Lecture 8: GE with Heterogeneity (No Aggregate Shocks)

Fatih Guvenen
University of Minnesota

December 2023

Bewley-Huggett-Aiyagari Models

Aiyagari (1994)

- 3,500 citations since 1994! It is the workhorse model in the incomplete markets literature.

Aiyagari (1994)

- 3,500 citations since 1994! It is the workhorse model in the incomplete markets literature.
- Individual's Problem:

$$\begin{aligned} & E_0 \left\{ \sum_{t=0}^{\infty} \beta^t U(c_t) \right\} \\ & \text{s.t} \\ c_t + a_{t+1} &= w l_t + (1+r) a_t \\ c_t &\geq 0, \; a_t \geq -b \\ & l_t \text{ is a stochastic w/ bdd support} \end{aligned}$$

b: natural limit $(w l_{min}/r)$ or an ad hoc one.

Aiyagari (1994)

- 3,500 citations since 1994! It is the workhorse model in the incomplete markets literature.
- Individual's Problem:

$$
\begin{aligned}
& E_0 \left\{ \sum_{t=0}^{\infty} \beta^t U(c_t) \right\} \\
& \text{s.t} \\
c_t + a_{t+1} &= wl_t + (1+r)\, a_t \\
c_t &\geq 0,\; a_t \geq -b \\
& l_t \text{ is a stochastic w/ bdd support}
\end{aligned}
$$

  b: natural limit (wl_{min}/r) or an ad hoc one.
- Competitive firm: $r = \alpha(K/L)^{(\alpha-1)}$ and $w = (1-\alpha)(K/L)^{-\alpha}$.

Aiyagari (1994)

- 3,500 citations since 1994! It is the workhorse model in the incomplete markets literature.
- Individual's Problem:

$$
\begin{aligned}
& E_0\left\{\sum_{t=0}^{\infty}\beta^t U\left(c_t\right)\right\} \\
& \text{s.t} \\
c_t + a_{t+1} &= wl_t + (1+r)\,a_t \\
c_t &\geq 0,\; a_t \geq -b \\
& l_t \text{ is a stochastic w/ bdd support}
\end{aligned}
$$

  b: natural limit (wl_{min}/r) or an ad hoc one.

- Competitive firm: $r = \alpha(K/L)^{(\alpha-1)}$ and $w = (1-\alpha)(K/L)^{-\alpha}$.
- Solve for K^*, L^*, implies r^*, w^*, and $\Gamma(a)$: wealth distribution.

Aiyagari (1994)

- Define:

$$\widehat{a}_t \equiv a_t + \phi$$
$$z_t = wl_t + (1+r)\,\widehat{a}_t - r\phi$$

- Asset demand is: $\widehat{a}_{t+1} = A(z_t, b, w, r)$:

Figure 1: Aiyagari (1994, QJE)

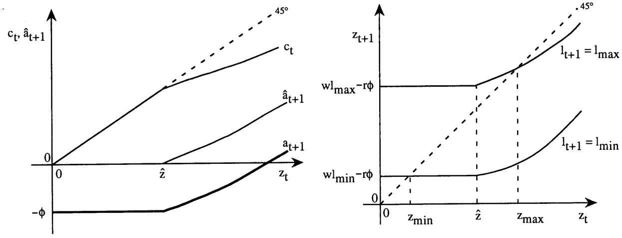


FIGURE Ia
Consumption and Assets as Functions of Total Resources

FIGURE Ib
Evolution of Total Resources

Concavity of the Consumption Function

- Although it may not be clear from this hand drawn figure, the consumption function is concave under very general conditions.
- In fact, when preferences are from HARA class and display "prudence", it is strictly concave (Carroll and Kimball (ECMA, 1996)).
- HARA class are preferences with "hyperbolic absolute risk aversion" or "linear absolute risk tolerance": $-\frac{U'(c)}{U''(c)} = a + bc.$
- Some key theorems on aggregation depend on HARA preferences so it's good to be familiar with them.
- CRRA, CARA, and quadratic utility are special cases of HARA. Only the first two display prudence: $\frac{U'''U'}{(U'')^2} > 0.$

General equilibrium: r* and K*

Figure 2: Aiyagari (1994, QJE)

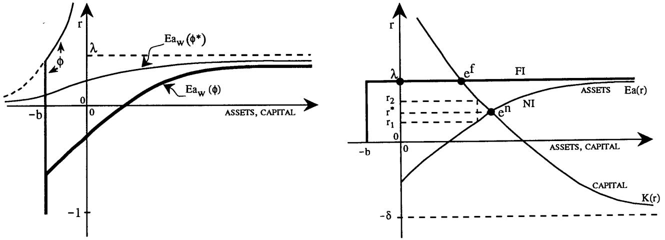


FIGURE IIa
Interest Rate versus Per Capita Assets

FIGURE IIb
Steady-State Determination

- Notice that when $r = \lambda$ long-run asset demand goes to infinity.

Comments on Aiyagari

- What do we learn from Aiyagari?
- Two key substantive results:

Comments on Aiyagari

- What do we learn from Aiyagari?
- Two key substantive results:
  - With incomplete markets, K^* is higher than under complete mkts (but typically not by much)

Comments on Aiyagari

- What do we learn from Aiyagari?
- Two key substantive results:
  - With incomplete markets, K^* is higher than under complete mkts (but typically not by much)
  - And r^* is lower (early papers tried to explain the equity premium puzzle by this).

Comments on Aiyagari

- What do we learn from Aiyagari?
- Two key substantive results:
  - With incomplete markets, K^* is higher than under complete mkts (but typically not by much)
  - And r^* is lower (early papers tried to explain the equity premium puzzle by this).
  - Figure IIb very useful for other incomplete mkts models too (e.g., Krusell-Smith (1998) stochastic-beta, Guvenen (2006) limited participation & Cagetti-De Nardi (2006) entrepreneurship, or Laitner (2002) bequest, models.

Comments on Aiyagari

- What do we learn from Aiyagari?
- Two key substantive results:
  - With incomplete markets, K^* is higher than under complete mkts (but typically not by much)
  - And r^* is lower (early papers tried to explain the equity premium puzzle by this).
  - Figure IIb very useful for other incomplete mkts models too (e.g., Krusell-Smith (1998) stochastic-beta, Guvenen (2006) limited participation & Cagetti-De Nardi (2006) entrepreneurship, or Laitner (2002) bequest, models.
- Methodological: He showed in detail how to solve these models.

Solving Aiyagari-style Models: Two Approaches

Two ways to solve these models:

Solving Aiyagari-style Models: Two Approaches

Two ways to solve these models:

1. In time series: by simulating the model
   - Pros: Conceptually easier and simpler to code
   - Cons: Often slower and less accurate (sometimes much slower)

Solving Aiyagari-style Models: Two Approaches

Two ways to solve these models:

1. In time series: by simulating the model
   - Pros: Conceptually easier and simpler to code
   - Cons: Often slower and less accurate (sometimes much slower)
2. In state space: using model's transition equation(s). No simulation
   - Cons: Could be a bit harder to wrap your head around it.
   - Pros: Can be much faster and more accurate.

Solution Algorithm by Simulation (Simple Version)

Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
   - We already know how to do this.

Solution Algorithm by Simulation (Simple Version)

Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
   - We already know how to do this.
2. Simulation:
   1. Starting from an initial distribution μ^0, simulate a single long time series of asset values: $\{a_t\}_{t=1}^T$.

Solution Algorithm by Simulation (Simple Version)

Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
   - We already know how to do this.
2. Simulation:
   1. Starting from an initial distribution μ^0, simulate a single long time series of asset values: $\{a_t\}_{t=1}^{T}$.
   2. Discard the first T^0 periods ($0 \ll T^0 \ll T$) and use $\{a_t\}_{t=T^0+1}^{T}$, to calculate the implied aggregate asset supply $K_r^S = (\frac{1}{T-T^0}) \sum_{t=T^0+1}^{T} a_t$.

Solution Algorithm by Simulation (Simple Version)

Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
   - We already know how to do this.

2. Simulation:
   1. Starting from an initial distribution μ^0, simulate a single long time series of asset values: $\{a_t\}_{t=1}^{T}$.
   2. Discard the first T^0 periods ($0 \ll T^0 \ll T$) and use $\{a_t\}_{t=T^0+1}^{T}$, to calculate the implied aggregate asset supply $K_r^S = (\frac{1}{T-T^0}) \sum_{t=T^0+1}^{T} a_t$.
   3. Check if $K_r^D = K_r^S$. If so, we have a steady state equilibrium and $K^* = K_r^D$. If $K_r^D \lessgtr K_r^S$, reduce/increase K_r^D and go back to step 1. Iterate until convergence.

Comments

- Notice that we are not simulating a panel of individuals, but a single time series.
- And in step 3, we are taking the time series average (instead of cross-sectional) to find steady state K^s.

Comments

- Notice that we are not simulating a panel of individuals, but a single time series.
- And in step 3, we are taking the time series average (instead of cross-sectional) to find steady state K^s.
- Why are we allowed to do that?
  - Because the model's solution is ergodic for the mean of capital.

Comments

- Notice that we are not simulating a panel of individuals, but a single time series.
- And in step 3, we are taking the time series average (instead of cross-sectional) to find steady state K^s.
- Why are we allowed to do that?
  - Because the model's solution is ergodic for the mean of capital.
- If, for example, the income process had a fixed effect we couldn't do that.

Comments

- Notice that we are not simulating a panel of individuals, but a single time series.
- And in step 3, we are taking the time series average (instead of cross-sectional) to find steady state K^s.
- Why are we allowed to do that?
  - Because the model's solution is ergodic for the mean of capital.
- If, for example, the income process had a fixed effect we couldn't do that.
- So, steps 2 and 3 would have to be replaced with simulating an $N \times T$ panel and taking an average over N once the model reaches steady state.

Comments

- Notice that we are not simulating a panel of individuals, but a single time series.
- And in step 3, we are taking the time series average (instead of cross-sectional) to find steady state K^s.
- Why are we allowed to do that?
  - Because the model's solution is ergodic for the mean of capital.
- If, for example, the income process had a fixed effect we couldn't do that.
- So, steps 2 and 3 would have to be replaced with simulating an $N \times T$ panel and taking an average over N once the model reaches steady state.
- Moreover, we have not checked if the distribution has converged, just checked if the mean has converged!

Solution Algorithm by Simulation: Panel Version

(Again) Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$

Solution Algorithm by Simulation: Panel Version

(Again) Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
2. Simulation:
    1. Starting from an initial distribution Γ^0, simulate an $N \times T$ $(N, T \gg 0)$ panel of asset values: $\{a_{i,t}\}_{i=1,t=1}^{N,T}$.

Solution Algorithm by Simulation: Panel Version

(Again) Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
2. Simulation:
   1. Starting from an initial distribution Γ^0, simulate an $N \times T$ $(N, T \gg 0)$ panel of asset values: $\{a_{i,t}\}_{i=1,t=1}^{N,T}$.
   2. Use the last cross section,$\{a_t\}_{i=1,t=T}^{N,T}$, to calculate the implied aggregate asset supply $K_r^S = (\frac{1}{N}) \sum_{i=1}^{N} a_{i,T}$.

Solution Algorithm by Simulation: Panel Version

(Again) Two Steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
2. Simulation:
   1. Starting from an initial distribution Γ^0, simulate an $N \times T$ $(N, T \gg 0)$ panel of asset values: $\{a_{i,t}\}_{i=1,t=1}^{N,T}$.
   2. Use the last cross section, $\{a_t\}_{i=1,t=T}^{N,T}$, to calculate the implied aggregate asset supply $K_r^S = (\frac{1}{N}) \sum_{i=1}^{N} a_{i,T}$.
   3. Check if $K_r^D = K_r^S$. Also check if $\mu_{T-1} = \mu_T$. If so, we have a steady state equilibrium and $K^* = K_r^D$ and $\mu^* = \mu_T$.
      1. If not and $K_r^D \lessgtr K_r^S$ reduce/increase K_r^D and go back to step 1. Iterate until convergence.
      2. If $\mu_{T-1} \neq \mu_T$, increase simulation length T until the distribution also converges.

Notes

- T here can be much smaller than the T when we simulate a single time series.
- It is more comparable to the T^0 in that context—we require the model to have converged to a steady state, nothing more. Think about 1,000 periods or so minimum though.
- It is a good idea to calculate K_r^s for more than one cross section and compare to ensure they are not moving over time. If the solution has truly converged to a steady state, the mean should remain (virtually) constant (say move less than 0.1%)
- In the last step, to check if the distribution has converged, you can compute a histogram with 20 to 50 equal size bins (or more finely at the top if it matters more for your problem) and calculate the fraction of individuals in each bin in periods $T - 1$ and T and make sure the fractions are (virtually) the same.

Solution Algorithm in State Space

Two steps:

1. Solve the consumption-savings problem for a given K_r^D level (r and w implied) and obtain $a_{t+1}(a_t, z_t; r, w)$
2. Solve for Steady State Equilibrium:
   1. Using decision rules, write the law of motion for the wealth distribution:
   $$\mu_{n+1}(\mathcal{A}, \mathcal{Z}; K_r^D) = \int \mathcal{Q}_r^D((a, z), (\mathcal{A}, \mathcal{Z}))\mu_n(da, dz; K_r^D)$$
   Take an initial guess for $\mu_{n=0}$ and iterate on this mapping N times.
   2. Calculate aggregate asset supply $K_r^S = \int a_{t+1}(a, z) d\mu_N(da, dz; K_r^D)$
   3. Check if $K_r^D = K_r^S$ and $\mu_{N-1} = \mu_N$. If yes, we have a steady state equilibrium and $K^* = K_r^D$ and $\mu^* = \mu_N(a, z, K_r^D)$.
      1. If $K_r^D \lessgtr K_r^S$ reduce/increase K_r^D and go back to step 1.
      2. If $\mu_{T-1} \neq \mu_T$, increase iteration length N until the distribution also converges.

1. How To Simulate? Generating (pseudo-) random numbers

- We often need to simulate a large number of individual histories for thousands of periods.

1. How To Simulate? Generating (pseudo-) random numbers

- We often need to simulate a large number of individual histories for thousands of periods.
- Generating (pseudo-) random numbers:
  - Very often we will want to simulate the exact same sequence of random numbers in repeated simulations.
  - This is the case when we solve a model via simulation, Aiyagari, Krusell-Smith, etc.
  - Also the case when we do any kind of simulation based estimation.
  - You can do this by using the same “seed” of the RN generator in subsequent simulations.

1. Generating (pseudo-) random numbers, cont'd

- Not all random number generators are created equal.

1. Generating (pseudo-) random numbers, cont'd

- Not all random number generators are created equal.
- Some are very slow. Others are not "random" enough. Some are not "portable": they will generate different sequences on different machines (so you cannot replicate your work elsewhere).
  - Just to give you an idea: wikipedia.org/wiki/List_of_random_number_generators

1. Generating (pseudo-) random numbers, cont'd

- Not all random number generators are created equal.
- Some are very slow. Others are not "random" enough. Some are not "portable": they will generate different sequences on different machines (so you cannot replicate your work elsewhere).
  - Just to give you an idea: wikipedia.org/wiki/List_of_random_number_generators
  - Make sure you know what your RN generator is.
  - I use **`ran2.f`** and **`ran3.f`** from NR. Lots of good properties.

1. Generating (pseudo-) random numbers, cont'd

- Not all random number generators are created equal.
- Some are very slow. Others are not "random" enough. Some are not "portable": they will generate different sequences on different machines (so you cannot replicate your work elsewhere).
  - Just to give you an idea: wikipedia.org/wiki/List_of_random_number_generators
  - Make sure you know what your RN generator is.
  - I use **`ran2.f`** and **`ran3.f`** from NR. Lots of good properties.
- Drawing random variables is not costless. Should you redraw every time or draw once and keep in RAM?

1. Generating (pseudo-) random numbers, cont'd

- Not all random number generators are created equal.
- Some are very slow. Others are not "random" enough. Some are not "portable": they will generate different sequences on different machines (so you cannot replicate your work elsewhere).
  - Just to give you an idea: wikipedia.org/wiki/List_of_random_number_generators
  - Make sure you know what your RN generator is.
  - I use **`ran2.f`** and **`ran3.f`** from NR. Lots of good properties.
- Drawing random variables is not costless. Should you redraw every time or draw once and keep in RAM?
- Depends:
  - #of RN numbers to be generated
  - speed of RN generator
  - size of RAM
  - speed of disk/read write.

1. Generating (pseudo-) random numbers, cont'd

- Do a back-of-the envelope calculation first:

1. Generating (pseudo-) random numbers, cont'd

- Do a back-of-the envelope calculation first:
  - Simulating N individuals for T years, with 1 shock per person/year
  - E.g., Double precision RN, $\mathrm{N} = 50,000$ and $\mathrm{T} = 1,000$, $\mathrm{N} \times \mathrm{T} \times 8$ bytes: $= 4 \times 10^8$ bytes or 400 Megabytes.

1. Generating (pseudo-) random numbers, cont'd

- Do a back-of-the envelope calculation first:
  - Simulating N individuals for T years, with 1 shock per person/year
  - E.g., Double precision RN, $\mathrm{N} = 50,000$ and $\mathrm{T} = 1,000$, $\mathrm{N} \times \mathrm{T} \times 8$ bytes: $= 4 \times 10^8$ bytes or 400 Megabytes.
  - Instead simulate, $\mathrm{T} = 10,000$: 4 Gigabytes!
  - This is a huge file. Will take lots of place in RAM and can be very slow to access.

1. Generating (pseudo-) random numbers, cont'd

- Do a back-of-the envelope calculation first:
  - Simulating N individuals for T years, with 1 shock per person/year
  - E.g., Double precision RN, $\mathrm{N} = 50,000$ and $\mathrm{T} = 1,000$, $\mathrm{N} \times \mathrm{T} \times 8$ bytes: $= 4 \times 10^8$ bytes or 400 Megabytes.
  - Instead simulate, $\mathrm{T} = 10,000$: 4 Gigabytes!
  - This is a huge file. Will take lots of place in RAM and can be very slow to access.
- Alternatives:
  - Draw RNs period by period, and discard once you do all the calculations for that period (e.g., calculate savings, consumption, etc. for each individual).

1. Generating (pseudo-) random numbers, cont'd

- Do a back-of-the envelope calculation first:
  - Simulating N individuals for T years, with 1 shock per person/year
  - E.g., Double precision RN, $N = 50,000$ and $T = 1,000$, $N \times T \times 8$ bytes: $= 4 \times 10^8$ bytes or 400 Megabytes.
  - Instead simulate, $T = 10,000$: 4 Gigabytes!
  - This is a huge file. Will take lots of place in RAM and can be very slow to access.
- Alternatives:
  - Draw RNs period by period, and discard once you do all the calculations for that period (e.g., calculate savings, consumption, etc. for each individual).
  - Simulate all $N \times T$ RNs at once. Split T into, e.g., 20, and store them as 20 files.

1. Generating (pseudo-) random numbers, cont'd

- Do a back-of-the envelope calculation first:
  - Simulating N individuals for T years, with 1 shock per person/year
  - E.g., Double precision RN, $N = 50,000$ and $T = 1,000$, $N \times T \times 8$ bytes: $= 4 \times 10^8$ bytes or 400 Megabytes.
  - Instead simulate, $T = 10,000$: 4 Gigabytes!
  - This is a huge file. Will take lots of place in RAM and can be very slow to access.
- Alternatives:
  - Draw RNs period by period, and discard once you do all the calculations for that period (e.g., calculate savings, consumption, etc. for each individual).
  - Simulate all $N \times T$ RNs at once. Split T into, e.g., 20, and store them as 20 files.
- Bottom line:
  - You must always "profile" your code. You will often be surprised where your code spends most of its time. You can then reoptimize to speed up that part.

1. How to Store Simulated Data

- Same back of the envelope calculation but for model variables

1. How to Store Simulated Data

- Same back of the envelope calculation but for model variables
  - Now suppose, you generate individual consumption, labor supply, wealth for each person/year

1. How to Store Simulated Data

- Same back of the envelope calculation but for model variables
  - Now suppose, you generate individual consumption, labor supply, wealth for each person/year
  - Same N and $\mathrm{T} = 100,000$. If you do double precision reals, each file will be 5GB, so you gave 15 GB of simulated data in RAM or SSD.

1. How to Store Simulated Data

- Same back of the envelope calculation but for model variables
  - Now suppose, you generate individual consumption, labor supply, wealth for each person/year
  - Same N and $T = 100,000$. If you do double precision reals, each file will be 5GB, so you gave 15 GB of simulated data in RAM or SSD.
- Again you can either save data period by period. Or chop off data into smaller T subperiods.

1. How to Store Simulated Data

- Same back of the envelope calculation but for model variables
  - Now suppose, you generate individual consumption, labor supply, wealth for each person/year
  - Same N and $\mathrm{T} = 100,000$. If you do double precision reals, each file will be 5GB, so you gave 15 GB of simulated data in RAM or SSD.
- Again you can either save data period by period. Or chop off data into smaller T subperiods.
- Another alternative: Do all calculations and save next period's state variables in double precision. Then save current period's variables in single precision. Half the file size.

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.
- Even when they match the Gini and top 1% share of wealth it misses several things:

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.
- Even when they match the Gini and top 1% share of wealth it misses several things:
  - It takes an extremely long time to get to steady state where such inequality exists. Several hundreds years or more, or dozens of generations.

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.
- Even when they match the Gini and top 1% share of wealth it misses several things:
  - It takes an extremely long time to get to steady state where such inequality exists. Several hundreds years or more, or dozens of generations.
    - But in the data, many super wealthy are self made: e.g., 54% of Forbes 400 billionaires.

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.
- Even when they match the Gini and top 1% share of wealth it misses several things:
  - It takes an extremely long time to get to steady state where such inequality exists. Several hundreds years or more, or dozens of generations.
    - But in the data, many super wealthy are self made: e.g., 54% of Forbes 400 billionaires.
    - Evidence on bequests inconsistent with transmission of such large wealth (e.g. Kopczuk's survey)

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.
- Even when they match the Gini and top 1% share of wealth it misses several things:
  - It takes an extremely long time to get to steady state where such inequality exists. Several hundreds years or more, or dozens of generations.
    - But in the data, many super wealthy are self made: e.g., 54% of Forbes 400 billionaires.
    - Evidence on bequests inconsistent with transmission of such large wealth (e.g. Kopczuk's survey)
  - Income shocks needed to generate top 1% and Gini is unrealistically large.

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.
- Even when they match the Gini and top 1% share of wealth it misses several things:
  - It takes an extremely long time to get to steady state where such inequality exists. Several hundreds years or more, or dozens of generations.
    - But in the data, many super wealthy are self made: e.g., 54% of Forbes 400 billionaires.
    - Evidence on bequests inconsistent with transmission of such large wealth (e.g. Kopczuk's survey)
  - Income shocks needed to generate top 1% and Gini is unrealistically large.
  - Even when top 1% is matched, nobody in simulated data has more than $20M or so.

Comments: Post-Aiyagari Models

- Models building on Aiyagari framework face some common challenges.
- Even when they match the Gini and top 1% share of wealth it misses several things:
  - It takes an extremely long time to get to steady state where such inequality exists. Several hundreds years or more, or dozens of generations.
    - But in the data, many super wealthy are self made: e.g., 54% of Forbes 400 billionaires.
    - Evidence on bequests inconsistent with transmission of such large wealth (e.g. Kopczuk's survey)
  - Income shocks needed to generate top 1% and Gini is unrealistically large.
  - Even when top 1% is matched, nobody in simulated data has more than $20M or so.
  - Most very wealthy do not work for wages. They are entrepreneurs.

How Much Inequality in Aiyagari-Style Models?

	U.S. Data	Gaussian	GKOS benchmark
Parametrization:		$\rho = 0.985, \sigma^2 = 0.0234$	Rich process
Gini	0.85	0.58	0.66
Top 0.1%	**14.8%**	**1.1%**	**2.2%**
Frac > $10M	**0.4–0.5%**	≈ 0	**0.02%**
Top 1%	35.5%	7.0%	9.2%
Top 10%	75.0%	37.9%	41.6%
Top 20%	87.0%	48.2%	52.8%

Back

POWER LAW MODELS

What is a Power Law?

- General definition: A power law (PL) is defined as a relationship between two variables, x and y, where:

$$y = a \times x^{-\alpha}, \tag{1}$$

for some scaling constant k.

  - α : is the power law exponent, and is a key parameter.

What is a Power Law?

- General definition: A power law (PL) is defined as a relationship between two variables, x and y, where:

$$\mathrm{y} = \mathrm{a} \times \mathrm{x}^{-\alpha}, \tag{1}$$

  for some scaling constant k.

  - α : is the power law exponent, and is a key parameter.

- Eq. (1) implies:

$$\log \mathrm{y} = -\alpha \times \log \mathrm{x},$$

  so a log-log plot of y and x should be a straight line with slope α, which allows us to see a power law visually (without fitting equation (1)).

Pareto Distribution

- Let w be a random variable whose distribution obeys the relationship:

$$P(w > x) = a \times x^{-\alpha}$$

where $P(w > x)$ is the counter-CDF of w, for some a and a positive α.

  - $\rightarrow$ w follows a PL or, alternatively, w has a Pareto distribution.

Pareto Distribution

- Let w be a random variable whose distribution obeys the relationship:

$$P(w > x) = a \times x^{-\alpha}$$

  where $P(w > x)$ is the counter-CDF of w, for some a and a positive α.
  - $\rightarrow w$ follows a PL or, alternatively, w has a Pareto distribution.
- Asymptotic Power Law:
  - Sometimes the power law holds only above a threshold, $\bar{x}$. In that case, w is said to be asymptotically PL.

Pareto Distribution

- Let w be a random variable whose distribution obeys the relationship:

$$P(w > x) = a \times x^{-\alpha}$$

  where $P(w > x)$ is the counter-CDF of w, for some a and a positive α.
  - $\rightarrow$ w follows a PL or, alternatively, w has a Pareto distribution.
- Asymptotic Power Law:
  - Sometimes the power law holds only above a threshold, $\bar{x}$. In that case, w is said to be asymptotically PL.
- PLs are pervasive in nature, including in many distributions social scientists are interested in.

Pareto Distribution

- Let w be a random variable whose distribution obeys the relationship:

$$P(w > x) = a \times x^{-\alpha}$$

where $P(w > x)$ is the counter-CDF of w, for some a and a positive α.

  - $\rightarrow w$ follows a PL or, alternatively, w has a Pareto distribution.
- Asymptotic Power Law:
  - Sometimes the power law holds only above a threshold, $\bar{x}$. In that case, w is said to be asymptotically PL.
- PLs are pervasive in nature, including in many distributions social scientists are interested in.
- Crucial property: a PL has finite moments only up to the α^{th} moment.
  - If $\alpha = 1$ (called Zipf's law): mean does not exist!
  - if $\alpha = 1.5$, variance does not exist.

Simple(st) Framework

- Proportional random growth model:

$$s_{t+1} = \alpha_t s_t \tag{2}$$

where α_t is a positive i.i.d random variable.

Simple(st) Framework

- Proportional random growth model:

$$s_{t+1} = \alpha_t s_t \tag{2}$$

  where α_t is a positive i.i.d random variable.

- Which yields:

$$\log s_t = \log s_0 + \Sigma_{n=1}^{t-1} \alpha_n$$

Simple(st) Framework

- Proportional random growth model:

$$s_{t+1} = \alpha_t s_t \tag{2}$$

where α_t is a positive i.i.d random variable.

- Which yields:

$$\log s_t = \log s_0 + \Sigma_{n=1}^{t-1} \alpha_n$$

- Two results:
  1. Assuming α_t has a well-behaved distribution so that the central limit theorem applies, $\frac{1}{t} s_t$ converges to a log-normal distribution.
  2. The distribution of s_t spreads without bound, so it has no stationary distribution (over time).

Power Law Models: Simple Mechanics

- W_t^i :wealth of individual i and $\overline{W}_t$: average (per-capita) wealth.

Power Law Models: Simple Mechanics

- W_t^i :wealth of individual i and $\overline{W}_t$: average (per-capita) wealth.
- Since wealth grows over time, define: $w_t^i \equiv \frac{W_t^i}{\overline{W}_t}$.

Power Law Models: Simple Mechanics

- W_t^i :wealth of individual i and $\overline{W}_t$: average (per-capita) wealth.
- Since wealth grows over time, define: $w_t^i \equiv \frac{W_t^i}{\overline{W}_t}$.
- Proportional random growth: $w_{t+1}^i = A_t^i \times w_{t^i}$, where A_t^i is i.i.d over t and i, with density $f(A)$.

Power Law Models: Simple Mechanics

- W_t^i :wealth of individual i and $\overline{W}_t$: average (per-capita) wealth.
- Since wealth grows over time, define: $w_t^i \equiv \frac{W_t^i}{\overline{W}_t}$.
- Proportional random growth: $w_{t+1}^i = A_t^i \times w_{t^i}$, where A_t^i is i.i.d over t and i, with density $f(A)$.

Power Law Models: Simple Mechanics

- W_t^i :wealth of individual i and $\overline{W}_t$: average (per-capita) wealth.
- Since wealth grows over time, define: $w_t^i \equiv \frac{W_t^i}{\overline{W}_t}$.
- Proportional random growth: $w_{t+1}^i = A_t^i \times w_{t^i}$, where A_t^i is i.i.d over t and i, with density $f(A)$.
- Define: $G_t(w_t) := P(w_t > x)$ as the counter-CDF, which evolves:

$$G_{t+1}(x) = P(w_{t+1}^i > x) = P(A_t^i w_t^i > x) > P(w_t^i > \frac{x}{A_t^i})$$
$$= \int_0^\infty G_t(\frac{x}{A})f(A)dA.$$

Power Law Models: Simple Mechanics

- W_t^i :wealth of individual i and $\overline{W}_t$: average (per-capita) wealth.
- Since wealth grows over time, define: $w_t^i \equiv \frac{W_t^i}{\overline{W}_t}$.
- Proportional random growth: $w_{t+1}^i = A_t^i \times w_{t^i}$, where A_t^i is i.i.d over t and i, with density $f(A)$.
- Define: $G_t(w_t) := P(w_t > x)$ as the counter-CDF, which evolves:

$$G_{t+1}(x) = P(w_{t+1}^i > x) = P(A_t^i w_t^i > x) > P(w_t^i > \frac{x}{A_t^i})$$
$$= \int_0^\infty G_t(\frac{x}{A})f(A)dA.$$

- If there is a steady state: $G(x) = \int_0^\infty G(\frac{x}{A})f(A)dA$.

Power Law Models: Simple Mechanics

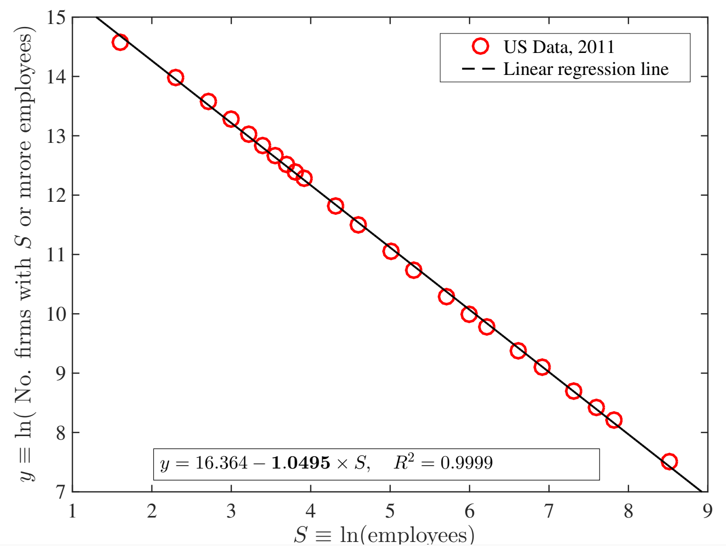
- W_t^i :wealth of individual i and $\overline{W}_t$: average (per-capita) wealth.
- Since wealth grows over time, define: $w_t^i \equiv \frac{W_t^i}{\overline{W}_t}$.
- Proportional random growth: $w_{t+1}^i = A_t^i \times w_{t^i}$, where A_t^i is i.i.d over t and i, with density $f(A)$.
- Define: $G_t(w_t) \coloneqq P(w_t > x)$ as the counter-CDF, which evolves:

$$G_{t+1}(x) = P(w_{t+1}^i > x) = P(A_t^i w_t^i > x) > P(w_t^i > \frac{x}{A_t^i})$$
$$= \int_0^\infty G_t(\frac{x}{A})f(A)dA.$$

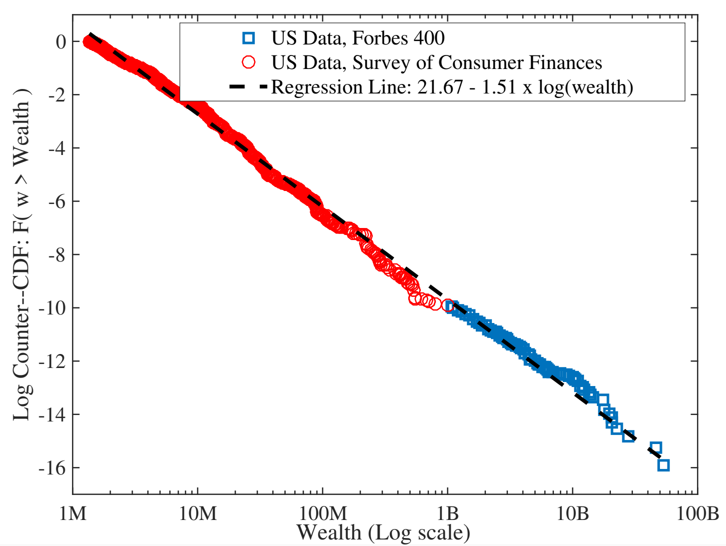
- If there is a steady state: $G(x) = \int_0^\infty G(\frac{x}{A})f(A)dA$.
- Guess Pareto distr.: $G(x) = \frac{c}{x^\alpha}$ and plugging in yields:

$$1 = \int_0^\infty A^\alpha f(A)dA = \mathbb{E}(A^\alpha).$$

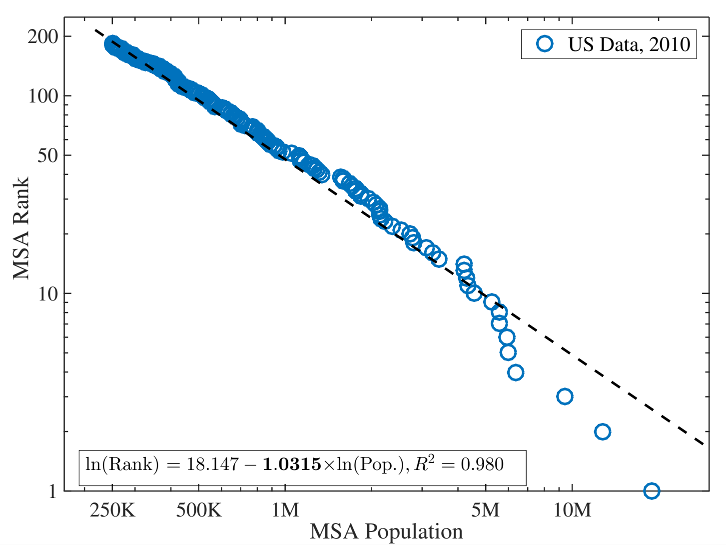
Size Distribution for Firms, $\alpha = 1.0495$



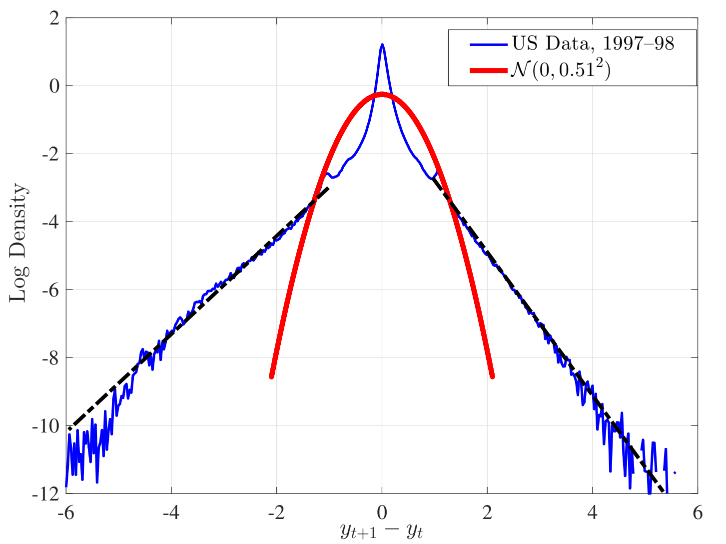
US Wealth Distribution, $\alpha = 1.51$



US Cities Size Distribution, $\alpha = 1.0315$



Annual Income Growth Distribution, US Males



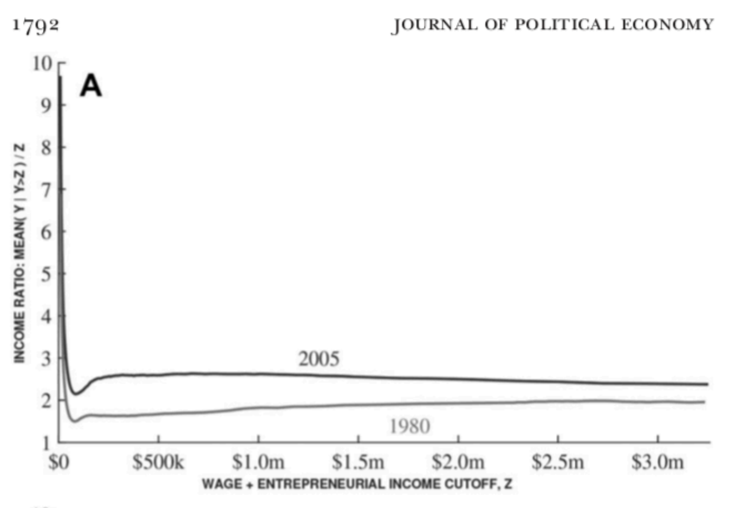
Measuring the Tail

- Researchers do not always have access to micro data to plot the log density vs log x graph and see linear relationship.
- A Pareto distribution can be verified and tail index estimated in a simpler way.
- First: If $\mathrm{P}(y > x) = k \times x^{-\alpha}$, the conditional mean of y above any $\overline{y}$ is $\mathrm{E}(y|y > \overline{y}) = \overline{y} \times \frac{\alpha}{1-\alpha}$.

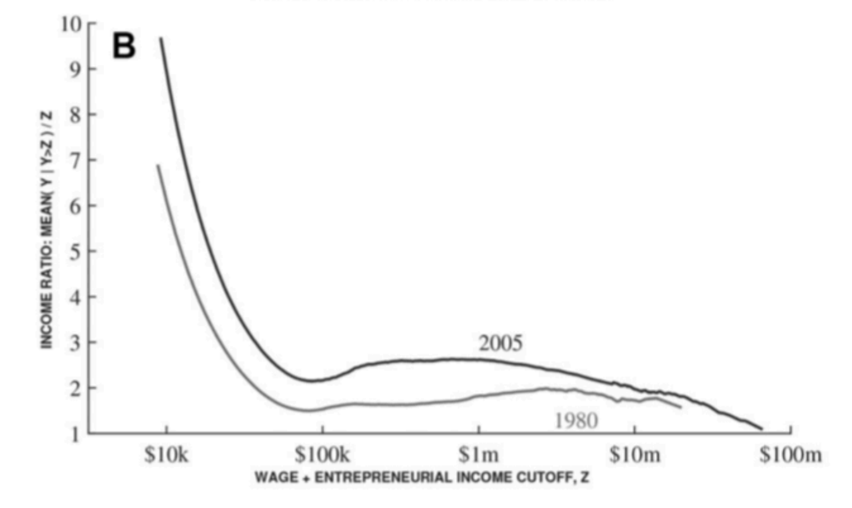
Two Implications:

1. $\frac{\mathrm{E}(y|y>\overline{y})}{\overline{y}} = \frac{\alpha}{1-\alpha}$. LHS can be measured by IRS tabulations.
2. $\frac{\mathrm{E}(y|y>\overline{y}_1)}{\mathrm{E}(y|y>\overline{y}_2)} = \frac{\overline{y}_1}{\overline{y}_2}$. Ratio of top income (or wealth) share must be constant.

Jones and Kim (2018, JPE) extending Saez (2001)



Jones and Kim (2018, JPE), very top end



Can Aiyagari-style Models Generate a Pareto Tail?

- In the Aiyagari-style models where the only risk is from idiosyncratic labor income shocks, the wealth distribution inherits the tail of the income distribution.

Can Aiyagari-style Models Generate a Pareto Tail?

- In the Aiyagari-style models where the only risk is from idiosyncratic labor income shocks, the wealth distribution inherits the tail of the income distribution.
- So, unless income shocks are modeled to have a Pareto tail, the wealth distribution will not either.

Can Aiyagari-style Models Generate a Pareto Tail?

- In the Aiyagari-style models where the only risk is from idiosyncratic labor income shocks, the wealth distribution inherits the tail of the income distribution.
- So, unless income shocks are modeled to have a Pareto tail, the wealth distribution will not either.
- Because the income distribution has a much thinner tail, the wealth distribution will also have the same.

Can Aiyagari-style Models Generate a Pareto Tail?

- In the Aiyagari-style models where the only risk is from idiosyncratic labor income shocks, the wealth distribution inherits the tail of the income distribution.
- So, unless income shocks are modeled to have a Pareto tail, the wealth distribution will not either.
- Because the income distribution has a much thinner tail, the wealth distribution will also have the same.
- See Hubmer, Krusell, Smith (NBER MA, 2020), Carroll, Slajek and Tokuo (2014, AER P&P) for quantitative demonstration of lack of Pareto tail.

Can Aiyagari-style Models Generate a Pareto Tail?

- In the Aiyagari-style models where the only risk is from idiosyncratic labor income shocks, the wealth distribution inherits the tail of the income distribution.
- So, unless income shocks are modeled to have a Pareto tail, the wealth distribution will not either.
- Because the income distribution has a much thinner tail, the wealth distribution will also have the same.
- See Hubmer, Krusell, Smith (NBER MA, 2020), Carroll, Slajek and Tokuo (2014, AER P&P) for quantitative demonstration of lack of Pareto tail.
- Bottom line: If top end inequality matters in your model (it often does), the Aiyagari model will fail to provide a good framework for your analysis.
- Power Law models generate a thick Pareto tail in wealth much more easily.