Reinforcement Learning:

MDP: Tuple $\langle S, A, P, R, \gamma \rangle$ | S: {States} A: {Actions}, P: transition prob $P^a_{ss'} = P(S_{t+1} = s' \mid S_t = s, A_t = a)$ (prob enter state), R: Reward function $R^a_s = E(R_{t+1} \mid S_t = s, A_t = a)$ (feedback signal tell agent how do in env), γ: discount factor

Return: Accumulated Reward for #steps → total reward over trajectory (regardless if flip/stuck) | G_t: discounted Reward, $G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots = \sum_{k=0}^{R} \gamma^k R_{t+k}$ (↳ Gaurantee Math. Convergence) | $\gamma = 0$: near sighted, $\gamma = 1$: farsighted ↳ 0.99 tipical

Policy Function: $\pi : S \to A$ w/ obj. to max G_t w/ $\pi^* = \arg\max_{\pi} E\left[\sum_{t \geq 0} \gamma^t R_t \mid \pi\right]$ → $E(x)$ b/c prob policy take action & prob enter state

Value Functions: How good is a state s, w/ $v(s)$ = Expected Cummulative reward from following π from s (on avg how good is state) | policy is stochastic ⇒ get diff traj. even if start same loc

$$v^*(s) = \mathbb{E}\left[\sum_{t \geq 0} \gamma^t r_t \mid S_t = s, \pi\right] = \underbrace{\frac{1}{m} \sum_{k=0}^{N} \sum_{t=0}^{T} \gamma^t r_t}_{\text{Exp reward over all traj}}$$

norm term $\approx N \cdot T$

Bellman: How to learn Value Function for whole env.

$$v^*(s) = \underbrace{E_x}_{\text{Exp w.r.t. to policy}}\Big[\underbrace{r_t}_{\text{reward current step}} + \underbrace{\sum_{t \geq 1} \gamma^t r_t}_{\text{val func next ts}} \mid S_0 = s, \pi\Big] = \underbrace{E_x\Big[\underbrace{r_t}_{\text{expected immediate reward}} + \underbrace{\gamma}_{\text{Discount Reward}} \underbrace{v(s_{t+1})}_{\text{enter new state}} \mid S_0 = s_t, \pi\Big]}_{\text{Bellman EQ}}$$

value map at next iter for state s: $v_{k+1}(s) = \sum_{a \in A} \pi(a|s)\left[R^a_s + \gamma \sum_{s' \in S} P^a_{ss'} v_k(s')\right]$

mult act + exec prob mult states

$P^a_{s,s'}$: Prob trans $s \to s'$ using action a

R^a_s: immediate reward taking action a in state S

$v_k(s')$: future reward state s'

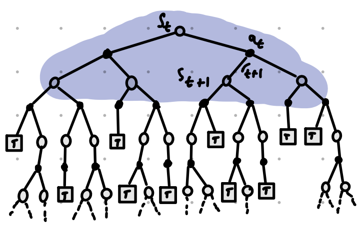
$\pi(a|s)$: prob take a in state s

Monte Carlo: $v_0 \to v_1 \to \dots \to v_\infty$. Check all actions, following 1 policy until terminates. $v_{k+1} = v_k(s_t) + \alpha(G_t - v_k(s_t))$ → [Consider avg return for all traj. start at S based on rolling avg

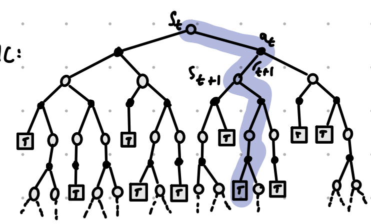
$$v_{k+1}(s) = \frac{1}{T}\sum_{i=1}^{T} G_i = \frac{1}{T}G_T + \frac{1}{T}\sum_{i=1}^{T-1} G_i = \frac{1}{T}G_T + \frac{T-1}{T}\frac{1}{T-1}\sum_{i=1}^{T-1} G_i = \frac{1}{T}G_T - \frac{1}{T}v_k(s_t) + v_k(s_t) \Rightarrow v_{k+1}(s) = v_k(s) + \alpha(G_T - v_k(s))$$

Temporal Distance: Execute 1 action, know value next state, then backup: $v_k(s_t) \leftarrow v_k(s_t) + \alpha\big(\underbrace{R_{t+1} + \gamma v(s_{t+1})}_{\text{Equiv to } G(t)\text{: Im reward + disc. factor future}} - v(s_t)\big)$

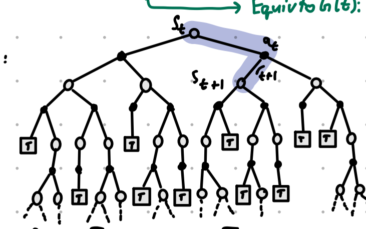
DP:



MC:



TD:



$S_0 \xrightarrow[\pi(S_0)]{a_0} o \xrightarrow{\tau} S_1 \xrightarrow{a_1} o \xrightarrow{\tau} \dots \to S_{H-1} \xrightarrow{a_{H-1}} o \xrightarrow{\tau} S_H$

τ: transition func of env

learning App: $a \leftarrow \pi(s)$ learn using stoch. gausian: output μ, σ + sample action

VPG:

Initialize π_θ w/ θ

for $j = 1$ to H:

- rollouts ← Execute π_θ m times, get $\{s^i_0, a^i_0, r^i_0, s^i_1, a^i_1, r^i_1, \dots, s_t\}$
- rollouts ← Replace rew. w/ ret, get $\{s^i_0, a^i_0, R^i_0, s^i_1, a^i_1, R^i_1, \dots, s_t\}$ (prevent Credit assign. prob. / future Rew)
- $b \leftarrow$ Calc Baseline w/ $\frac{1}{m}\sum_{i=1}^{M} R(\tau^i)$
- rollouts ← Replace R w/ $A = R - b$, get: $\{s^i_0, a^i_0, A^i_0, s^i_1, a^i_1, A^i_1, \dots, s_t\}$
- $\theta \leftarrow \theta - \frac{1}{m}\sum_{i=1}^{M}\sum_{t=0}^{H-1} \nabla_\theta \log \pi_\theta(a^i_t; s^i_t) A^i_t$ A: advantage

RL Gradient $a \leftarrow \pi(s)$ learn using stoch. gausian: output μ, σ + sample action

max reward:
$$\max_\theta E\left[\sum_{t=0}^{H} R(s_t, a_t) \mid \pi_\theta\right] = \max_\theta \sum_\tau \underbrace{P(\tau, \theta)}_{\text{prob trans + get act}} \underbrace{R(\tau)}_{\text{reward}} = \underbrace{\sum_\tau \nabla_\theta P(\tau, \theta) R(\tau)}_{\text{Expected Reward over } \tau \text{ traj}} = \sum_\tau P(\tau, \theta) \nabla_\theta \ln(P(\tau, \theta)) R(\tau) = E_{P(\tau,\theta)} \nabla_\theta \left[\sum_{t=0}^{H-1} \log \pi(a_t \mid s_t, \theta)\right] R(\tau)$$

$$P(\tau, \theta) = \prod_{t=1}^{H-1} \underbrace{\tau(s_{t+1} \mid a_t, s_t)}_{\text{with what prob reach next state on current action}} \pi(a_t \mid s_t, \theta) \Rightarrow \nabla_\theta \log(P(\tau, \theta)) = \underbrace{\nabla_\theta \sum_{i=1}^{H-1} \log(\tau(s_{t+1} \mid a_t, s_t))}_{\text{is } 0} + \sum_{i=1}^{H-1} \log(\pi(a_t \mid s_t, \pi_\theta))$$