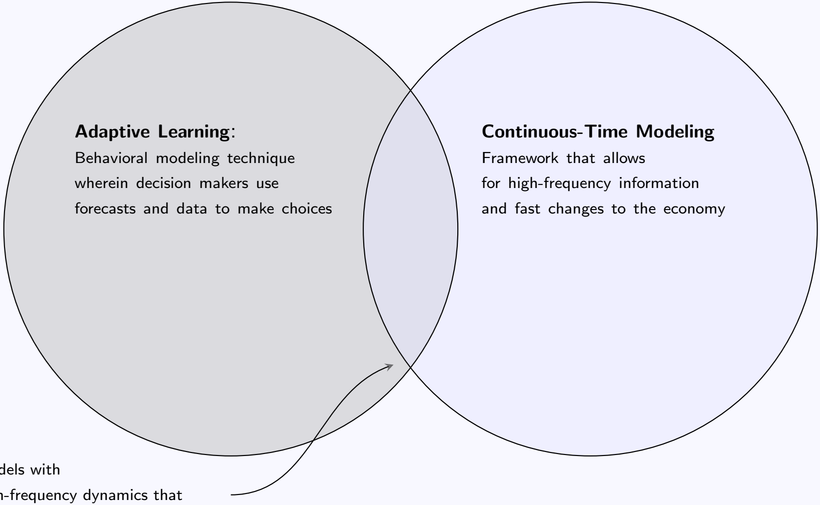
Bounded Rationality in Macroeconomic Models:

A Continuous-Time Approach

Chandler Lester

Overview


Adaptive Learning:
Behavioral modeling technique
wherein decision makers use
forecasts and data to make choices
Continuous-Time Modeling
Framework that allows
for high-frequency information
and fast changes to the economy
Models with
high-frequency dynamics that
allow for expectations to develop

Motivation

- Continuous-Time modeling is emerging as a powerful modeling paradigm
    - Faster computation, often using sparse grids
    - Detailed information on distributions
    - Addition of financial frictions to macroeconomic models
    - Examination of systems with kinks
    - Simulation of models with high frequency data
- Adaptive learning is a beneficial tool used in discrete-time modeling
    - Relaxes strict assumption of rational expectations (RE)
    - Provides insight into how individuals react to certain policy rules or changes
- Using adaptive learning in continuous-time will eventually open up tools

Main Questions

- Can models with continuous-time learning dynamics converge to RE?
- How do results of learning models differ in continuous-time vs discrete-time?
  - Do certain properties differ between continuous and discrete versions of the model?
- Are there alternative ways to study learning in high frequency settings?
  - Do agents need to take in data points continuously?
  - Can agents approximate continuous-time processes using partial information?

Results

- Continuous-time models can converge to rational expectations
    - I build a framework for learning in dynamic programming problems
- In continuous-time learning an agent converges to RE differently
    - Near RE equilibrium agents have less volatile forecasts of key parameters
    - Adjustments to forecasts tend to be smaller
- Agent in continuous model can sample at lower frequencies with similar results

Creating a Continuous-Time Adaptive Learning Model

Building Our Framework

1. Continuous-time linear quadratic (LQ) Model
   - Outline a continuous-time LQ framework
   - Linearize a Real-Business Cycle (RBC) model
2. Mapping between our agent's perceptions and reality
   - Define learning dynamics, the T-map, for our continuous-time model
3. Continuous-time updating rule
   - Continuous-time analog of recursive least squares

A Linear-Quadratic Real Business-Cycle Model in Continuous-Time

Why Use a Linear-Quadratic Real Business-Cycle Model?

- Linear-Quadratic Framework
  - Contains feedback mechanisms useful for learning
  - Tractable methods for solving models
- Real-Business Cycle Models
  - Want to test adaptive learning on a workhorse model
  - Has linear constraints that allow it to be put in the LQ format
  - Can examine learning dynamics without market imperfections or heterogeneity

A General Linear-Quadratic Format

- To use the LQ setting, the problem first needs to be in the following form

$$V(x_0) = \max_{u_t} \; -\mathbb{E}\int_{t=0}^{\infty} e^{-\rho t}(\hat{x}_t' R\hat{x}_t + \hat{u}_t' Q\hat{u}_t + 2\hat{x}_t' W\hat{u}_t)$$

  where the state variables evolve according to

$$dx_t = Ax_t + Bu_t + CdZ_t$$

- x_t and u_t are vectors of state and control variables respectively

The Continuous-Time Real Business-Cycle Model

- Macroeconomic fluctuations are driven by changes in productivity/technology
- These fluctuations impact consumption and labor decisions
- The continuous RBC model:

$$V(k_0, z_0) = \max_{c_t, k_t, h_t} \mathbb{E} \int_{t=0}^{\infty} e^{-\rho t} \left\{ \frac{c_t^{1-\sigma}}{1-\sigma} - \chi \frac{h_t^{1+\varphi}}{1+\varphi} \right\}$$

- subject to

$$\begin{aligned} c_t + i_t &= A k_t^{\alpha} (z_t h_t)^{1-\alpha} \\ dk_t &= (-\delta k_t + i_t) dt \\ d\log(z_t) &= -\theta \log(z_t) dt + \sigma_z dZ_t \end{aligned}$$

- dZ_t is the increment of the Wiener process
  - Approximate this as $\varepsilon_t \sqrt{dt}$, $dZ_t \sim N(0, dt)$
- Increments of the Wiener process are therefore independent and Gaussian

Linearizing the RBC Model I

- Rewrite the objective function

$$r(x_t, u_t) = \frac{1}{1-\sigma}[Ak_t^\alpha (z_t h_t)^{1-\alpha} - i_t]^{1-\sigma} - \chi \frac{h_t^{1+\varphi}}{1+\varphi}$$

- In this case,

$$x_t = \begin{bmatrix} 1 \\ k_t \\ \log(z_t) \end{bmatrix} \quad u_t = \begin{bmatrix} h_t \\ i_t \end{bmatrix}$$

- Then linearize using a Taylor expansion about the non-stochastic steady state

$$\begin{aligned} r(x,u) = {} & r(\bar{x},\bar{u}) + (x-\bar{x})' r_x(\bar{x},\bar{u}) + (u-\bar{u})' r_u(\bar{x},\bar{u}) \\ & + \frac{1}{2}(x-\bar{x})' r_{xx}(\bar{x},\bar{u})(x-\bar{x}) + \frac{1}{2}(u-\bar{u})' r_{uu}(\bar{x},\bar{u})(u-\bar{u}) \\ & + (x-\bar{x})' r_{xu}(\bar{x},\bar{u})(u-\bar{u}) \end{aligned}$$

Linearizing the RBC Model II

Now the problem can be put into the LQ format

$$V(x_0) = \max_{u_t} \; -\mathbb{E}\int_{t=0}^{\infty} e^{-\rho t}(\hat{x}_t' R \hat{x}_t + \hat{u}_t' Q \hat{u}_t + 2\hat{x}_t' W \hat{u}_t)$$

where the state variables evolve according to

$$d\hat{x}_t = A\hat{x}_t + B\hat{u}_t + C dZ_t$$

here $\hat{x}_t = x_t - \bar{x}$ and $\hat{u} = u_t - \bar{u}$

Linearization Details

Solving the LQ Model I

- The Hamilton-Jacobi-Bellman (HJB) equation for the LQ problem

$$\rho V(x) = \max_u \; - x'Rx - u'Qu - 2x'Wu + \mathbb{E}\left(V_x(x)dx_t + \frac{1}{2} V_{xx}(x)(dx_t)^2 \right)$$

- Using "guess and verify," posit that $V(x) = -x'Px - \xi$

$$\rho x'Px + \rho\xi = \max_u \{x'Rx + u'Qu + 2x'Wu + 2x'P(Ax + Bu) + P(CC')\}$$

- Using first order conditions, the policy function for u is

$$u = -(Q')^{-1}(W + PB)'x = -\tilde{F}x$$

Solving the LQ Model II

P and ξ can now be found with the following recursive algorithm

$$P_i = -(2\tilde{A}_i')^{-1}(\tilde{F}_i'Q^{-1}\tilde{F}_i + R - 2W\tilde{F}_i)$$
$$\xi_i = \rho^{-1}\text{trace}(P_{i-1}CC'),$$

where $\tilde{A}_i = (A - B\tilde{F}_i - .5\rho)$ and $\tilde{F}_i = (Q')^{-1}(W + P_{i-1}B)'$

Adaptive Learning Dynamics

Continuous-Time Shadow-Price Learning

- Our setting contains a distinct relationship between perceptions and actuality

$$\rho V(x) = \max_u \; -x'Rx - u'Qu - 2x'Wu + \mathbb{E}\left(V_x(x)dx_t + \frac{1}{2}V_{xx}(x)(dx_t)^2 \right)$$

  - $V_x(x)$ is a vector of state variable shadow prices, under RE $V_x(x) = 2Px$

- Use this to develop a mapping between agent's forecasts and actuality
  - Our agent forms expectations of future shadow prices via linear forecasting model

$$\lambda_t = Hx_t + \mu_t$$

  - We can re-analyze the value function problem using,

$$\mathbb{E}[V_x(x)] = \lambda^e = Hx$$

Continuous-Time Shadow-Price Learning

- Using Hx as an approximation for $V_x(x)$, we can rewrite the HJB as,

$$\rho V^P(x) = \max_u \{-x'Rx - u'Qu - 2x'Wu + (Hx)'(Ax + Bu) + \frac{1}{2}(H'CC')\}.$$

- The policy function for this problem is then,

$$u = -\frac{1}{2}(Q^{-1})'(2W - H'B)x = -F^{SP}(H, B)x$$

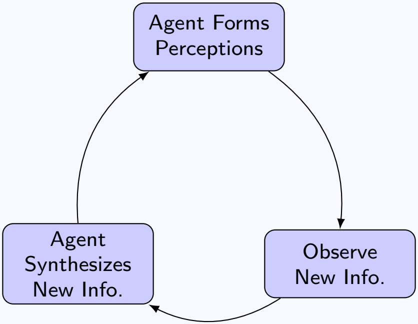
- Map perceptions to actuality using the policy function & the envelope theorem

$$\begin{aligned} \mathbb{E}[V_x(x)] = \lambda^e &= T^{SP}(H, A, B)x \\ &= \rho^{-1}\big(-2R + 2H'A - (H'B - 2W)F^{SP}(H, B)\big)x \end{aligned}$$

- This mapping is called the T-map, it directly impacts agents expectations

Continuous-Time Updating Rule

The System


Agent Forms Perceptions
Observe New Info.
Agent Synthesizes New Info.

Recursive Least Squares (RLS)

- Suppose we want to estimate coefficients for:

$$\underbrace{y_t}_{\text{Dependent Variable}} = \underbrace{\theta}_{\text{Coefficients}} \cdot \underbrace{x_{t-1}}_{\text{Observations}} + \underbrace{\varepsilon_t}_{\text{White Noise}}$$

- In discrete-time RLS takes the following form:

$$\underbrace{\theta_t}_{\text{Parameter of interest}} = \underbrace{\theta_{t-1}}_{\text{Previous value}} + \underbrace{\gamma_t}_{\text{Strength of Response to new info.}} \cdot \underbrace{R_{t-1}^{-1} \cdot x_{t-1}}_{\text{System Variance Weighted by Obs.}} \underbrace{(y_t - \theta_{t-1}' x_{t-1})}_{\text{Forecast Error}}$$

Continuous-Time Recursive Least Squares I

- The Kalman filter has a continuous-time analog
- We can use this to build a continuous-time version of recursive least squares

$$\begin{aligned} \dot{x}_t &= Ax_t + \nu_t, && \nu_t \sim N(0, R_t) \\ y_t &= \theta' x_t + e_t, && e_t \sim N(0, r_t) \end{aligned}$$

- Kalman Filter for this system

$$\begin{aligned} \text{Covariance Update:} \quad & \dot{\mathcal{P}} = A\mathcal{P} + \mathcal{P}A' + R_t - \mathcal{P}\theta_t' r_t^{-1} \theta_t \mathcal{P} \\ \text{Kalman Gain:} \quad & K = \mathcal{P}\theta_t' r_t^{-1} \\ \text{Forecast Update:} \quad & \dot{\hat{x}}_t = A\hat{x}_t + K[y_t - \theta_t' \hat{x}_t] \end{aligned}$$

Continuous-Time Recursive Least Squares II

- Now, re-imagine the state-space model as

$$\begin{aligned} \dot{\theta}_t &= \nu_t, & \nu_t &\sim N(0, R_t) \\ y_t &= \theta_t' x_t + e_t, & e_t &\sim N(0, r_t) \end{aligned}$$

- where $R_t = 0$ and $r_t = 1/\alpha_t$
- The RLS system will be

$$\begin{aligned} \text{Covariance Update:} \quad & \dot{\mathcal{P}} = -\alpha_t \mathcal{P} x_t' x_t \mathcal{P} \\ \text{Kalman Gain:} \quad & K = \alpha_t \mathcal{P} x_t' \\ \text{Parameter Update:} \quad & \dot{\hat{\theta}}_t = K[y_t - \hat{\theta}_t' x_t]. \end{aligned}$$

Discrete and Continuous-Time RLS

Discrete-Time RLS:

$$\underbrace{\theta_t}_{\text{Parameter of interest}} = \underbrace{\theta_{t-1}}_{\text{Previous value}} + \underbrace{\gamma_t}_{\substack{\text{Strength of}\\\text{Response}\\\text{to new info.}}} \cdot \underbrace{R_{t-1}^{-1} \cdot x_{t-1}}_{\substack{\text{System Variance}\\\text{Weighted by Obs.}}} \underbrace{(y_t - \theta_{t-1}' x_{t-1})}_{\text{Forecast Error}}$$

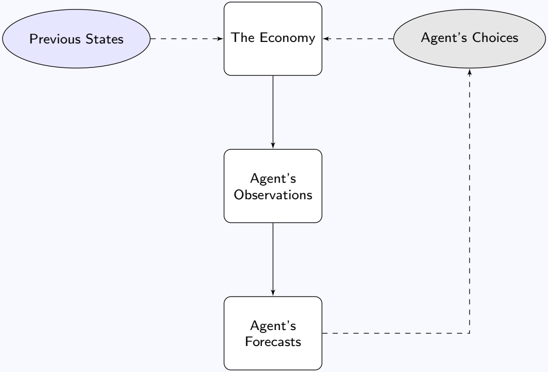
Continuous-Time RLS:

$$\underbrace{\dot{\theta}_t}_{\substack{\text{Change in}\\\text{Parameter of Interest}}} = \underbrace{\gamma_t}_{\substack{\text{Strength of}\\\text{Response}\\\text{to new info.}}} \cdot \underbrace{\mathcal{P}_t \cdot x_t}_{\substack{\text{System Variance}\\\text{Weighted by Obs.}}} \underbrace{(y_t - \theta_t' x_t)}_{\text{Forecast Error}}$$

The Shadow-Price Learning Model

Shadow-Price Learning

- Agents learn to optimize and forecast
- They make decisions based on the economy around them
- AND the economy is impacted by these choices



Shadow-Price Learning Algorithm

- Mathematically we can model our economy as

$$
\begin{aligned}
dx_t &= Ax_t dt + Bu_t dt + CdZ_t \\
d\mathcal{P}_t &= -\gamma_t \mathcal{P}_t x_t x_t' \mathcal{P}_t dt \\
dH_t' &= \gamma_t \mathcal{P}_t x_t (\lambda_t - H_t x_t)' dt \\
dA_t' &= \gamma_t \mathcal{P}_t x_t (dx_t - Bu_t dt - A_t x_t dt)' \\
u_t &= -F^{SP}(H_t, B) x_t = -\frac{1}{2}(Q')^{-1}(2W - H_t' B) x_t \\
\lambda_t &= T^{SP}(H_t, A_t, B) x_t \\
\gamma_t &= \kappa (t + N)^{-\nu}
\end{aligned}
$$

- $\mathcal{P}_t$ is a covariance matrix for the state variables x_t
- Agents do know B but must estimate A

Analyzing the Models

Comparing Learning Outcomes I

- Both shadow-price learning models ran for $50,000$ periods (quarters)
- The increment of time for the continuous model was set to $dt = 1/100$
- The discrete model updated $50,000$ times
- The continuous model updated $50,000 \times 100$ times
- Models were initialized equal distances from REE values of A and H

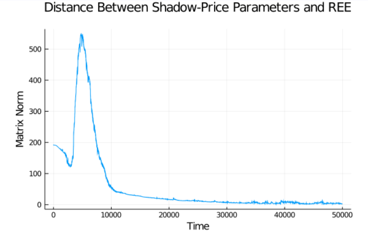
Comparing Learning Outcomes III

Shadow-Price Parameter Outcomes

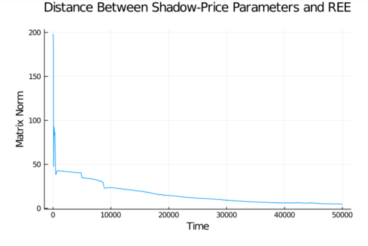
	Learning Outcome		REE Value	
Variable	**Discrete**	**Continuous**	**Discrete**	**Continuous**
Constant	-189.909	-190.564	-190.764	-190.642
	(0.026)	(0.0018)		
Capital	-0.077	-0.075	-0.075	-0.072
	(0.0002)	(0.0000)		
Productivity	2.544	2.548	2.731	2.644
	(4.64)	(0.004)		

- Standard deviation were measured over the last 1,000 simulated iterations
- Discrete matrix norm has a mean of 4.05 and standard deviation of 1.61
- Same measures for the continuous-time version are 2.36 and 1.84×10^{-5}

Matrix Norms Over Time



(a) Discrete-time Matrix Norms



(b) Continuous-Time Matrix Norms

Second Moment Analysis I

- Compare second moments of continuous and discrete RBC models against data
- Using data on consumption, output, investment, hours, and wages (1960-2019)
- Detrend data using a HP-filter and logarithmic transformations
- Compared these data against 1,000 simulations of the learning models

Second Moment Analysis II

Second Moments and Correlations with Output of Key Economic Variables

	Standard Deviation*			Correlation with Output		
Variable	**Data**	**Discrete**	**Cont.**	**Data**	**Discrete**	**Cont.**
Output	1.43%	1.02%	0.99%	1.00	1.00	1.00
Consumption	0.510	0.300	0.474	0.748	0.970	0.957
Investment	2.88	2.76	2.80	0.799	0.989	0.985
Hours	0.646	0.362	0.370	0.650	0.982	0.980
Wage	0.660	0.650	0.644	0.172	0.994	0.993

*standard deviations for variables other than output are measured relative to output

- Second moments of the continuous-time model are slightly closer
- Correlations with output also match the data more closely

Computational Advantages of Alternative Data Sampling Schemes

An Alternative Sampling Scheme

- Assuming agents get information and update continuously may be unrealistic
- Data is generated according to a continuous-time process that depends on dt
- The agent samples the data at a lower-frequency Δ, for simplicity $\frac{dt}{\Delta} \in \mathbb{N}$

$$
\begin{aligned}
dx_t &= Ax_t dt + Bu_t dt + CdZ_t \\
\Delta \mathcal{P}_t &= -\gamma_t \mathcal{P}_t x_t x_t' \mathcal{P}_t \Delta \\
\Delta H_t' &= \gamma_t \mathcal{P}_t x_t (\lambda_t - H_t x_t)' \Delta \\
\Delta A_t' &= \gamma_t \mathcal{P}_t x_t (x_t - Bu_t \Delta - A_t x_t \Delta - x_{t-\Delta})' \\
u_t &= -F^{SP}(H_t, B) x_t = -\frac{1}{2}(Q')^{-1}(2W - H'B) x_t \\
\lambda_t &= T^{SP}(H_t, A_t, B) x_t \\
\gamma_t &= \kappa (t + N)^{-\nu}
\end{aligned}
$$

- The agent still views the world as continuous
- However, they understand that their data observations are at a lower-frequency

Results of Alternative Data Sampling Schemes

Continuous-Time Learning Results under Varying Data Frequencies

dt	Specification	Matrix Norm	Computational Time (sec)
$dt = 1/364$	$\Delta = 1/364$	2.44	4429
	$\Delta = 1/91$	2.41	560
	$\Delta = 1/52$	2.39	504
	$\Delta = 1/26$	2.37	231
dt=1/100	$\Delta = 1/100$	2.35	664
	$\Delta = 1/50$	2.43	287
	$\Delta = 1/25$	2.57	132

- Continuous-time learning can take a lot of computational time
- Computational time can be reduced without losing precision

Concluding Remarks

Conclusion

- The model does converge to REE under shadow-price learning dynamics
- Near the REE the continuous-time model's estimates of shadow-price parameters have less volatility than the discrete-time model's estimates
- Continuous-time convergence appears to be less volatile may be more stable
- Alternative sampling schemes maintain the continuous-time model's accuracy and take less computational time

Linearization Details

Now the problem can be put into the LQ format

$$V(x_0) = \max_{u_t} \; -\mathbb{E}\int_{t=0}^{\infty} e^{-\rho t}(\hat{x}_t' R \hat{x}_t + \hat{u}_t' Q \hat{u}_t + 2\hat{x}_t' W \hat{u}_t)$$

where the state variables evolve according to

$$d\hat{x}_t = A\hat{x}_t + B\hat{u}_t + CdZ_t$$

here $\hat{x}_t = x_t - \bar{x}$ and $\hat{u} = u_t - \bar{u}$ and

$$\underset{3\times 3}{R} = \begin{bmatrix} r(\bar{x},\bar{u}) & \frac{1}{2}r_x(\bar{x},\bar{u}) \\ \frac{1}{2}r_x(\bar{x},\bar{u}) & \frac{1}{2}r_{xx}(\bar{x},\bar{u}) \end{bmatrix} \quad \underset{2\times 2}{Q} = \begin{bmatrix} \frac{1}{2}r_{uu}(\bar{x},\bar{u}) \end{bmatrix} \quad \underset{3\times 2}{W} = \begin{bmatrix} r_u(\bar{x},\bar{u}) \\ r_{xu}(\bar{x},\bar{u}) \end{bmatrix}.$$

The continuous-time version of the state transition matrix A will differ from the discrete time version while B and C will remain the same. Thus,

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\delta & 0 \\ 0 & 0 & -\theta \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 \\ 0 \\ \sigma_\varepsilon \end{bmatrix}$$

Back