Reinforcement Learning
    Learning and Value-based Methods

Reinforcement Learning

Learning and Value-based Methods

- An RL agent may include one or more of these components
  - Policy : agent's behavior func:on
  - Value func : how good is a state and/or ac:on
  - Model: agent's representa:on of the environment
- Markov Decision Processes $$ \langle\mathcal{S}, \mathscr{A}, \mathscr{P}, \mathscr{R}, \gamma\rangle $$： RL 的学习环境
  - $ \mathcal{S} $ -a set of states
  - $\mathscr{A}$- a set of ac:ons
  - $\mathscr{P}$ - transi:on probability function, $\mathscr{P}{s s^{\prime}}^{a}=\mathbb{P}\left[S{t+1}=s^{\prime} \mid S{t}=s, A{t}=a\right]$
  - $\mathscr{R}$ - reward function, $\mathscr{R}{s}^{a}=\mathbb{E}\left[R{t+1} \mid S{t}=s, A{t}=a\right]$
  - γ - discounting factor for future reward
  - Polity π : $\pi(a \mid s)=\mathbb{P}\left[A{t}=a \mid S{t}=s\right]$
- **Value Function** :

  $\begin{aligned} v{\pi}(s) &=\mathbb{E}{\pi}\left[G{t} \mid S{t}=s\right] \ &=\mathbb{E}{\pi}\left[R{t+1}+\gamma R{t+2}+\gamma^{2} R{t+3}+\ldots \mid S{t}=s\right] \ &=\mathbb{E}{\pi}\left[R{t+1}+\gamma\left(R{t+2}+\gamma R{t+3}+\ldots\right) \mid S{t}=s\right] \ &=\mathbb{E}{\pi}\left[R{t+1}+\gamma G{t+1} \mid S{t}=s\right] \ &=\mathbb{E}{\pi}\left[R{t+1}+\gamma v{\pi}\left(S{t+1}\right) \mid S_{t}=s\right] \end{aligned}$

- $v{\pi}(s)=\sum{a \in \mathscr{A}} \pi(a \mid s) q_{\pi}(s, a)$
- $q{\pi}(s, a)=\mathscr{R}{s}^{a}+\gamma \sum{s^{\prime} \in \mathcal{S}} \mathscr{P}{s s^{\prime}}^{\alpha} v_{\pi}\left(s^{\prime}\right)$
- $v{\pi}(s)=\sum{a \in \mathscr{A}} \pi(a \mid s)\left(\mathscr{R}{s}^{a}+\gamma \sum{s^{\prime} \in \mathcal{S}} \mathscr{P}{s s^{\prime}}^{a} v{\pi}\left(s^{\prime}\right)\right)$
- $q{\pi}(s, a)=\mathscr{R}{s}^{a}+\gamma \sum{s^{\prime} \in \delta} \mathscr{P}{s s^{\prime}}^{a} \sum{a^{\prime} \in \mathscr{A}} \pi\left(a^{\prime} \mid s^{\prime}\right) q{\pi}\left(s^{\prime}, a^{\prime}\right)$
- **Optimal Value Function**
  - $v{*}(s)=\max _{\pi} v{\pi}(s)$
  - $q{*}(s, a)=\max _{\pi} q{\pi}(s, a)$
  - $v{*}(s)=\max _{a} q{*}(s, a)$
  - $q{*}(s, a)=\mathscr{R}{s}^{a}+\gamma \sum{s^{\prime} \in \delta} \mathscr{P}{s s^{\prime}}^{a} v_{*}\left(s^{\prime}\right)$
  - $v{*}(s)=\max _{a}\left(\mathscr{R}{s}^{a}+\gamma \sum{s^{\prime} \in \delta} \mathscr{P}{s s^{\prime}}^{a} v_{*}\left(s^{\prime}\right)\right)$

- $q_{*}(s, a)=\mathscr{R}_{s}^{a}+\gamma \sum_{s^{\prime} \in \mathcal{S}} \mathscr{P}_{s s^{\prime}}^{a} \max _{a^{\prime}} q_{*}\left(s^{\prime}, a^{\prime}\right)$
- Dynamic Programming：假设我们已知 MDP
  - Policy Evaluation：利用迭代法计算给定 policy π的 value function $v^{k+1}=\mathscr{R}^{\pi}+\gamma \mathscr{P}^{\pi} v^{k}$
  - Policy Improvement：利用贪心法

    $\pi^{\prime}(s)=\underset{a \in \mathscr{A}}{\operatorname{argmax}} q_{\pi}(s, a)$

    $q_{\pi}\left(s, \pi^{\prime}(s)\right)=\max _{a \in \mathscr{A}} q_{\pi}(s, a) \geq q_{\pi}(s, \pi(s))=v_{\pi}(s)$
  - value iteration: update policy every iteration $v_{*}(s) \leftarrow \max _{a \in \mathscr{A}}\left(\mathscr{R}_{s}^{a}+\gamma \sum_{s^{\prime} \in \mathcal{S}} \mathscr{P}_{s s^{\prime}}^{a} v_{*}\left(s^{\prime}\right)\right)$
  - 可以证明，这样迭代下去，最后 value function 会以γ的比例线性收敛到最优解
  - 每个迭代复杂度是$O(m n ^2)$ ， m 是action 数目，n 是 state 数目
- Monte-Carlo Methods
  - 给定