

An Introduction to Deep Reinforcement Learning

Mingli Chen

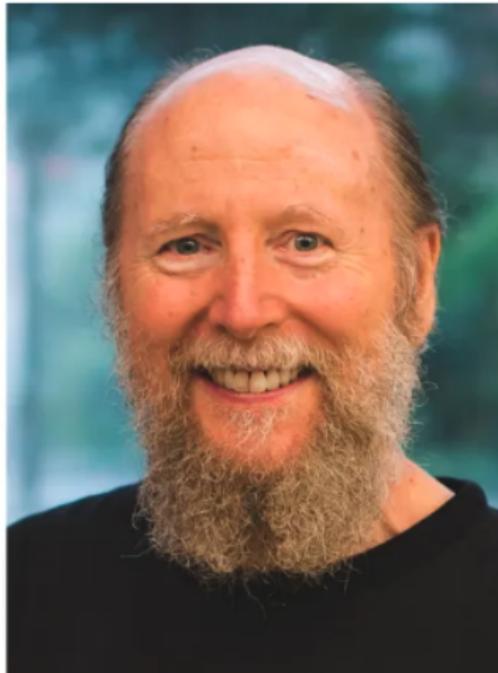
University of Warwick

July 7, 2025

Overview

- ▶ This talk introduces **Deep Reinforcement Learning (DRL)**, an important branch of modern **Artificial Intelligence (AI)**.
- ▶ We will examine the **fundamental concepts and core principles** of Reinforcement Learning (RL).
- ▶ The close relationship between RL and **Dynamic Programming (DP)**—a foundational tool in economics—will be emphasized.
- ▶ We will explore how DRL can be applied in **economics, finance, and complex decision-making problems**.
- ▶ Our goal is to inspire students to consider applying DRL in **economic modeling and related research**.

The Recent News...



Andrew Barto (left) and Richard Sutton (right) have won the 2024 Turing Award. Image: Association for Computing Machinery

Figure: AI pioneers Andrew Barto and Richard Sutton win 2025 Turing Award for groundbreaking contributions to reinforcement learning

What is Artificial Intelligence?

- ▶ AI is the study of how to make computers **think and learn like humans**.
- ▶ It combines ideas from **computer science, math, statistics, and cognitive science**.
- ▶ Goal: Build systems that can **solve problems, learn from data, and adapt to new situations**.

Simple Examples:

- ▶ Siri, Alexa (voice assistants)
- ▶ Spam filters in email
- ▶ Netflix recommendations

Types of AI

- ▶ **Narrow AI:** Designed for specific tasks (e.g., playing chess, recognizing faces)
- ▶ **General AI:** Human-level intelligence across a wide range of tasks (still theoretical)
- ▶ **Superintelligent AI:** Hypothetical AI that surpasses human intelligence (philosophical/futuristic)

Today, almost all AI is Narrow AI.

Key Techniques in AI

- ▶ **Machine Learning (ML)** – Letting computers learn patterns from data.
- ▶ **Deep Learning** – A powerful type of ML using neural networks with many layers.
- ▶ **Natural Language Processing (NLP)** – Enabling machines to understand and generate human language.
- ▶ **Computer Vision** – Teaching machines to "see" and interpret images/videos.

AI in Everyday Life

- ▶ Google Maps (route prediction)
- ▶ Spotify/Netflix (recommendations)
- ▶ Online shopping (personalized ads)
- ▶ Face ID (facial recognition)
- ▶ Self-driving cars

AI is already everywhere — often behind the scenes!

Challenges and Ethics in AI

- ▶ Bias in algorithms
- ▶ Privacy and surveillance
- ▶ Job displacement and automation
- ▶ Accountability and transparency

AI is powerful — but we must use it responsibly.

AI Economist

搜索 ai economist

综合 视频 99+ 番剧 0 影视 0 直播 0 专栏 99+ 用户 0

综合排序 最多播放 最新发布 最多弹幕 最多收藏 更多筛选 ▾



How AI is revolutionising science
The Economist | How AI is revolutionising science
@ 菲瑞瑞瑞大蚊蝎 · 2024-12-01



MIT Economist on Finance, AI, and Human Behavior
The Economist | MIT Economist on Finance, AI, and Human Behavior
@ 就是不健 · 06-30



How AI is transforming the creative industries
The Economist | How AI is transforming the creative industries
@ 雅思考官教雅思 · 2021-05-24



Free exchange (Update in progress)
The Economist | Free exchange (Update in progress)
@ 妈妈不用担心我的英语 · 2022-01-30



The Economist | How AI is generating a revolution in entertainment
The Economist | How AI is generating a revolution in entertainment...
@ 冬冬秋雨 · 2024-01-04



The Economist | AI and the US election
The Economist | AI and the US election
@ 三读经济学人 · 2023-07-07



AI如何彻底改变科研
The Economist | AI如何彻底改变科研
@ 三读经济学人 · 2023-09-20



The absent student
The Economist | The absent student
@ 三读经济学人 · 2023-09-13



The Economist | AI如何影响美国大选
The Economist | AI如何影响美国大选
@ 三读经济学人 · 2023-08-05



Generative AI for Economic Research: Use Cases and Implications for Economists
The Economist | Generative AI for Economic Research: Use Cases and Implications for Economists...
@ 山东大学陈强教授 · 2024-12-29

Large Language Model

- ▶ A type of artificial intelligence trained to understand and generate human language.
- ▶ Based on deep learning, especially **transformer** architecture (e.g., GPT, BERT).
- ▶ Trained on vast amounts of text data to learn grammar, facts, reasoning patterns, and more.

Key Features

- ▶ Predicts the next word in a sentence (language modeling).
- ▶ Can perform tasks like translation, summarization, question answering, and conversation.
- ▶ Learns *statistical patterns* in language—not conscious or understanding like humans.

Famous Examples: ChatGPT, Claude, Gemini, LLaMA, Grok, Deepseek

Deep Reinforcement Learning

Deep Reinforcement Learning in this Talk

- ▶ **Fundamental Concepts of Reinforcement Learning**
 - ▶ Definitions and illustrative examples
 - ▶ Connections between Dynamic Programming and RL
 - ▶ Comparing RL with Supervised and Unsupervised Learning
- ▶ **Advanced Topics and Applications**
 - ▶ Applications of RL in economics
 - ▶ Non-stationary and non-homogeneous environments in RL
 - ▶ Multi-agent reinforcement learning
 - ▶ Distributional reinforcement learning

What is Reinforcement Learning?

- ▶ Wikipedia: "Reinforcement learning is an interdisciplinary area of machine learning and optimal control concerned with how an intelligent **agent** should take **actions** in a dynamic **environment** in order to maximize a **reward** signal."
- ▶ Literal Decomposition:
 - ▶ Reinforcement: reward-driven
 - ▶ Learning: (optimal) policy function
- ▶ Components:
 - ▶ State of the Environment
 - ▶ Action taken by the Agent
 - ▶ Reward as a sequence of the State and the Action
- ▶ Suitable for problems involving **sequential decisions under uncertainty**.

What is Reinforcement Learning? (Cont.)

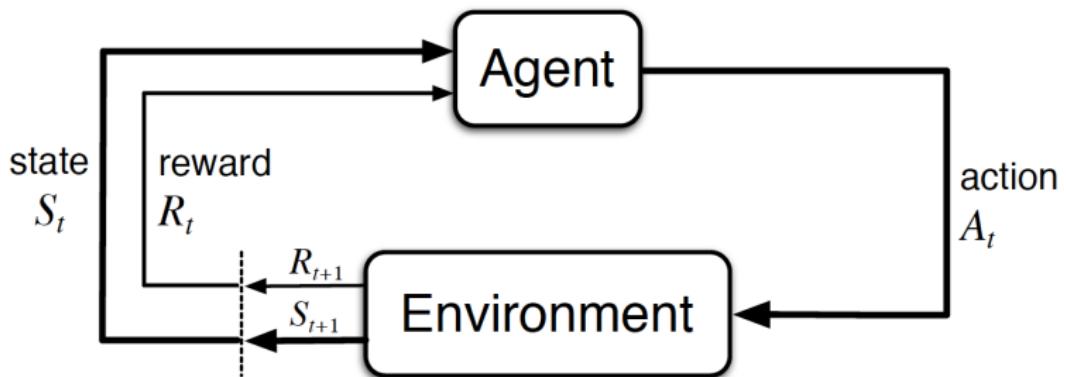


Figure: Agent-Envrionement Interaction by [Sutton and Barto \(2018\)](#)

What is RL: Mathematical Definition

- ▶ Definition: A Markov decision process (MDP) is a 5-tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where:
 - ▶ \mathcal{S} is a set of states called the state space
 - ▶ \mathcal{A} is a set of actions called the action space
 - ▶ $P(s, a, s') = \Pr(s_{t+1} = s' | s_t = s, a_t = a)$ is the prob. that action a in state s at time t will lead to state s' at time $t + 1$
 - ▶ $R(s, a, s')$ is the immediate reward received after transitioning from state s to state s' , due to action a
 - ▶ γ is the discount rate

What is RL: Mathematical Definition

- ▶ Definition: A Markov decision process (MDP) is a 5-tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where:
 - ▶ \mathcal{S} is a set of states called the state space
 - ▶ \mathcal{A} is a set of actions called the action space
 - ▶ $P(s, a, s') = \Pr(s_{t+1} = s' | s_t = s, a_t = a)$ is the prob. that action a in state s at time t will lead to state s' at time $t + 1$
 - ▶ $R(s, a, s')$ is the immediate reward received after transitioning from state s to state s' , due to action a
 - ▶ γ is the discount rate
- ▶ RL solves an MDP problem:
 - ▶ An Agent observes state $s_t \in \mathcal{S}$, takes an action $a_t \in \mathcal{A}$ based on a policy $g \in \mathcal{S} \rightarrow \mathcal{A}$, the environment produces a reward r_t and moves to s_{t+1}
 - ▶ The goal is to find an optimal policy that obtaining accumulative rewards $\sum_{i=1}^n \gamma^t R_t$ using a training algorithm

RL and Dynamic Programming (DP)

- ▶ RL is closely related to DP through the **Bellman Equation** (which will be introduced later).
- ▶ DP assumes a known environment model; RL learns through interaction when the model is **unknown**.
- ▶ DP methods: **Value Iteration, Policy Iteration**
- ▶ RL methods: **Q-learning, Policy Gradient**

In Economics: DP is widely used in models like consumption-savings, investment, and dynamic games. RL provides a data-driven alternative.

What is Deep Reinforcement Learning (DRL)?

- ▶ DRL combines **Reinforcement Learning** with **Deep Neural Networks**.
- ▶ Deep networks approximate policies or value functions — useful in **high-dimensional** settings.
- ▶ Breakthroughs in DRL allow AI to operate in complex, real-world environments.

Famous Applications:

- ▶ AlphaGo (Go playing AI)
- ▶ Autonomous driving, robotics
- ▶ Game AI (Atari, StarCraft)

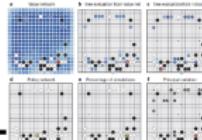
DRL at Centre of Recent Advances in Artificial Intelligence

ARTICLE

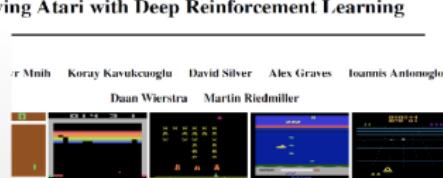
Mastering the game of Go with deep neural networks and tree search

David Silver¹, Alejo Huang¹, Chris J. Maddison², Arthur Guez¹, Li Julian Schrittwieser¹, Ioannis Antonoglou³, Veda Panneershelvam¹, John Nham¹, Nal Kalchbrenner¹, Ilya Sutskever¹, Timothy Lillicrap¹, Thore Graepel¹ & Demis Hassabis¹

DOI:10.1038/nature14961



Playing Atari with Deep Reinforcement Learning



Screen shots from five Atari 2600 Games: (Left-to-right) Pong, Breakout, Space Invaders, Asteroids, Asteroid Rider

Sources: Nature, arXiv, Boston Dynamics



Figure: Caption

Applications of DRL in Economics

- ▶ **Optimal consumption and saving** under uncertainty with large state spaces.
- ▶ **Asset allocation and trading strategies** in financial markets.
- ▶ **Macro policy simulation:** AI agents adjusting fiscal/monetary policy in simulated economies.
- ▶ **Mechanism design and market experiments** using simulated agents.

Strengths: Model-free, scalable, adaptive to nonlinear and dynamic environments.

What is RL: Example I

- ▶ State: current position
- ▶ Action: Up, Down, Left, Right
- ▶ Reward: ?

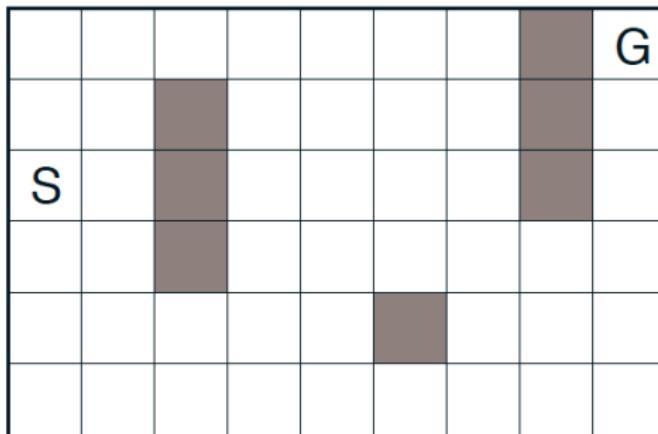


Figure: An Maze Problem

What is RL: Example II

- ▶ The *Frozen-Lake Environment*:

“The ice is slippery, so you won’t always move in the direction you intend.”

<i>SFFF</i>	(<i>S</i> : starting point, safe)
<i>FHFH</i>	(<i>F</i> : frozen surface, safe)
<i>FFFH</i>	(<i>H</i> : hole, fall to your doom)
<i>HFFG</i>	(<i>G</i> : goal, where the frisbee is located)

Figure: Frozen-Lake

What is RL: Example III

- ▶ The *Cart-Pole* Environment:
[GIF](#)
- ▶ State:
 - ▶ Cart Position: [-4.8, 4.8]
 - ▶ Cart Velocity: [-Inf, Inf]
 - ▶ Pole Angle: [-24°, 24°]
 - ▶ Pole Angular Velocity: [-Inf, Inf]
- ▶ Action: 0 (Left) or 1 (Right)
- ▶ Reward: +1 for every step unless failed

What is RL: Example IV

- ▶ A McCall job search model in labor economics
- ▶ State: features that characterize an offer, for example, (w_t) where $w_t \in [w_{\min}, w_{\max}]$
- ▶ Action: 0 (Reject) or 1 (Accept)
- ▶ Reward: unemployment compensation c if Reject and w_t if Accept and the game ends

What is RL: Example V

- ▶ A consumption-saving model (finite or infinite horizon) in macroeconomics
- ▶ State: (k_t, ϵ_t) , where $k_t \in [k_{\min}, k_{\max}]$ is the capital holding, $\epsilon_t \in \{0, 1\}$ is the employment status
- ▶ Action: c_t , the consumption
- ▶ Reward: $u(c_t)$, the utility

Intuition Behind RL Algorithms

- ▶ Define the accumulative reward $G_t = \sum_{t=1}^n \gamma^t R_t$

- ▶ The celebrated Bellman Equation:

$$\begin{aligned} V^*(s) &= \max_a \mathbb{E}[R_t + \gamma G_{t+1} \mid S_t = s, A_t = a] \\ &= \max_a \mathbb{E}[R_t + \gamma V_*(S_{t+1}) \mid S_t = s, A_t = a] \\ &= \max_a R_t + \gamma \sum_{s'} P(s'|s, a) V^*(s') \end{aligned}$$

- ▶ Version for State-Action Value Function (Q-Function):

$$Q^*(s, a) = R(s, a) + \gamma \sum_{s'} P(s'|s, a) \max_{a'} Q^*(s', a')$$

- ▶ Note that:

$$g^*(s) = \arg \max_a Q^*(s, a)$$

$$V^*(s) = \max_a Q^*(s, a)$$

The Agent

- ▶ The decision-making policy g :
 - ▶ Indirect: value function approach: $V(s)$ or $Q(s, a)$
 - ▶ Direct: policy function approach: $a = g(s)$
 - ▶ How to parameterize the value/policy function?
- ▶ The behavioral policy:
 - ▶ E.g., the ϵ -greedy policy:

$$\pi(a|s) = \begin{cases} 1 - \epsilon + \frac{\epsilon}{|\mathcal{A}(s)|}, & \text{if } a = \operatorname{argmax}_{a'} Q(s, a') \\ \frac{\epsilon}{|\mathcal{A}(s)|}, & \text{otherwise} \end{cases}$$

- ▶ The *exploration-exploitation trade-off*
- ▶ Other structures facilitate the solution: e.g. the “memory for experiences”

Machine Learning: SL, UL, RL

- ▶ Three broad categories: Supervised Learning (SL), Unsupervised Learning (UL) and Reinforcement Learning (RL)
- ▶ SL: “You know what is true”
 - ▶ Data: $\{x_i, y_i\}_{i=1 \dots N}$
 - ▶ Task: find $f : \mathbb{X} \rightarrow \mathbb{Y}$ such that $f(x) \approx y$
- ▶ UL: “You don't know what is true”
 - ▶ Data: $\{x_i\}_{i=1 \dots N}$
 - ▶ Task: find some sort of underlying structure, correctly label/group the data based on x_i
- ▶ RL: “You know what shall be true”
 - ▶ Data: $\{x_t\}_{t=1 \dots T}$ is our generated state, $\{r_t\}_{t=1 \dots T}$ “signals of correctness”
 - ▶ Task: find $f : \mathbb{X} \rightarrow \mathbb{Y}$ an optimal policy function

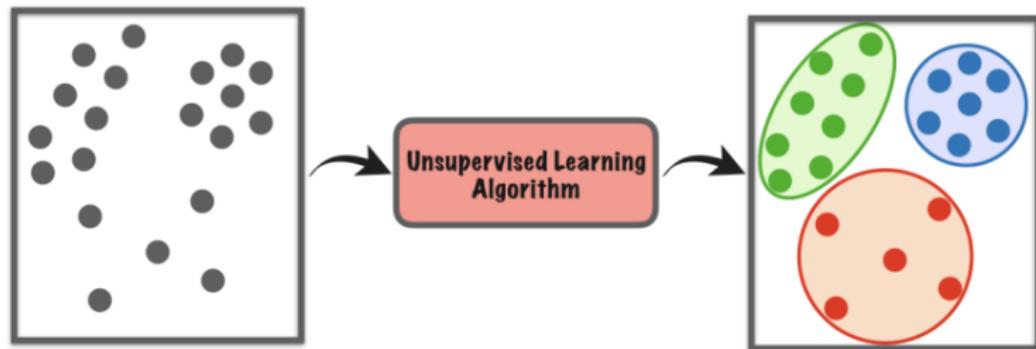
Supervised Learning: An illustration

The "Hello World" problem in supervised learning



Figure: MNIST data

Unsupervised Learning: An illustration



Optimal Control: DP and RL

- ▶ Recall the Bellman Equation in terms of Q-Function:
$$Q_*(s, a) = R(s, a) + \gamma \sum_{s'} P(s'|s, a) \max_{a'} Q_*(s', a')$$
- ▶ Dynamic Programming (DP): P is known and closed-form
- ▶ In practice:
 - ▶ P and R is not known or hard to express in closed-form
 - ▶ \mathcal{S}, \mathcal{A} is continuous/high-dimensional
 - ▶ the `max` operator is computationally expensive
- ▶ Problem 1: Simulation. The celebrated Q-learning algorithm:
$$Q^{i+1}(s, a) = (1 - \alpha)Q^i(s, a) + \alpha(r + \gamma \max_{a'} Q^i(s', a'))$$
- ▶ Problem 2 & 3: we use, e.g., Neural Network (Deep RL)
 - ▶ Critic: A Value Network $Q_\theta(s, a)$
 - ▶ Actor: A Policy Network $g_\phi(s)$

RL in Economics: Literature

- ▶ DRL in a Monetary Model ([Chen, Joseph, Kumhof, Pan and Zhou, 2021](#))
- ▶ AI, algorithmic pricing and collusion ([Calvano, Calzolari, Denicolo and Pastorello, 2020](#))
- ▶ AI as structural estimation: Deep Blue, Bonanza, and AlphaGo ([Igami, 2020](#))
- ▶ RL for Optimization of COVID-19 Mitigation policies ([Kompella, Capobianco, Jong, Browne, Fox, Meyers, Wurman and Stone, 2020](#))
- ▶ AI-Economist with tax policies ([Zheng, Trott, Srinivasa, Naik, Gruesbeck, Parkes and Socher, 2020](#))
- ▶ Multi-agent RL in Cheap Talk ([Condorelli and Furlan, 2023](#))

Non-stationary/Non-homogeneous RL

- ▶ We currently assume the MDP is stationary, i.e., time-invariant (no subscript t).
- ▶ Non-stationarity arises when:
 - ▶ A game ends after a finite number of stages (For example, optimality for a finitely-lived agent).
 - ▶ The MDP structure changes due to exogenous or endogenous shocks. For example, ([Li et al., 2022](#)): “In an mHealth study aiming to optimize the timing of smartphone-delivered interventions (designed to encourage anti-sedentary behavior among interns), non-stationarity emerges as users gradually habituate to or become overwhelmed by repeated prompts, leading to diminished responsiveness.”

Multi-agent Reinforcement Learning

Definition

- ▶ Extension of DRL to environments with multiple interacting agents
- ▶ Each agent learns its own policy $\pi_i : \mathcal{O}_i \rightarrow \mathcal{A}_i$
- ▶ Collective behavior emerges from individual learning

Challenges:

- ▶ Non-stationarity
- ▶ Credit assignment
- ▶ Scalability
- ▶ Partial observability

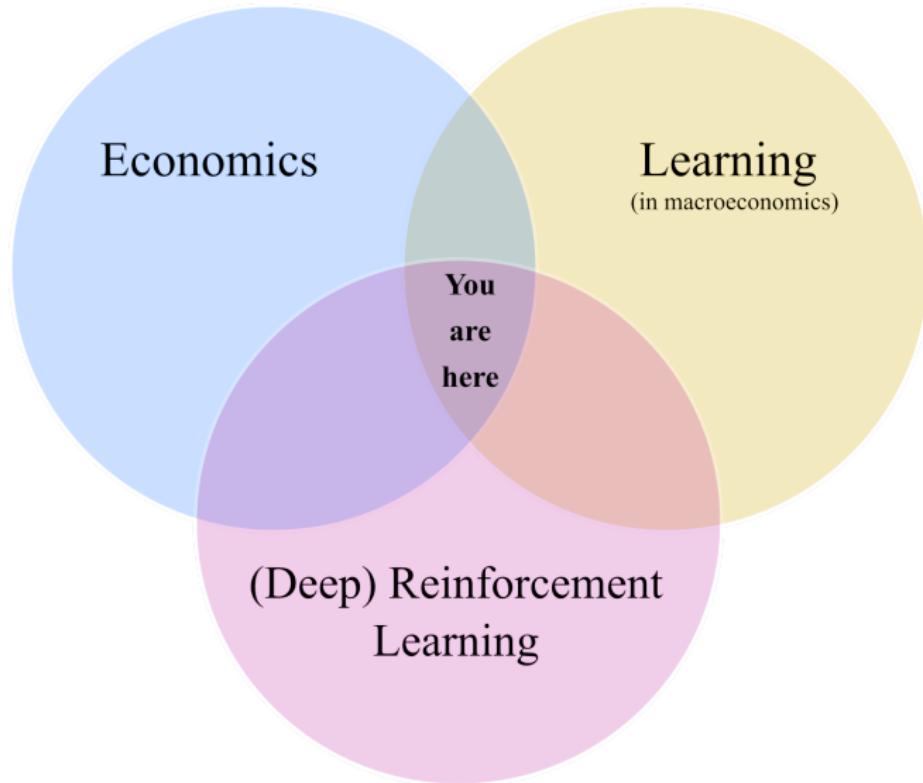
Approaches:

- ▶ Independent DQN
- ▶ MADDPG
- ▶ QMIX
- ▶ MAPPO

Video for application

Two Research Projects

I. Deep Reinforcement Learning in a Monetary Model



Learning in (macro)economics

- ▶ Rational Expectation (RE) convenience choice to solve a model, but not necessarily how people and businesses actually behave
- ▶ Learning approach to **bounded rationality (BR)**: specify agent knowledge and behaviour away from RE (often ad hoc)
- ▶ BR broadens available state space. See also [Moll \(2024\)](#).
- ▶ Example: Monetary policy reaction functions possibly very different under learning, such as forward guidance or the stability of Taylor rules

See [Eusepi and Preston \(2018\)](#) and [Hommes \(2021\)](#) for recent reviews.

Example: Adaptive learning

Agents are “econometricians” trying to estimate expected quantities via

$$x_{t+1}^e = x_t^e + \phi_t(x_t - x_t^e), \quad (1)$$

with a gain series ϕ_t .

Under least-squares learning it is usually taken to be $1/t$. Together with the (optimal) behavioural rules, i.e. linearised FOCs, this leads to a set of ordinary differential equations determining the expectations (E-)stability of the model.

That is, if a steady state is **stable under learning**, which then serves as a selection criterion.

See, [Sargent \(1993\)](#) and [Evans and Honkapohja \(2001\)](#).

Models populated with *Adaptively Learning Agents* put the agents on an equal footing with the econometrician who is observing data from the model.

- ▶ However, this type of **parametric** recursive method assumes that agents correctly specify the laws of motion and other relevant functional relationships of the model

We work with models populated by *Deep Reinforcement Learning Agents (a.k.a. Artificially Intelligent Agents)* who

- ▶ have no a priori knowledge about the structure of the economy
- ▶ use their utility realisations in response to their actions in order to learn nonlinear decision rules via deep artificial neural networks

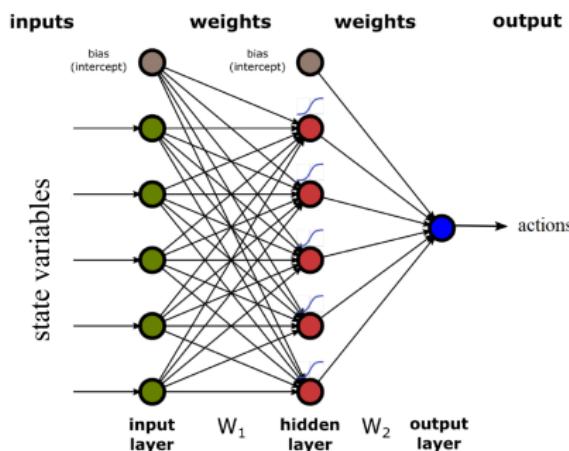
We adopt a policy-based deep reinforcement learning approach that can deal with high dimensional continuous action spaces.

Our approach enables agents to learn flexibly, as our learning algorithms are **nonparametric** and recursive, reducing the risk of misspecification

Allowing for misspecification and learning via expelling rational expectation agents and replacing them with “artificially intelligent” ones is also **reminiscent Sargent (1993)**

Deep Learning + Reinforcement Learning = DRL

In DRL, functions \mathcal{P} and V/Q are parameterised using **deep artificial neural networks** (Goodfellow et al., 2016), i.e. neural nets with several hidden layers, \mathcal{P}_ϕ and Q_θ :



Actor-critic DRL setting

\mathcal{P} and Q fulfil the **Bellman equation**

$$Q(s_t, a_t) = r(s_t, a_t) + \beta \mathbb{E}_{\mathcal{P}}[Q(s_{t+1}, a_{t+1})]. \quad (2)$$

using sampled state transitions as observations, i.e. interactions of the agent and the environment, and standard optimisation techniques like stochastic gradient descent, the policy and action-value function networks can be trained by iteratively minimising the Bellman residuum,

$$L(\phi, \theta) = \mathbb{E}_{s_t, a_t, r_t} \left[\frac{1}{2} (Q_\theta(s_t, a_t) - \hat{Q}_\theta(s_t, a_t))^2 \right], \quad (3)$$

$$\text{with } \hat{Q}_\theta(s_t, a_t) = r(a_t, s_t) + \beta \mathbb{E}_{\mathcal{P}}[Q_\theta(s_{t+1}, \mathcal{P}_\phi(s_{t+1}))]. \quad (4)$$

We use [Haarnoja et al. \(2018\)](#). The code we used for optimisation is available at

<https://github.com/pranz24/pytorch-soft-actor-critic>.

back

General DRL setting for (macro)economics

- ▶ Write down model (environment and state)
- ▶ Specify **learning** agents, e.g. households, firms, etc., and their **actions**
- ▶ Specify state transitions as MDP
- ▶ Learning using DRL algorithm (e.g. Haarnoja et al. (2018)):
 1. sample state transition(s) and store in memory
 2. train \mathcal{P}_ϕ and Q_θ from memory
 3. test \mathcal{P}_ϕ and Q_θ with new state transitions and metric of choice

II. Distributional RL (Bellemare et al., 2023)

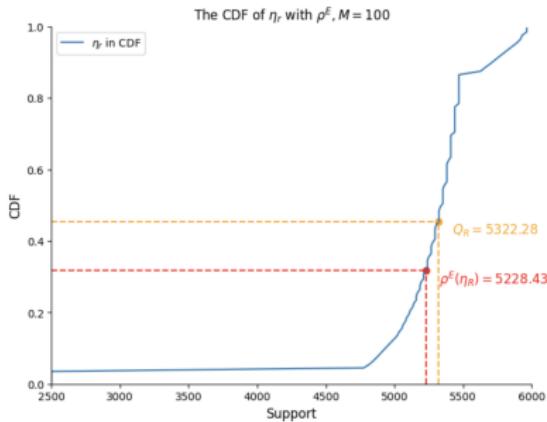
- ▶ Consider the case that you need to catch an early morning flight to a conference and want to maximize your sleep by choosing the shortest trip to the airport.
- ▶ Options:
 - ▶ Routine A: Always takes 1.5 hours.
 - ▶ Routine B: Typically takes 1.2 hours, but has a 10% chance of requiring maintenance, which could extend the trip to over 3 hours and result in missing your flight.
- ▶ Decision Trade-off:
 - ▶ An expectation maximizer would favor Routine B for its shorter average travel time.
 - ▶ However, if avoiding the risk of missing your flight is a high priority, you might prefer the consistent timing of Routine A.

Distributional RL (Bellemare et al., 2023)

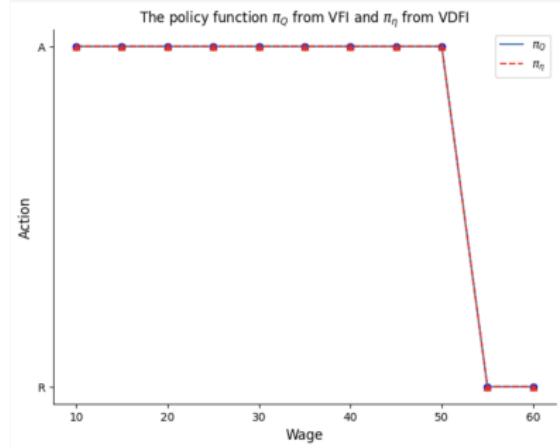
- ▶ η^* : value distribution function, where $\eta^*(s, a)$ represents the distribution of $G^*(s, a)$ for a given state-action pair (s, a)
- ▶ Our objective is to seek an optimal policy π^* such that for each state s we maximize a generalized value of return measure by a particular ρ :

$$\pi^*(s) = \arg \max_a \rho(\eta^*(s, a))$$

Distributional RL: The McCall Model



(a) The CDF of η_r



(b) The policy functions π_Q and π_η

Conclusion

- ▶ RL is nothing far away from economists
- ▶ RL could potentially help us to solve some complex settings where we should rely on simulations to solve agents' decision-makings
- ▶ MARL could even go further to study more interactive settings
 - ▶ policy-makers' problem in macro
 - ▶ strategic plays in game theory
 - ▶ firms' interaction in IO
- ▶ Distributional RL could help us to take into consideration the agent's risk aversion

Summary Outlook

- ▶ RL offers new ways to approach **dynamic decision-making under uncertainty**.
- ▶ DRL expands the power of RL to handle **complex, high-dimensional problems**.
- ▶ Many economic problems are naturally sequential and benefit from RL techniques.
- ▶ Students are encouraged to explore how AI can enhance research in economics.

The economist of the future may also be an AI engineer.

- Bellemare, Marc G, Will Dabney, and Mark Rowland,**
Distributional reinforcement learning, MIT Press, 2023.
- Calvano, Emilio, Giacomo Calzolari, Vincenzo Denicolo, and Sergio Pastorello**, “Artificial intelligence, algorithmic pricing, and collusion,” *American Economic Review*, 2020, 110 (10), 3267–3297.
- Chen, Mingli, Andreas Joseph, Michael Kumhof, Xinlei Pan, and Xuan Zhou**, “Deep reinforcement learning in a monetary model,” *arXiv preprint arXiv:2104.09368*, 2021.
- Condorelli, Daniele and Massimiliano Furlan**, “Cheap Talking Algorithms,” *arXiv preprint arXiv:2310.07867*, 2023.
- Eusepi, Stefano and Bruce Preston**, “The Science of Monetary Policy: An Imperfect Knowledge Perspective,” *Journal of Economic Literature*, March 2018, 56 (1), 3–59.
- Evans, George W. and Seppo Honkapohja**, *Learning and Expectations in Macroeconomics*, Princeton University Press, 2001.
- Goodfellow, Ian, Yoshua Bengio, Aaron Courville, and Yoshua Bengio**, *Deep learning*, Vol. 1, MIT press Cambridge, 2016.

Haarnoja, Tuomas, Aurick Zhou, Pieter Abbeel, and Sergey Levine, "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor," *arXiv-eprint*, 2018, *1801.01290*.

Hommes, Cars, "Behavioral and Experimental Macroeconomics and Policy Analysis: A Complex Systems Approach," *Journal of Economic Literature*, March 2021, *59* (1), 149–219.

Igami, Mitsuru, "Artificial intelligence as structural estimation: Deep Blue, Bonanza, and AlphaGo," *The Econometrics Journal*, 2020, *23* (3), S1–S24.

Kompella, Varun, Roberto Capobianco, Stacy Jong, Jonathan Browne, Spencer Fox, Lauren Meyers, Peter Wurman, and Peter Stone, "Reinforcement learning for optimization of COVID-19 mitigation policies," *arXiv preprint arXiv:2010.10560*, 2020.

Li, Mengbing, Chengchun Shi, Zhenke Wu, and Piotr Fryzlewicz, "Testing stationarity and change point detection in reinforcement learning," *arXiv preprint arXiv:2203.01707*, 2022.

Moll, Benjamin, "The Trouble with Rational Expectations in Heterogeneous Agent Models: A Challenge for Macroeconomics," *London School of Economics, mimeo, available at <https://benjaminmoll.com>*, 2024.

Sargent, Thomas J, "Bounded rationality in macroeconomics: The Arne Ryde memorial lectures," *OUP Catalogue*, 1993.

Sutton, Richard S and Andrew G Barto, *Reinforcement learning: An introduction*, MIT press, 2018.

Zheng, Stephan, Alexander Trott, Sunil Srinivasa, Nikhil Naik, Melvin Gruesbeck, David C Parkes, and Richard Socher, "The ai economist: Improving equality and productivity with ai-driven tax policies," *arXiv preprint arXiv:2004.13332*, 2020.