# Policy Evaluation

## Monte Carlo Evaluation

Marius Lindauer

Leibniz Universität Hannover

Automated Machine Learning Hannover
AutoML.org

# Policy Evaluation without a model

- Goal: Policy Evaluation without a model
  - Given data and/or ability to interact with the environment
  - Efficiently compute a good estimate of a policy $\pi$
- For example: Estimate expected total purchases during an online shopping session for a new automated product recommendation policy

# Monte Carlo (MC) Policy Evaluation

- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots$ in MDP $M$ under policy $\pi$
- $V\pi(s) \approx \mathbb{E}_{T \sim \pi}[G_t \mid s_t = s]$
  - Expectation over trajectories $T$ generated by following $\pi$
- Simple idea: Value = mean return
- If trajectories are all finite, sample set of trajectories & average returns

Automated Machine Learning Hannover

# Monte Carlo (MC) Policy Evaluation

- If trajectories are all finite, sample set of trajectories & average returns
- Does not require MDP dynamics/rewards
- No bootstrapping
- Does not assume state is Markov
- Can only be applied to episodic MDPs
  - Averaging over returns from a complete episode
  - Requires each episode to terminate

# Monte Carlo (MC) Policy Evaluation

- Aim: estimate $V^\pi(s)$ given episodes generated under policy $\pi$
  - $s_1, a_1, r_1, s_2, a_2, r_2, \ldots$ where the actions are sampled from $\pi$
- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots$ in MDP $M$ under policy $\pi$
- $V^\pi(s) = \mathbb{E}[G_t \mid s_t = s]$
- MC computes empirical mean return
- Often do this in an incremental fashion
  - After each episode, update estimate of $V^\pi$

# First-Visit Monte Carlo (MC) On Policy Evaluation

Initialize $N(s) = 0$, $G(s) = 0 \; \forall s \in S$

Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \dots$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \dots$
- For each state $s$ visited in episode $i$
  - for first time $t$ that state $s$ is visited in episode $i$
    - Increment counter of total first visits: $N(s) = N(s) + 1$
    - Increment total return $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^\pi(s) = G(s)/N(s)$

Automated Machine Learning Hannover