

Scientific Computing in Economics and Finance

Past, Present and Future

John Stachurski

Tokyo College and Australian National University

April 25th 2025

What is economics?

The study of allocation of scarce resources among competing human users.

Combines:

- Social science (history, politics, etc.)
- Data science (statistics, machine learning)
- Mathematical modeling
- Computer science

What is economics?

The study of allocation of scarce resources among competing human users.

Combines:

- Social science (history, politics, etc.)
- Data science (statistics, machine learning)
- Mathematical modeling
- Computer science

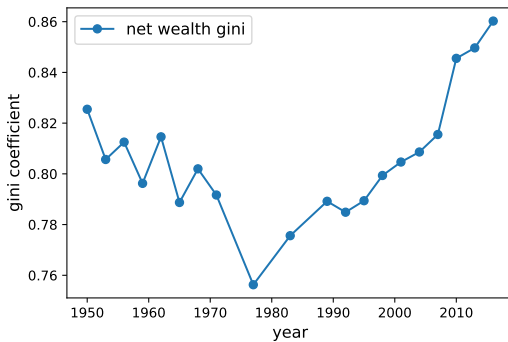
What do economists do?

Study economic phenomena and trends

Examples.

- Booms and busts
- Long run growth
- Concentration in industries
- Changes in the distribution of wealth

Example. Long run trends in wealth inequality



Practical tasks

- Design pricing mechanisms for Amazon
- Set interest rates at the BOJ
- Assess tax policies
- Advise on competition policy
- Assess pension plans
- Study the impact of COVID / COVID relief policy

Many problems addressed by economists are quantitative

Examples.

- How will a 10% increase in the top income tax rate affect GDP over the next decade?
- How will a two year increase in the retirement age affect government debt?
- What will be the impact of a 25 BPS \uparrow in the federal funds rate on unemployment next year?

Quantitative problems require mathematical/computational modeling

Basic research also requires mathematical modeling

The goal of the scientist is to comprehend the phenomena of the universe that he observes around him.

To prove that he understands he must be able to predict.

To predict quantitatively one must have a mechanism for producing numbers.

This necessarily entails a mathematical model.

– Richard Bellman (1920 – 1984)

Basic research also requires mathematical modeling

The goal of the scientist is to comprehend the phenomena of the universe that he observes around him.

To prove that he understands he must be able to predict.

To predict quantitatively one must have a mechanism for producing numbers.

This necessarily entails a mathematical model.

– Richard Bellman (1920 – 1984)

How should economists build models?

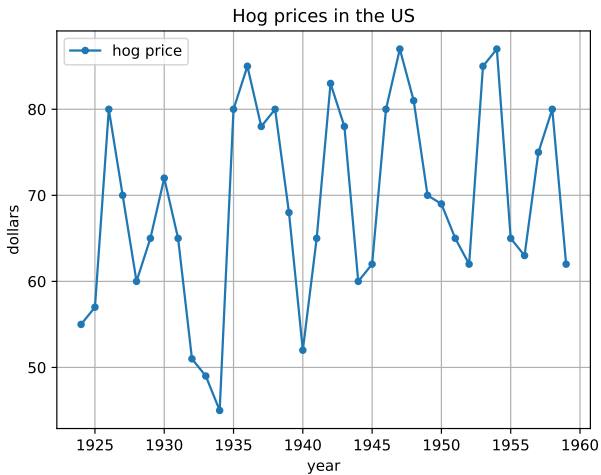
Why is economics different to astrophysics, chemistry, etc.?

- Economic processes are nonstationary (no immutable laws)
- Economic outcomes depend on human choices
- Human choices depend on beliefs, incentives, etc.

Example: the cobweb model

An “old” economic model

- Benner (1876)
- Haas and Ezekil (1926)
- Ricci (1930)
- Kaldor (1934, 1938)
- Ezekil (1938)
- Harlow (1960)
- Rosen, Murphy, and Scheinkman (1994)



Hypotheses:

- Farmers need time to raise hogs (say, one “period”)
- Farmers forecast future prices using current prices

Outcomes:

- 1 Suppose price is currently high
- 2 Farmers \uparrow capacity, shift towards hog production
- 3 Next period, high supply floods the market, prices \downarrow
- 4 Seeing this low price, farmers \downarrow capacity
- 5 Next period, supply is low and prices \uparrow ...

In this scenario,

$$q^d(p_t) = q^s(p_{t-1}^e)$$

- p_{t-1}^e is the **expected** time t price, formed at $t - 1$

But how to farmers form expectations?

First guess:

$$p_{t-1}^e = p_{t-1}$$

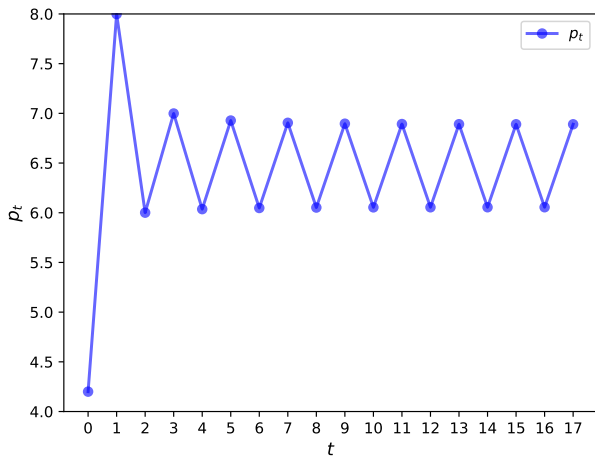
So now we have

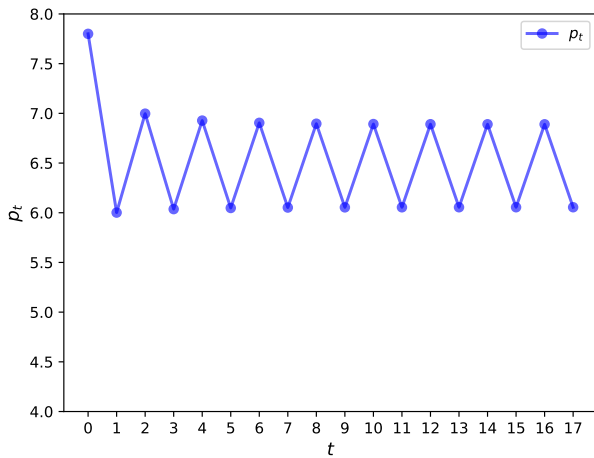
$$q^d(p_t) = q^s(p_{t-1})$$

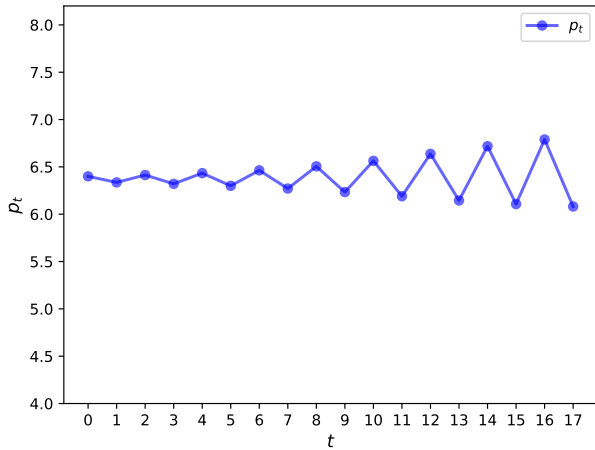
Solving for p_t gives

$$p_t = f(p_{t-1}) \quad \text{where} \quad f(p) = (q^d)^{-1}(q^s(p))$$

Now let's simulate...





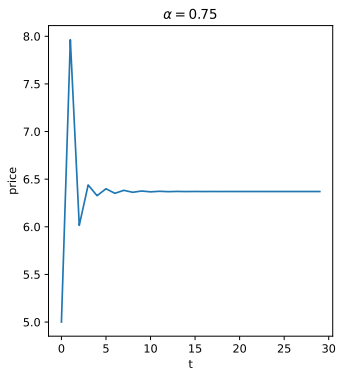
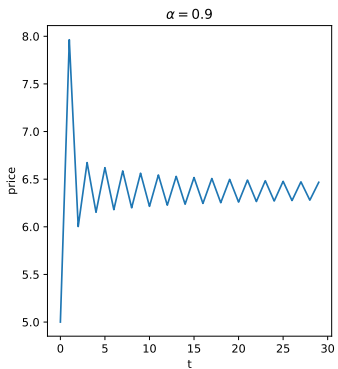


The model replicates cycles — but there are problems!

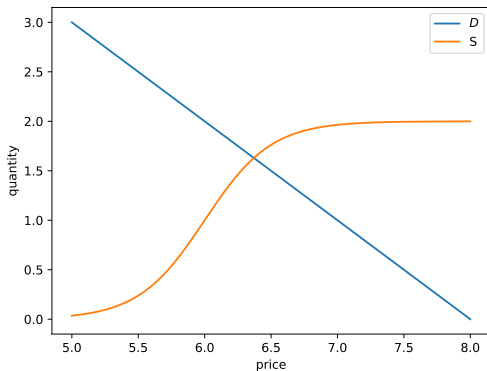
Predictions are **very sensitive** to how we model **expectations**

Example. Suppose we switch to $p_{t-1}^e = p_{t-2}^e + \alpha(p_{t-1} - p_{t-2}^e)$

- Called “adaptive expectations”



Or we could use “rational expectations”



Lessons:

- ① modeling human behavior is essential
- ② the “right” way to model humans is unclear
- ③ predictions are very sensitive to their expectations
- ④ models are nonstationary because the way we predict is nonstationary

Summary: economic modeling is hard – but we shouldn't give up!

Scientific computing in economics

Anonymous consultant's report on HPC in economics:

Economists are relative newcomers to the field of computational sciences...

Economists have long been influenced by dogmatic tribalism...

It would appear that many (so called) 'theories' have been poorly (if at all!) proven...

Computational models in economics are still often simplistic...

Scientific computing in economics

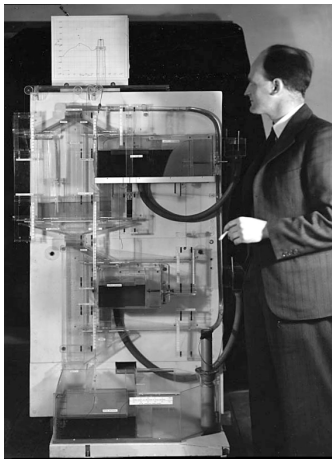
Anonymous consultant's report on HPC in economics:

Economists are relative newcomers to the field of computational sciences...

Economists have long been influenced by dogmatic tribalism...

It would appear that many (so called) 'theories' have been poorly (if at all!) proven...

Computational models in economics are still often simplistic...



Actually economists are pioneers
(William Phillips, 1949)

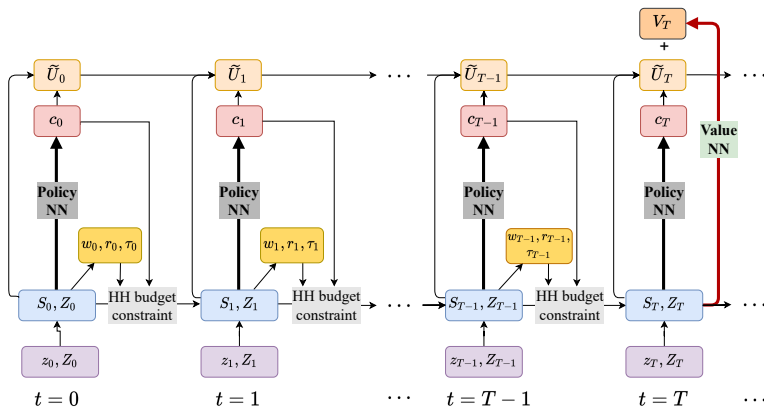
Modern scientific computing in economics

There are many examples of modern, sophisticated scientific computing in economics

Example. DeepHAM by Jiequn Han, Yucheng Yang, and Weinan E

- heterogeneous agent model
- firms and households linked by markets
- individual and aggregate risk
- general equilibrium
- NNs to represent human reaction functions

KS DeepHAM



Trends in scientific computing

Technology is advancing rapidly along many dimensions

- improvements to algorithms
- new programming languages
- new hardware
- AI, etc.

What trends are affecting economics?

How are they changing economic research?

Trend 1: Proprietary → Open Source

Proprietary

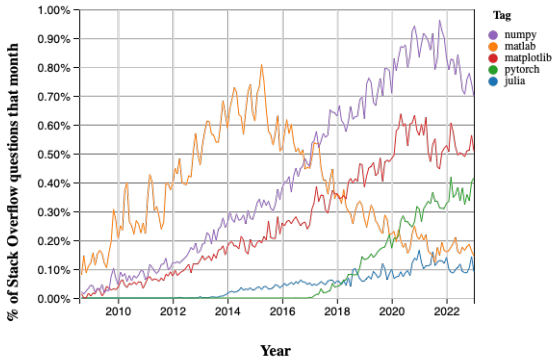
- Excel
- MATLAB, Mathematica
- STATA, Eviews, SPSS.

Open Source / Open Standard

- Python
- Julia
- R

closed and stable vs open and fast moving

Popularity:



Trend 2: Low Level → High Level

Low level

- C/C++
- Fortran
- Assembly

High level

- Python
- Javascript
- PHP

Low level languages give us control

- control CPU
- control memory

High level languages give us

- abstraction
- automation
- flexibility, etc.

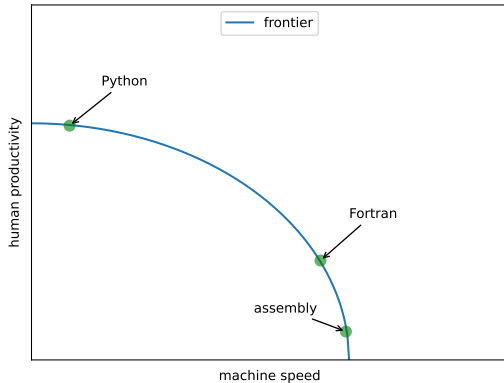
Example. 1 + 1 in C

```
#include <stdio.h>
int main() {
    int sum = 1 + 1;
    printf("1 + 1 = %d\n", sum);
    return 0;
}
```

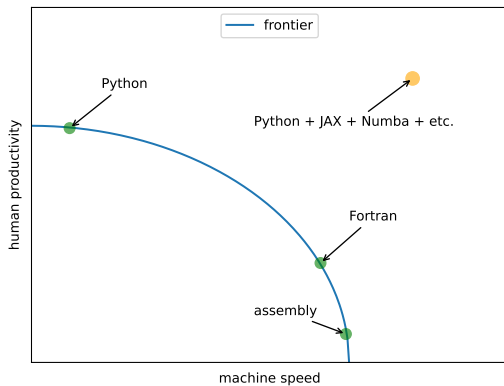
Example. $1 + 1$ in Python

```
sum = 1 + 1
print("1 + 1 = ", sum)
```

Trade-offs:



New trend — a shifting frontier!



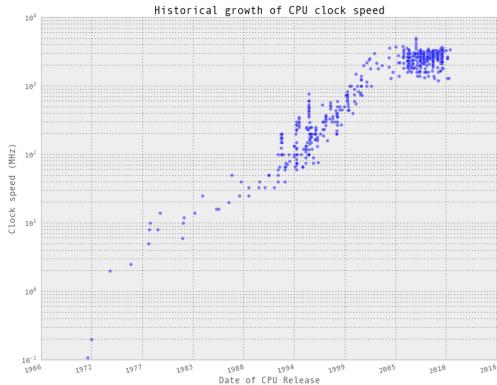
Example. Numba / codon generate fast machine code from Python

See code at

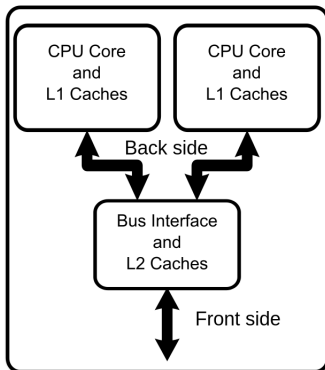
https://github.com/QuantEcon/tokyo_college_2023/

Trend 3: Parallelization

CPU frequency (clock speed) growth is slowing



Chip makers have responded by developing multi-core processors



Source: Wikipedia

GPUs are becoming increasingly important



Applications: machine learning, deep learning, etc.

Example. JAX

See code at

https://github.com/QuantEcon/tokyo_college_2023/

The limits of computer power

Consider optimizing a function via brute force

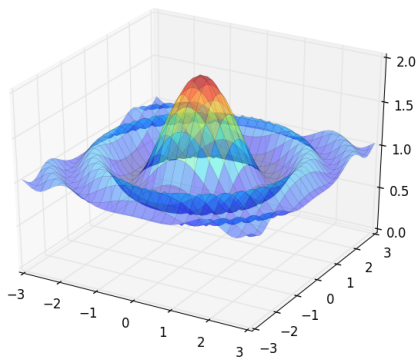


Figure: The function to maximize

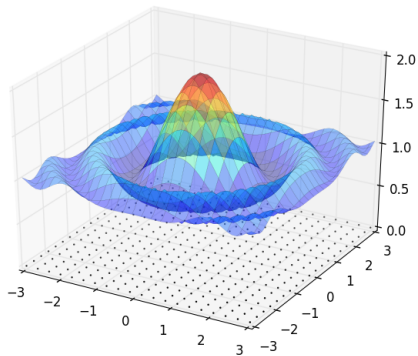


Figure: Grid of points to evaluate the function at

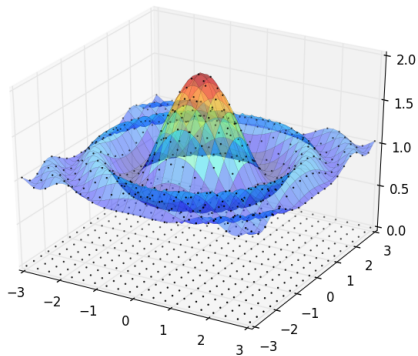


Figure: Evaluations

Grid size = $20 \times 20 = 400$

Outcomes

- function evaluations = 400
- Time taken ≈ 0
- Max value = 1.951
- True maximum = 2

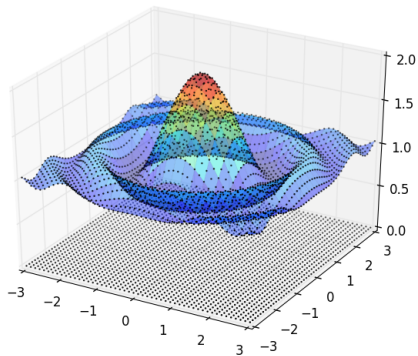


Figure: $50^2 = 2500$ evaluations

- function evaluations = 50^2
- Time taken = 101 microseconds
- Max value = 1.992
- True maximum = 2

But now suppose we have more choice variables

- 3 vars: $\max_{x_1, x_2, x_3} f(x_1, x_2, x_3)$
- 4 vars: $\max_{x_1, x_2, x_3, x_4} f(x_1, x_2, x_3, x_4)$
- ...

If we have 50 grid points per variable and

- 2 variables then evaluations = $50^2 = 2500$
- 3 variables then evaluations = $50^3 = 125,000$
- 4 variables then evaluations = $50^4 = 6,250,000$
- 5 variables then evaluations = $50^5 = 312,500,000$
- ...

Example. Recent study: Optimal placement of drinks across vending machines in Tokyo

Approximate dimensions of problem:

- Number of choices for each variable = 2
- Number of choice variables = 1000

Hence number of possibilities = 2^{1000}

How big is that?

```
In [10]: 2**1000
```

```
Out[10]:
```

```
107150860718626732094842504906000181056140481170  
553360744375038837035105112493612249319837881569  
585812759467291755314682518714528569231404359845  
775746985748039345677748242309854210746050623711  
418779541821530464749835819412673987675591655439  
460770629145711964776865421676604298316526243868  
37205668069376
```

Suppose my machine evaluates about 10^9 choices per second

How long would that take?

```
In [16]: (2**1000 / 10**9) / 31556926 # In years
```

Out[16]:

339547840365144349278007955863635707280678989995
899349462539661933596146571733926965255861364854
060286985707326991591901311029244639453805988092
045933072657455119924381235072941549332310199388
301571394569707026437986448403352049168514244509
939816790601568621661265174170019913588941596

What about high performance computing?

- faster CPUs
- clusters of GPUs
- ...

Let's say speed up is 10^{12} (wildly optimistic)

```
In [19]: (2**1000 / 10**(9 + 12)) / 31556926
```

```
Out[19]:
```

```
3395478403651443492780079558636357072806789899958  
9934946253966193359614657173392696525586136485406  
0286985707326991591901311029244639453805988092045  
9330726574551199243812350729415493323101993883015  
7139456970702643798644840335204916851424450993981  
6790601568621661265174170019
```

For comparison:

```
In [20]: 5 * 10**9 # Expected lifespan of sun
```

```
Out[20]: 5000000000
```

Message: There are serious limits to computation

What's required is clever analysis

Exploit all available structure

Good algorithms are still crucial

Algorithms interact with advances in hardware/software

What about machine learning & AI?

AI is helping economists

- find and summarize information (e.g., [Scholarcy](#))
- find patterns in large data sets
- write code (e.g., ChatGPT, Copilot)

Current gen AI cannot solve most economic problems

Consider the current global inflationary episode

- Many aspects are unprecedented

What will be the impact of policy interventions?

- Often we have no data
- Limitations on running experiments

An ongoing need for careful mathematical modeling

Conclusion

Economics is an important field that needs many perspectives

Economics is inherently quantitative / computational

High quality computational analysis is becoming far easier

In the next decade, AI will assist economic research but not radically transform it