Function Approximate:

Go → $|S| = 10^{170}$

Helicopter
Drone
} → state is cts.

Instead of represent $V(s)$, $Q(s,a)$ in a look up table/ Tabular case

we use a function approximation.

$\hat{V}(s; \vec{w})$ the value fcn is determined by parameter w.

$\hat{q}(s,a; \vec{w})$

- Even if the data doesn't see the state, using fcn approx will generalise to unseen state.

1). $s \rightarrow [\ w\ ] \rightarrow V(s; w)$
input; w is determined by w

$s \rightarrow$
$a \rightarrow$ $[\ w\ ] \rightarrow Q(s,a; w)$

$s \rightarrow$ $[\ \ ]$ $\rightarrow Q(s,a_1; w)$, $Q(s,a_2; w)$, ⋮, $Q(s,a_{|A|}; w)$

Type of approx: - linear combination of feature. ⟸
- NN (nonlinear fcn approx) ⟸ → linear approx.
- Fourier / wavelet bases. ⟸

Q: How to determine the parameter w?

1. Recall SGD:

$$\min_w \quad J(w)$$

$$w_{k+1} = w_k - \alpha \nabla_w J(w_k)$$

$$\min_w \quad \mathbb{E}_{x \sim p(x)} J(w, x)$$

$$w_{k+1} = w_k - \alpha \nabla_w J(w_k, x_k) \Leftarrow \text{stochastic GD.}$$

2. Imagine in policy prediction case, we know $V^\pi(s)$

$$\min \quad J(w) = \mathbb{E}_s \left[\frac{1}{2} \left(V^\pi(s) - \hat{V}(s; w) \right)^2 \right]$$

GD: $$w_{k+1} = w_k + \alpha_k \mathbb{E} \left[(V^\pi(s) - \hat{V}(s; w_k)) \nabla_w \hat{V}(s; w_k) \right]$$

SGD: $$w_{k+1} = w_k + \alpha_k (V^\pi(s_t) - \hat{V}(s_t; w_k)) \nabla_w \hat{V}(s_t; w_k)$$

e.g. $\hat{V}(s;w)$ is represented as $\{x_j(s)\}_{j=1}^n$

$$\hat{V}(s;w) = \sum_{j=1}^{n} w_j x_j(s) = (x_1(s), \cdots, x_n(s)) \begin{pmatrix} w_1 \\ \vdots \\ w_n \end{pmatrix} = (x(s))^T w \quad \downarrow \in \mathbb{R}^n$$

GD: $w_{k+1} = w_k + \alpha_k \mathbb{E}\left[\left(V^\pi(s) - x(s)^T w_k\right) x(s) \right]$

SGD: $w_{k+1} = w_k + \alpha_k \left(V^\pi(s_t) - x(s_t)^T w_k\right) x(s_t)$

e.g.: Tabular is a special case of linear approx:

with $x_i(s) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \leftarrow i\text{-component} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$.

$$\hat{V}(s;w) = \sum_{i=1}^{|S|} x_i(s) w_i$$

3. However, in RL, we don't really know the true $V^\pi(s)$.

1). We can substitute $V^\pi(s)$ by an unbiased estimate $G_t^{(\infty)}$

$$G_t^{(\infty)} = r_t + \gamma r_{t+1} + \cdots + \gamma^T r_{t+T}$$

MC learning: $w_{k+1} = w_k + \alpha_k \left(G_t^{(\infty)} - \hat{V}(s_t, w_k) \right) \nabla_w \hat{V}(s_t, w_k)$

$\downarrow$

$\mathbb{E}[G_t^{(\infty)}] = V^\pi(s_t)$

↳ an unbiased estimate for the true $\nabla J(w)$.

2). TD learning: substitute $G_t^{(1)} = r_t + \gamma \hat{V}(S_{t+1}; w_k)$.

$$w_{k+1} = w_k + \alpha_k \left(G_t^{(1)} - \hat{V}(S_t; w_k) \right) \nabla_w \hat{V}(S_t, w_k)$$

* Note: here $G_t^{(1)}$ is not necessarily an unbiased estimate for $V(S_t)$.

3). TD(λ): $w_{k+1} = w_k + \alpha_k \left(G_t^{\lambda} - \hat{V}(S_t; w_k) \right) \nabla_w \hat{V}(S_t, w_k)$

$$G_t^{\lambda} = (1-\lambda) \sum_{i=0}^{\infty} \lambda^i G_t^{(i)}$$

e.g. linear approx:

$$w_{k+1} = w_k + \alpha_k \left(r_t + \gamma \, x(S_{t+1})^T w_k - x(S_t)^T w_k \right) x(S_t)$$

$$w_{k+1} = w_k + \alpha_k \underbrace{d_t}_{\text{TD: } r_t + \gamma \hat{V}(S_{t+1}; w_k) - \hat{V}(S_t; w_k)} x(S_t)$$

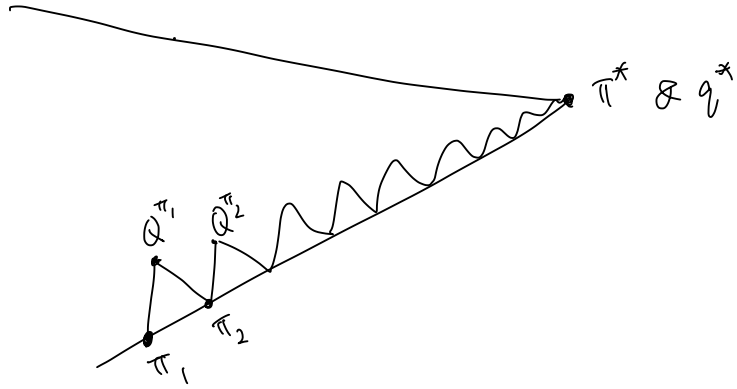
"Convergence Thm"

MC → converges to a local min for both linear & nonlinear approx

TD → converge (close to) a local min for linear approx.
(close to: γ)

TD(λ) → similar as TD.

function approximation for control:



Algo: update $q(s,a;w)$ with one step update + ε-greedy policy improvement

(SARSA in fcn Approx).

Initialize s, a, w

→ Observe s', r

$a' \sim \varepsilon$-greedy π from $\hat{q}(s,a;w)$.

$w \leftarrow w + \alpha\left(r + \gamma \hat{q}(s',a';w) - \hat{q}(s,a;w)\right) \nabla_w \hat{q}(s,a;w)$

⑧ Gradient TD.

- Prediction:

$$V(s) = r(s) + \gamma\, \mathbb{E}[V(S_{t+1}) \mid S_t = s]$$

$$\underset{\Delta}{\Pi}\ V(s) = r(s) + \gamma\, \mathbb{E}[V(S_{t+1}) \mid S_t = s]$$

Bellman

$$\Pi V = r + \gamma P^{\pi} V$$

$\Updownarrow$ has contraction property.

$$V^{\pi}(s) = \underset{\Delta}{\Pi} V^{\pi}(s)$$ ← view $V^{\pi}(s)$ as the fixed point of the operator.

- Instead of view the solution $V^\pi(s)$ as the fixed pt of Π.
  we can view it as the minimizer of $\int (V(s) - \Pi V(s))^2 ds$.

- When $V_\theta(s)$ is parametrized by $\theta \in \mathbb{R}^d$

$$\min_\theta \mathop{\mathbb{E}}_s \left(r(s) - \gamma \mathbb{E}[V_\theta(s_{t+1}) \mid s_t = s] - V_\theta(s) \right)^2$$

- One can solve it by GD.

$$\theta_{k+1} = \theta_k - \alpha_k \underbrace{\mathop{\mathbb{E}}_s \left[r(s) - \gamma \mathbb{E}[V_\theta(s_{t+1}) \mid s_t = s] - V_\theta(s) \right] \left[-\gamma \mathbb{E}[\nabla_\theta V_\theta(s_{t+1}) \mid s_t = s] - \nabla_\theta V_\theta(s) \right]}_{(*)}$$

- SGD:

  unbiased estimate for $(*)$

$$\left(r(s_t) - \gamma \mathbb{E}[V_\theta(s_{t+1}) \mid s_t] - V_\theta(s_t) \right) \left(-\gamma \mathbb{E}[\nabla_\theta V(s_{t+1}) \mid s_t] - \nabla_\theta V(s_t) \right)$$

$$= \left[r(s_t) - V_\theta(s_t) \right] \left(-\gamma \mathbb{E}[\nabla_\theta V(s_{t+1}) \mid s_t] \right) - \gamma \mathbb{E}[V_\theta(s_{t+1}) \mid s_t] \left(-\nabla_\theta V(s_t) \right)$$

$$+ \gamma^2 \mathbb{E}[V_\theta(s_{t+1}) \mid s_t] \mathbb{E}[\nabla_\theta V(s_{t+1}) \mid s_t]$$

unbiased $\approx$ $\left(r(s_t) - V_\theta(s_t) \right) \left(-\gamma \nabla_\theta V(s_{t+1}) \right) - \gamma V_\theta(s_{t+1}) \left(-\nabla_\theta V(s_t) \right)$

$$+ \gamma^2 V_\theta(s_{t+1}) \nabla_\theta V(s'_{t+1})$$

where s_{t+1} & s'_{t+1} are two independent sample starting from s_t.

① when the underlying dynamics is deterministic.

$$S_{t+1} = S'_{t+1}$$

② when the underlying dynamics is stochastic, when we only given one trajectory data,

$\{S_t\}_{t=0}^{T}$, then it is hard to find two samples from S_t

"Double sampling problem".

One way to avoid "DS"

$$\min_\theta \mathop{\mathbb{E}}_s \left(r(s) - \gamma \mathbb{E}[V_\theta(S_{t+1}) \mid S_t = s] - V_\theta(s)\right)^2$$

$$\min_\theta \quad f^2(\theta)$$

$$\min_\theta \max_y \quad f(\theta)\, y - \frac{1}{2} y^2$$

$$\min_\theta \max_y \mathop{\mathbb{E}}_s \left(r(s) - \gamma \mathbb{E}[V_\theta(S_{t+1}) \mid S_t = s] - V_\theta(s)\right) y - \frac{1}{2} y^2$$

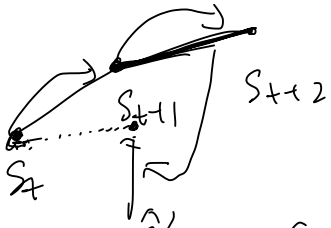
given fixed y:

$$\theta_{k+1} = \theta_k - \alpha_k \left[\left(-\gamma\, \mathbb{E}[\nabla V_\theta(S_{t+1}) \mid S_t = s] - \nabla_\theta V_\theta(s)\right) y - \frac{1}{2} y^2\right]$$

$$\begin{cases} \theta_{k+1} = \theta_k - \alpha_k \left[\left(-\gamma \nabla V_\theta(s_{t+1}) - \nabla_\theta V_\theta(s_t) \right) y - \frac{1}{2} y^2 \right] \\ y_{k+1} = y_k + \beta_k \left(r(s) - \gamma V_\theta(s_{t+1}) - V_\theta(s_t) \right) y_k - \frac{1}{2} y_k^2 \end{cases}$$

Another way: Borrowing from the future.

$s_t, s_{t+1}, s_{t+2}, s_{t+3}, \cdots$



$$\hat{s}_{t+1} = s_t + (s_{t+2} - s_{t+1})$$

↓

This could be a good approximation when the transition dynamics is smooth.

$$ds_t = \mu(s_t, a_t)dt + \sigma(s_t, a_t)dB_t$$

$$\|\nabla \mu\|, \|\sigma\| \le L$$

$$\theta_{k+1} = \theta_k - \alpha_k \left(r(s_t) - \gamma V_\theta(s_{t+1}) - V_\theta(s_t) \right) \left(-\gamma \nabla_\theta V(s_t + s_{t+2} - s_{t+1}) - \nabla_\theta V(s_t) \right)$$

— control:

$$\min_\theta \mathbb{E}_{s,a} \left(r(s,a) - \gamma \max_a \mathbb{E}\left[Q_\theta(s_{t+1}, a) \mid s_t = s, a_t = a \right] - Q_\theta(s,a) \right)^2$$

Convegence for fun Approx.

(Prediction)	Tabular	linear	non-linear
MC	✓	✓	✓
TD(0)	✓	✓	X
TD(λ)	✓	✓	X
▽TD	✓	✓	✓

(control)			
MC	✓	✓	X
SARSA	✓	✓	X
Q-learning	✓	X	X
▽TD	✓	✓	X

- Disadvantage of Value-based method.

$\max_a Q_\theta(s, a)$ → could be hard to solve when $|A| \gg 1$ or its action space

- Policy ∇:

Value-based Algo	Policy-based
- learnt Value fcn	- No value fcn
- Implicit policy based on value fcn	- learn policy.