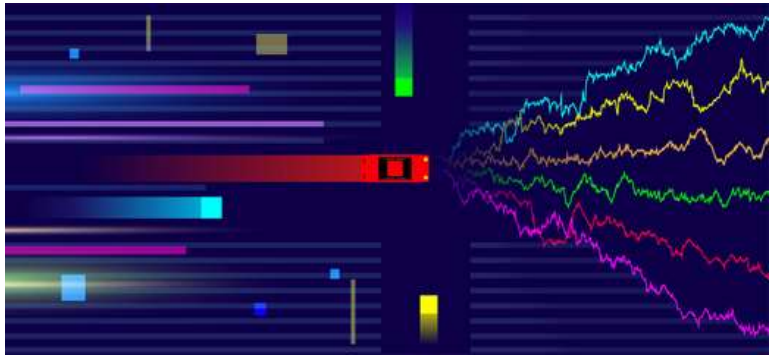
RL for Operations
Day 1: MDP Basics, VI+PI, Deep RL

Sean Sinclair, Sid Banerjee, Christina Yu
Cornell University



Plan for Today

MDP Basics

- Basic framework for Markov Decision Processes
- Tabular RL Algorithms with policy iteration + value iteration
- DeepRL algorithms (and their "tabular" counterparts)

Simulation Packages

- OpenAI Framework for simulation design
- Existing packages and code-bases for RL algorithm development

Simulation Implementation

- Develop simulator for problem using OpenAI Gym API

Tabular RL Algorithms

- Implement basic tabular RL algorithms to understand key algorithmic design aspects of *value estimates + value iteration*, *policy iteration*

Plan for Today

MDP Basics

- Basic framework for Markov Decision Processes
- Tabular RL Algorithms with policy iteration + value iteration
- DeepRL algorithms (and their "tabular" counterparts)

Simulation Packages

- OpenAI Framework for simulation design
- Existing packages and code-bases for RL algorithm development

Simulation Implementation

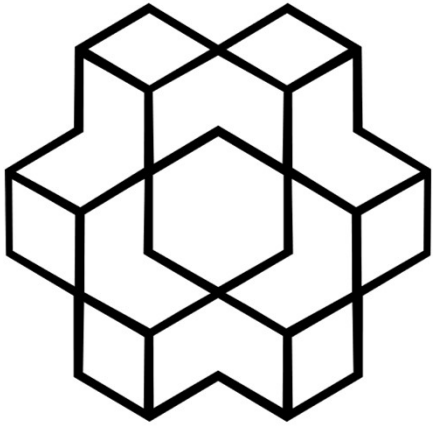
- Develop simulator for problem using OpenAI Gym API

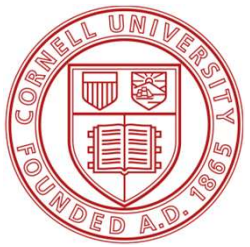
Tabular RL Algorithms

- Implement basic tabular RL algorithms to understand key algorithmic design aspects of *value estimates + value iteration*, *policy iteration*

Custom Simulator

Sean Sinclair,
Cornell University





Markov Decision Process (MDP)

Environment: Determine reward and new state

Policy: Determine action based on state

Finite Horizon

An MDP is defined by: $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, r, T, s_0, \gamma\}$

$\mathcal{S}$	State space
$\mathcal{A}$	Action space
$r_h : \mathcal{S} \times \mathcal{A} \to [0, 1]$	Reward
$T_h : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$	Transitions
H	Time horizon
$\pi_h : \mathcal{S} \to \Delta(\mathcal{A})$	Policy

This Code Demo

- Develop simulator for problem using OpenAI Gym API

- “Register” the environment with OpenAI Gym, check the environment for bugs via the stable baselines environment checker

References

https://github.com/seanrsinclair/RLinOperations

Step-By-Step

1. Open Anaconda Prompt via search toolbar
2. git clone https://github.com/seanrsinclair/RLinOperations

(Note: If git is not installed then can download from: https://github.com/git-guides/install-git)

3. cd RLinOperations
4. code .

This will open the visual studio code window. There will be five folders, one for the slides, and one for each of the code demos.

5. cd custom_simulator\ORSuite (or custom_simulator/ORSuite depending on platform)
6. conda env create --name custom_simulator --file environment.yml
7. conda activate custom_simulator
8. pip install -e .
9. conda install jupyter
10. jupyter notebook

Can now navigate to examples folder and open the demo.