path



- CRRA-Linear is is way off for thousands of periods at a time. Which is why it generates very different results for some statistics.
- CES-L (marked EZ) is a little off in extremes but hard to even see. With 40-point grid, all stats within 1% of more accurate CES-spline.

Wrap Up

- Verifying the accuracy of the solution typically requires careful detective work.
  - Checking the convergence criterion is the first of many steps!
  - Plotting the value function and decision rules is essential (esp. scaling up, plotting derivatives, etc.). It is necessary and can reveal very obvious problems but it is nowhere near sufficient.

Wrap Up

- Verifying the accuracy of the solution typically requires careful detective work.
  - Checking the convergence criterion is the first of many steps!
  - Plotting the value function and decision rules is essential (esp. scaling up, plotting derivatives, etc.). It is necessary and can reveal very obvious problems but it is nowhere near sufficient.
- Solve the model again on a much finer grid, with tighter tolerance (for maximization, integration, interpolation, etc) is a must.

Wrap Up

- Verifying the accuracy of the solution typically requires careful detective work.
  - Checking the convergence criterion is the first of many steps!
  - Plotting the value function and decision rules is essential (esp. scaling up, plotting derivatives, etc.). It is necessary and can reveal very obvious problems but it is nowhere near sufficient.
- Solve the model again on a much finer grid, with tighter tolerance (for maximization, integration, interpolation, etc) is a must.
- Then simulate all key variables and compare the statistics that are critical for your work.

Wrap Up

- Verifying the accuracy of the solution typically requires careful detective work.
  - Checking the convergence criterion is the first of many steps!
  - Plotting the value function and decision rules is essential (esp. scaling up, plotting derivatives, etc.). It is necessary and can reveal very obvious problems but it is nowhere near sufficient.
- Solve the model again on a much finer grid, with tighter tolerance (for maximization, integration, interpolation, etc) is a must.
- Then simulate all key variables and compare the statistics that are critical for your work.
- It is typically much harder to compute prices accurately than quantities! We will see examples of this in lectures 6 and 7.