Monte-Carlo Learning

Mattia Pellegrino, Ph.D Fellow
mattia.pellegrino@unipr.it

Summary

- Monte-Carlo Learning (Model free prediction)
- Monte-Carlo Learning (Model free control)

MONTE-CARLO RF

- Monte Carlo methods learn directly from episodes of experience
- Monte Carlo is classified as model-free (no knowledge of MDP transitions/rewards)
- MC learns from complete episodes
- Mac use the man return as value → value = mean return

- MC can only be applied to episodic MDPS
  - All episodes must terminate

MC POLICY EVALUATION

- We want to learn V_π from episodes of experience under policy π

$$S_1, A_1, R_2, ..., S_k \sim \pi$$

- Recap: the return is the total discounted reward:

$$G_t = R_{t+1} + \gamma R_{t+2} + ... + \gamma^{T-1} R_T$$

- Recap: value function is the expected return:

$$v_\pi(s) = \mathbb{E}_\pi \left[G_t \mid S_t = s \right]$$

- Monte-Carlo policy evaluation used empirical mean return instead of expected return

POLICY EVALUATION: FIRST ITERATION

- To evaluate state s
- The FIRST time-step t that state s is visited in an episode:
  - Counter incrementation: $N(s) \rightarrow N(s) + 1$
  - Total Return incrementation: $S(s) \leftarrow S(s) + G_t$
  - Value is the mean return: $V(s) = S(s) / N(s)$

  - By the law of large numbers $\rightarrow V(s) \rightarrow V_\pi(s)$ as $N(s) \rightarrow \infty$

POLICY EVALUATION: EVERY ITERATION

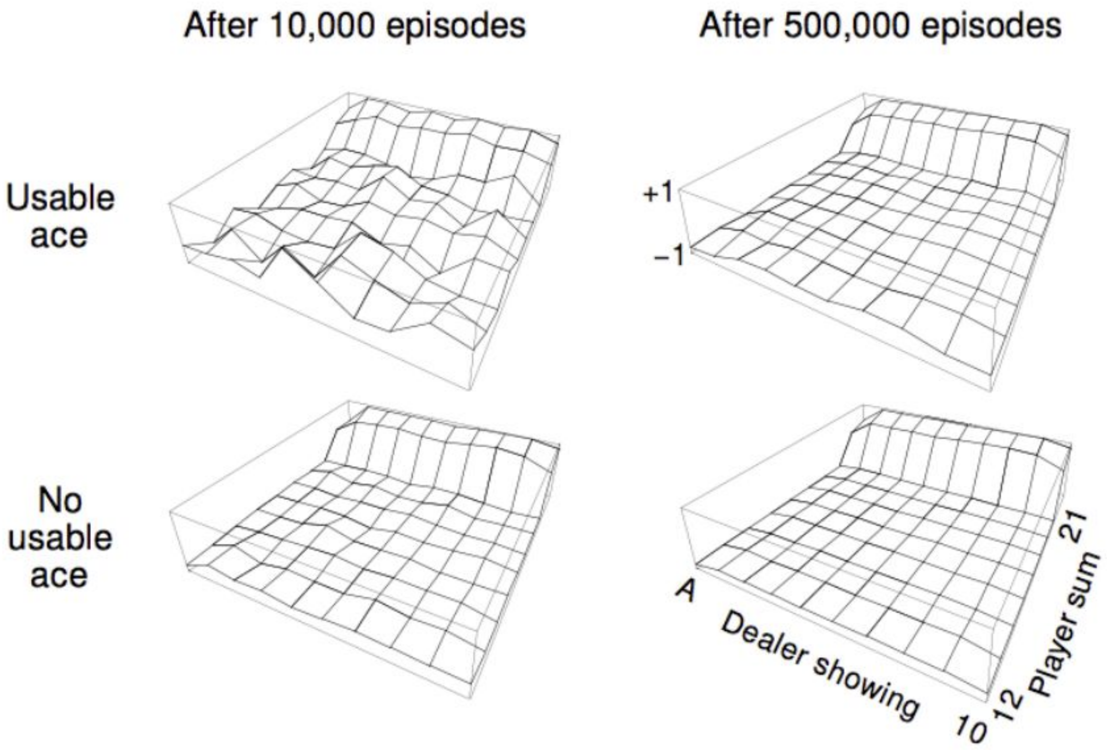
- To evaluate state s
- EVERY time-step t that state s is visited in an episode:
  - Counter incrementation: $N(s) \rightarrow N(s) + 1$
  - Total Return incrementation: $S(s) \leftarrow S(s) + G_t$
  - Value is the mean return: $V(s) = S(s) / N(s)$

  - By the law of large numbers $\rightarrow V(s) \rightarrow V_\pi(s)$ as $N(s) \rightarrow \infty$

EXAMPLE – BLACK JACK

- States (200 of them):
  - Current sum (12-21)
  - Dealer's showing card (ace-10)
  - Do I have a "useable" ace? (yes-no)
- Action stick: Stop receiving cards (and terminate)
- Action twist: Take another card (no replacement)
- Reward for stick:
  - +1 if sum of cards > sum of dealer cards
  - 0 if sum of cards = sum of dealer cards
  - -1 if sum of cards < sum of dealer cards
- Reward for twist:
  - -1 if sum of cards > 21 (and terminate)
  - 0 otherwise
- Transitions: automatically twist if sum of cards < 12

EXAMPLE – BLACK JACK



- Policy:
    - Stick
        - Sum of cards >= 20
    - Twist
        - Otherwise

INCREMENTAL MEAN

- The mean $\mu_1, \mu_2, \ldots$ of a sequence $x_1, x_2, \ldots$ can be computed incrementally

$$
\begin{aligned}
\mu_k &= \frac{1}{k} \sum_{j=1}^{k} x_j \\
&= \frac{1}{k} \left(x_k + \sum_{j=1}^{k-1} x_j \right) \\
&= \frac{1}{k} \left(x_k + (k-1)\mu_{k-1} \right) \\
&= \mu_{k-1} + \frac{1}{k} \left(x_k - \mu_{k-1} \right)
\end{aligned}
$$

INCREMENTAL UPDATES

- Update $V(s)$ incrementally after episode $S_1,A_1,R_2, \ldots , S_T$
- For each S_t with return G_t

$$N(S_t) \leftarrow N(S_t) + 1$$

$$V(S_t) \leftarrow V(S_t) + 1 / N(S_t) (G_t - V(S_t))$$

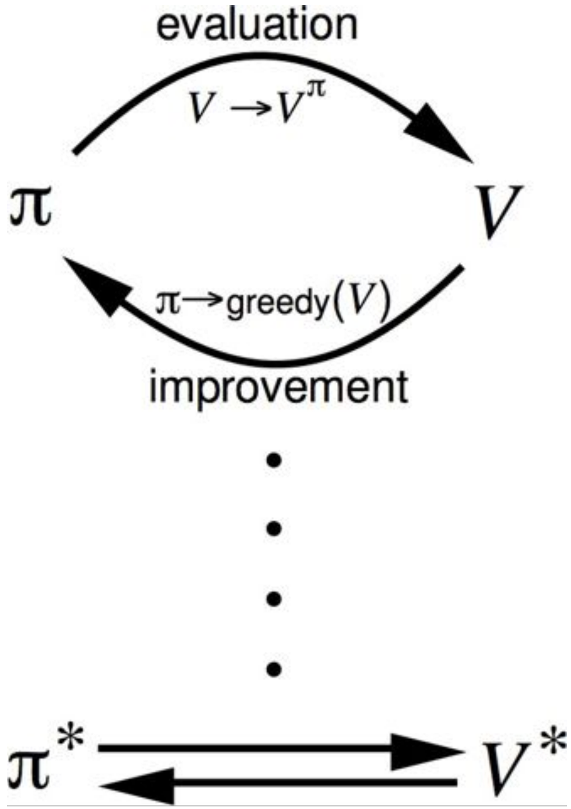
- In non-stationary problems, it can be useful to track a running mean, i.e. forget old episodes

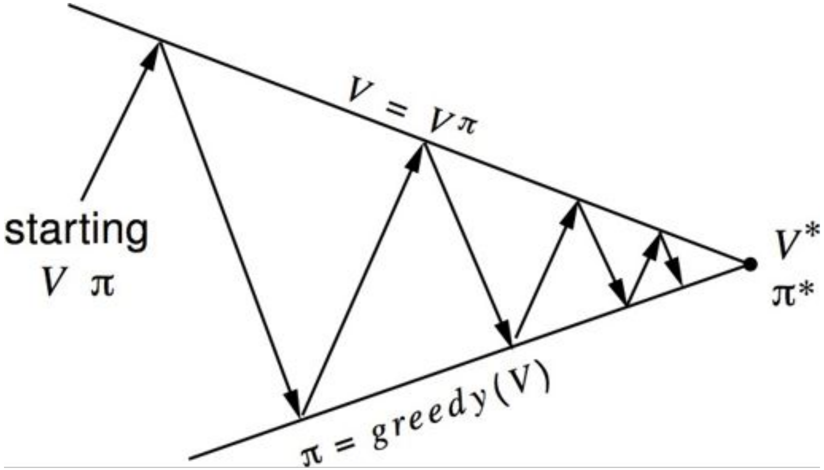
$$V(S_t) \leftarrow V(S_t) + \alpha(G_t - V(S_t))$$

MONTECARLO - MODEL CONTROL

- Monte Carlo is an On-policy learning algorithm
    - “Learn on the job”
    - Learn about policy π from experience sampled from π

POLICY ITERATION (REFRESH)

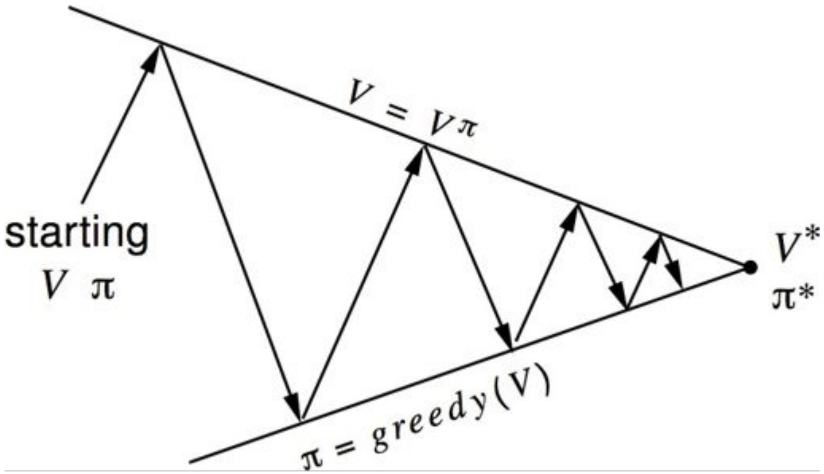




Policy evaluation → Estimate v_{π}

Policy improvement → Generate $\pi' >= \pi$

POLICY ITERATION (MONTECARLO)



Policy evaluation Monte-Carlo policy evaluation $V = v_{\pi}$?

Policy improvement: Greedy?

MONTE-CARLO - POLICY ITERATION WITH ACTION-VALUE FUNCTION

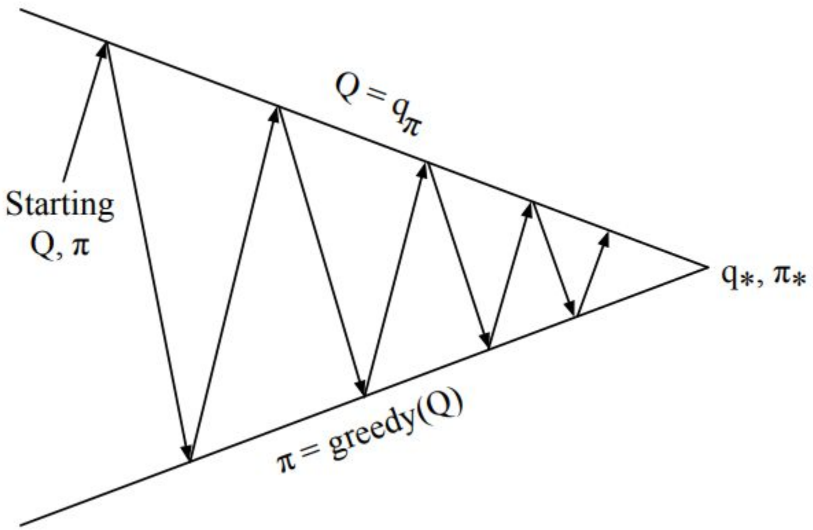
- Greedy policy improvement over V(s) requires a model of MDP

$$\pi'(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \, \mathcal{R}_s^a + \mathcal{P}_{ss'}^a V(s')$$

- Greedy policy improvement over Q(s,a) is model-free

$$\pi'(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \, Q(s, a)$$

POLICY ITERATION (MONTECARLO)



Policy evaluation Monte-Carlo policy evaluation $Q = q_\pi$

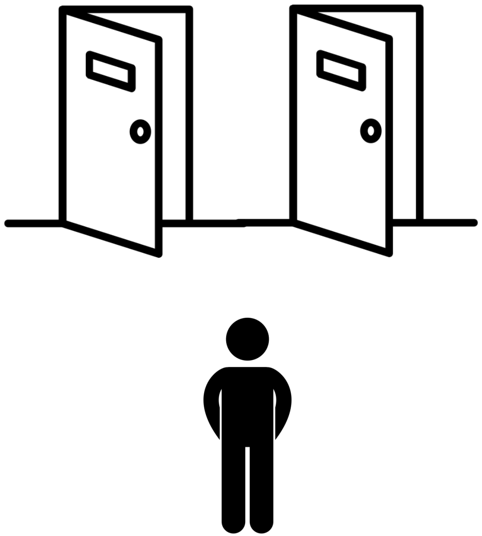
Policy improvement: Greedy?

EXAMPLE OF GREEDY ACTION SELECTION

- There are two doors in front of you

- Open left door (reward 0) → V(left) = 0
- Open left door (reward +1) → V(right) = +1
- Open left door (reward +3) → V(right) = +2
- Open left door (reward +2) → V(right) = +2

- We’ve chosen the best door?



ε-Greedy Exploration

- Simplest idea for ensuring continual exploration
- All m actions are tried with non-zero probability
- With probability 1 − ε choose the greedy action
- With probability choose an action at random

$$\pi(a|s) = \begin{cases} \epsilon/m + 1 - \epsilon & \text{if } a^* = \underset{a \in \mathcal{A}}{\operatorname{argmax}}\ Q(s,a) \\ \epsilon/m & \text{otherwise} \end{cases}$$

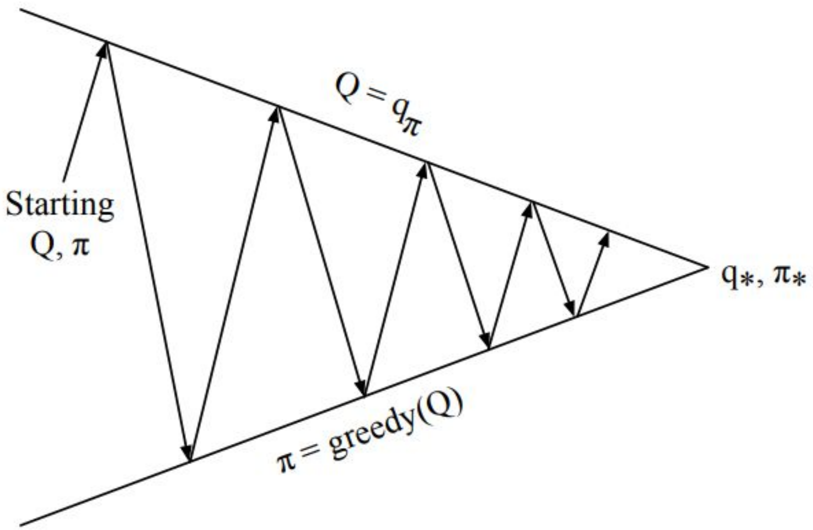
ε-Greedy Improvement

- Theorem
  - For any ε-greedy policy π, the ε-greedy policy π′ with respect to q_π is an improvement, $v_{\pi'}(s) \geq v_\pi(s)$

$$
\begin{aligned}
q_\pi(s, \pi'(s)) &= \sum_{a \in \mathcal{A}} \pi'(a|s) q_\pi(s, a) \\
&= \epsilon/m \sum_{a \in \mathcal{A}} q_\pi(s, a) + (1 - \epsilon) \max_{a \in \mathcal{A}} q_\pi(s, a) \\
&\geq \epsilon/m \sum_{a \in \mathcal{A}} q_\pi(s, a) + (1 - \epsilon) \sum_{a \in \mathcal{A}} \frac{\pi(a|s) - \epsilon/m}{1 - \epsilon} q_\pi(s, a) \\
&= \sum_{a \in \mathcal{A}} \pi(a|s) q_\pi(s, a) = v_\pi(s)
\end{aligned}
$$

- From policy improvement theorem $v_{\pi'}(s) \geq v_\pi(s)$

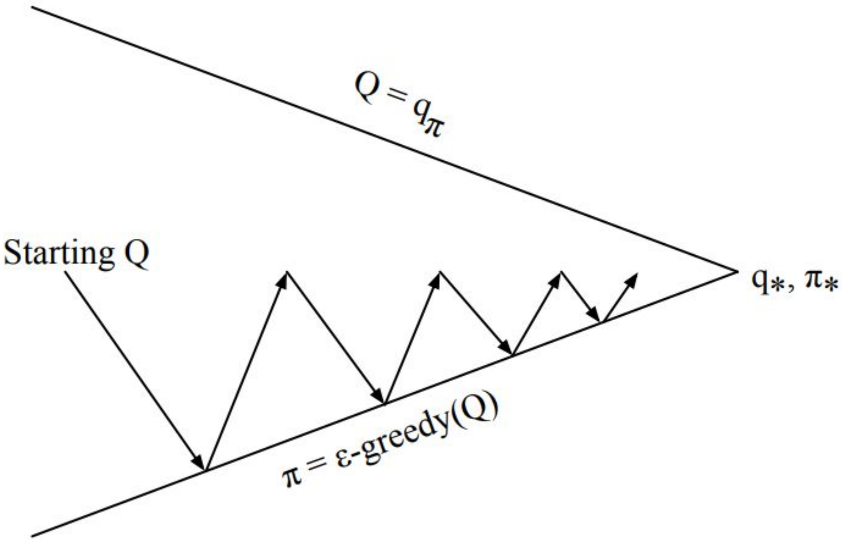
POLICY ITERATION (MONTE-CARLO)



Policy evaluation Monte-Carlo policy evaluation $Q = q_\pi$

Policy improvement: ε-Greedy Policy Improvement

MONTE-CARLO CONTROL



Policy evaluation Monte-Carlo policy evaluation $Q \approx q_\pi$

Policy improvement: ε-Greedy Policy Improvement

GLIE

- Greedy in the Limit with Infinite Exploration (GLIE)
  - All state-action pairs are explored infinitely many times

$$\lim_{k \to +\infty} N_k(s, a) = \infty$$

  - The policy converges on a greddy policy

$$\lim_{k \to +\infty} \pi_k(a|s) = 1(a = argmax_{a' \in A} Q_k(s, a'))$$

GLIE - MONTE-CARLO

- Sample kth episode using $\pi : S_1, A_1, R_2, \dots, S_T \sim \pi$
- For each state S_t and action A_t in the episode

  $N(S_t, A_t) \leftarrow N(S_t, A_t) + 1$

  $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + 1/N(S_t, A_t)\,(G_t - Q(S_t, A_t))$

- Improve policy based on new action-value function

  $\varepsilon \leftarrow 1/k$

  $\pi \leftarrow \varepsilon\text{-greedy}(Q)$

- Theorem: GLIE Monte-Carlo converges to the optimal action-value function

  $Q(s,a) \rightarrow q_*(s,a)$

MONTE-CARLO CONTROL (BLACKJACK)