— Review

- First order MP: $P(X_t | X_{t-1} \cdots X_0) = P(X_t | X_{t-1})$
- Second order " : " " $= P(X_t | X_{t-1}, X_{t-2})$
- $P\left((S_{t+1}, R_{t+1}) \mid (S_t, A_t), (S_{t-1}, A_{t-1}) \cdots (S_0, A_0)\right)$

$= P\left((S_{t+1}, R_{t+1}) \mid (S_t, A_t)\right) \equiv P(s', r \mid s, a)$

$\Rightarrow P(s' \mid s, a) = \sum_{r \in R} P(s', r \mid s, a)$

$P(r \mid s, a) = \sum_{s \in S} P(s', r \mid s, a)$

- Deepmind uses past 4 states (4th order MP)

— Markov decision process:

- States: s, action: a, reward: r, prob: $P(s', r \mid s, a)$, discount factor
- Policy: π; not part of MDP, it's a algorithm agent uses to navigate in env
- Value func & policy form the solution
- Policy cannot be quantified except optimal policy.
- State diagram

- State transition prob: $P(s'|s,a)$

  it only represent immediate state, not a good rep of env

- Total reward: $G(t) = \sum_{t=1}^{\infty} R(t+t)$ for future

- Discount factor, γ: $G(t) = \sum_{t=0}^{\infty} \gamma^t R(t+t+1)$

  normally $\gamma \approx 0.9$

- Value function:

$$V_\pi(s) = E_\pi[G(t) \mid S_t = s]$$

$$= E_\pi\left[\sum_{t=0}^{\infty} \gamma^t R(t+t+1) \mid S_t = s\right]$$

$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)\{r + \gamma V_\pi(s')\}$$

- If $V_{\pi_1}(s) > V_{\pi_2}(s)$, $\pi_1 > \pi_2$

Optimal value func: $V_*(s) = \max_{\forall \pi} \{V_\pi(s)\}$

Optimal policy isn't unique.

- State value func:

$$V_\pi(s) = E_\pi[G(t) \mid S_t = s]$$

Action value func:

$$Q_\pi(s,a) = E_\pi[G(t) \mid S_t = s, A_t = a]$$

- Since the project has infinite states, we will use police gradient algorithm