Dynamic Programming

Algos for optimal policies given perfect environment model as MDP. Assume discrete state and action spaces.

- Online, iterative approximations
- $V_{k+1}(s) = \sum_a \pi(a|s) \sum_{s'} p(s', r|s, a)\left[r + \gamma V_k(s')\right]$, $V_k = V_\pi$ fixed point. **Full backup operation**: new value: old successor states + expected immediate rewards for all one-step transitions
- Sweep through state space. • Stop when $|V_{k+1}(s) - V_k(s)| \leq \epsilon$
- **Policy improvement theorem**: $Q_\pi(s, \pi'(s)) \geq V_\pi(s) \longrightarrow V_{\pi'}(s) \geq V_\pi(s)$ $\forall s \in S$. 'Better to select a in s, and then π'; than π all the time!
  - Stochastic policies: $Q_\pi(s, \pi'(s)) = \sum \pi'(a|s) q_\pi(s, a)$, if ties can partition probabilities among maximal ties
- **Policy iteration**: we can iteratively obtain better policies. **Evaluation-improvement cycle**. MDP has finite policies → convergence is fast
  $\pi_0 \xrightarrow{E} V_{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} V_{\pi_1} \xrightarrow{I} \dots$ • Each evaluation step can be done iterative computation → we can truncate before full convergence
- **Value iteration**: combines one E and one I step in each pass. $V_{k+1}(s) = \max \sum p(s', R|s, a)\left[R + \gamma V_k(s')\right]$
  • Turns Bellman optimality equation into update rule • Is only max on policy eval sweep. (state space)
- **Asynchronous DP**: in-place iterative DP algos, do not require systematic sweep of state space. They backup and use whatever available values of states in any order → full backup required for convergence. Super-flexible. Mixes policy eval, iteration, value iteration, truncation
  → Allow to run the RL agent online, in real-time → focus on the interesting/relevant portion of statespace
- **Generalized policy iteration**: General idea to make policy evaluation and improvement interact.
  - VF consistent with current policy → E
  - Make π greedy wrt current VF → I
  - Stabilized → we have optimality. Policy is greedy wrt its eval fcn
  - Convergence guaranteed for MDP
  - 'Bootstrapping'
- **Efficiency**: DP is yay. Finds optimal policy in P-time even if $|\pi|$ is $\$A^s$. ? Any direct search. Bombs down 2 orders of magnitude better than linear programming. Handles millions of states

Monte Carlo Methods

Do not assume complete knowledge of environment. Generate only sample transitions. Not complete distributions of all transitions. Average sample returns. All tasks formulated as episodic. Updates only at episode termination. Sample and average returns foreach state-action pair. Returns of action depend on actions taken later in same episode. Nonstationarity wrt earlier state. Do not bootstrap.

- **MC prediction**: Estimate $V_\pi(s)$ given set of episodes by following π and passing through s. Average returns observed after visits to s.
  First-visit vs all-visits: Disregard return on 1st visit or not. Convergence because CLT, finite variances
- **MC estimation of action values**: No model → state values alone not sufficient: let's estimate Q_π. Basically same first-visit, all visits algos
  → **Issue**: If π is deterministic we only get returns for one action from each state. Other estimates will not improve with experience.
  - Maintain the exploration → impose every s-a pair as starting pair. On episodic basis, **exploring starts**.
  - Only consider stochastic policies with nonzero actions in each state.
- **MC control**: To approximate policies. $\pi \xrightarrow{Q \to q_\pi} Q$, $\pi \to \text{greedy}(Q)$ (I)
  • E like before
  • I for any Q, optimal π is one choosing action with maximal action-value $\pi(s) = \text{argmax}_a Q(s,a)$
  → P.I. theorem → $q_{\pi_k}(s, \pi_{k+1}(s)) = q_{\pi_k}(s, \text{argmax}_a q_{\pi_k}(s,a)) = v_{\pi_k}(s)$

  MC-ES algo: Returns foreach s-a pair are accumulated and averaged regardless of enforced π. Not sure if converges. But likely

How to guarantee actions selected infinitely often? (without exploring starts)

- **On-policy**: E/I decision making policy → make policies **ϵ-soft** $\pi(a|s) \geq \epsilon/|A(s)|$ $\forall s, a$ → ϵ-greedy $\subset$ ϵ-soft. PI theorem guarantees convergence. Policy-iteration works
  → Best ϵ-soft π
- **Off-policy**: E/I another policy

• OFF-POLICY, IMPORTANCE SAMPLING

WE HAVE EPISODES GENERATED BY DIFFERENT POLICY μ, $\pi \neq \mu$ IS TARGET POLICY. μ IS **BEHAVIOR POLICY**

COVERAGE ASSUMPTION: $\pi(a|s) > 0 \rightarrow \mu(a|s) > 0$. ~~USUALLY~~ μ MUST BE STOCHASTIC. π CAN BE DETERMINISTIC. π USUALLY GREEDY, μ EXPLORATORY IE ε-GREEDY

IMPORTANCE SAMPLING: ESTIMATE SAMPLES FOR A DISTRIBUTION GIVEN SAMPLES FROM ANOTHER. **IS-RATIO:** $\rho_t^T = \prod \frac{\pi(A_u|S_u)}{\mu(A_u|S_u)}$ ONLY DEPENDS ON THE 2 POLICIES!

- $V_\pi(s) - V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_t^T G_t}{|\mathcal{T}(s)|}$. $\mathcal{T}(s)$ IS SET OF TIME STEPS WE VISIT s. $T(t)$ IS 1ST TERMINATION TIME AFTER t. G_t IS RETURN UP TO $T(t)$.
  - **ORDINARY IS** UNBIASED BUT VARIANCE IS EXTREME, UNBOUNDED, MAY NOT CONVERGE IF SCALED RETURNS HAVE ∞ VARIANCE
  - $\frac{\sum_{t \in \mathcal{T}(s)} \rho_t^{T(t)} G_t}{\sum_{t \in \mathcal{T}(s)} \rho_t^{T(t)}}$ **WEIGHED IS:** BIASED, VARIANCE LIMITED, FAST TO CONVERGE

OBS! THIS IS INCREMENTAL, NO EPISODIC 'RESET'

INCREMENTAL EPISODIC METHODS:
- SAME AS NON MC ALGOS, BUT AVG RETURNS INSTEAD THAN REWARDS → IF ~~OFFLINE~~ **ON-POLICY**

IMPLEMENTATION
- **OFF-POLICY:** ORDINARY IS → USE STD INCREMENTAL METHODS BUT USE SCALED RETURNS

  WEIGHED IS → $V_{N+1} = V_N + \frac{W_N}{C_N}[G_N - V_N]$, $C_{N+1} = C_N + W_{N+1}$. • $W=1$ → CAN USE FOR ON-POLICY

- OFF-POLICY METHODS MIGHT ONLY LEARN FROM THE TAILS, AFTER LAST NONGREEDY ACTION → LEARNING COULD GET SLOW.

- **IS, TRUNCATED RETURNS:** IF HIGH DISCOUNTING, FACTORS OF RETURN SERIES γ^k ~~GO~~ MATTER LITTLE BUT INCREASE VARIANCE.
  → THINK OF γ AS DEGREE OF PARTIAL TERMINATION. • $G_T = \gamma^{T-t-1}\bar{G}_t^T + (1+\gamma)\sum_{h=t+1}^{T-1}\gamma^{h-t-1}\bar{G}_t^h$ h IS HORIZON FOR RETURNS

  FLAT PARTIAL RETURNS

- FPR ~~HAS~~ TO BE SCALED BY SIMILARLY TRUNCATED IS RATIO (SUTTON-BARTO P. 134 2 ED) → WEIGHED, ORDINARY ESTIMATOR FORMULAS

$\bar{G}_t^h = \sum_t^h R_i$