Discussion

Reinforcement learning for household finance: designing policy via responsiveness

Bandyopadhyay & Maliar

Ed Hill

Advanced Analytics, Bank of England

19th January 2024

What’s it about? What’s the conclusion?

- Finding the optimal policy for a servicer of mortgages
  - Against various levels of delinquency and modes of eventual default
  - Servicers usually use more ad-hoc harsh or lenient policies

- The authors define a metric of ‘responsiveness’ to motivate the need to solve the problem…

  …and then solve it using RL, specifically Q-learning

- The policy found with RL is very different to harsh or lenient policies
  - And provides significantly better outcomes

Things I didn't like

- I think the separation/distinction between the motivation and the solution needed to be clearer - a computational implementation section with pseudocode would help

- I assumed that the responsiveness would be used as a state variable in the RL – it isn't, and I think this would be a good direction for future work

Things I did like

- A nice use of RL to solve a useful problem with a strikingly improved result
  - Often see ML give small changes in performance
- Detailed discussion around the intuition behind the change
  - Using the results to drive and validate a discussion – showing how ML and domain knowledge can play well together

Reinforcement Learning (RL) and Deep RL

- Foundational RL methods have been around for decades

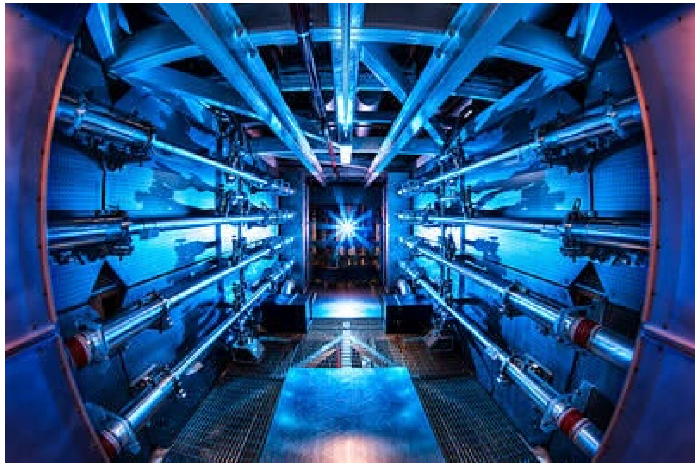
- In 2014, Deep RL = Deep Neural Networks + RL
  - Superhuman performance on Chess, Go, Shogi, and ATARI games
  - Robot control, self-driving cars, …
  - RLHF for training/aligning LLMs, Quantum RL

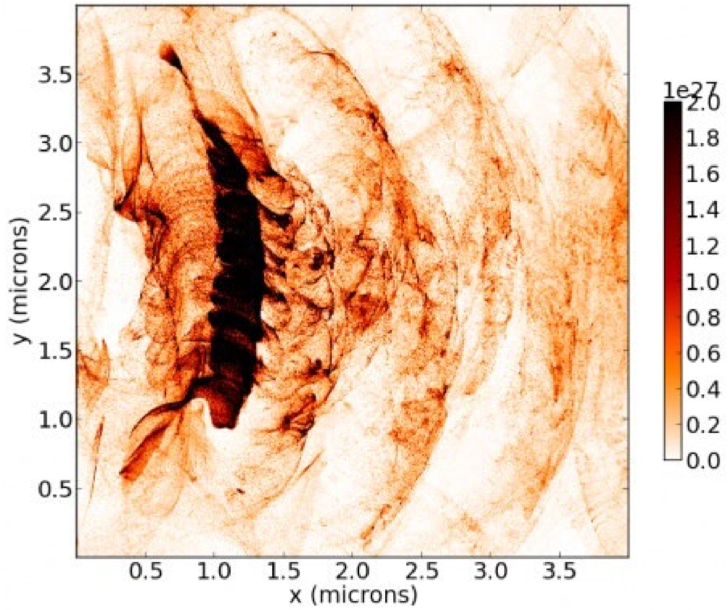
- As users: Powerful toolkit for <u>"forward-looking-ifying" models</u>

How to train	Networks / Aims	Architecture	E&E	Combining	Meta-learning	
						2013
Experience replay						
						2014
Prioritised ER TRPO	Double DQN	Distributed DQN				2015
	Duelling DQN Recurrent DQN		Intrinsic motivation			
	A3C	Asynchronous	Episodic control Bootstrapped DQN			2016
PPO	Distributional RL			Rainbow	Hierarchical learning	2017
			Curiosity Driven			
			Random Network Distillation			2018
				R2D2		
						2019
				NGU	Agent 57	2020

Developments in DRL as of 2020 [BoE ML Meetup]

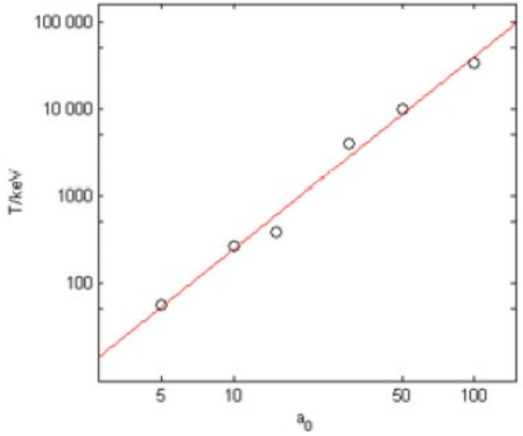
Computational experiments





$$T = e\vec{E} \cdot \Delta\vec{r} \propto \frac{a_0^2}{n_e \lambda^2},$$

$$U = n_e \sigma |\Delta\vec{r}| T \propto \frac{a_0^3}{n_e \lambda^3}.$$



Actual experiments
- $1 million per go
- Messy & partially observable
- Hard to tell if/why theory matches reality

Computational experiments
- University scale computing
- Clean and fully observable, intervenable
- Easy to tell if/why theory matches computational reality

Simple models
- Validated and tested against the computational experiment