Advancd Algorithmic Analysis

Value Iteration

- $||B^k Q_1 - B^k Q_2||_\infty \leq \gamma^k ||Q_1 - Q_2||_\infty$
  - k-step value iteration is kind of like a super contraction mapping
- The greedy policy optimal in some reasonable amount of time
  - for some $t^* < \infty$ polynomial in $|S|, |A|, R_{\max} = \max_{a,s} |R(s,a)|, \frac{1}{1-\gamma}$, bits of precision of transition function such that $\pi(s) = \arg\max_a Q_{t^*}(s,a)$ is optimal
  - consequence of Cramer's Rule
- If the difference in consecutive value functions approximations is than some ϵ, the max difference between the value produced by the corresponding policy and the optimal value function is $< \frac{2\epsilon\gamma}{1-\gamma}$
- Notice that for both of the above, γ being small is good (effective horizon is ~$\frac{1}{1-\gamma}$), but this makes agent myopic.
  - Isbell: there is a trade-off between effective horizon and time to convergence in value iteration

Linear Programming

- The only way to solve MDPs in polynomial time is with linear programming
  - linear prog = optimization framework in which you can give linear contraints and linear objective function to get solution in polynomial time
- Bellman is mostly linear, but max is not
  - Why is max non-linear: $\max(x, x_0) = x_0, \max(y, x_0) = x_0$ does not imply $\max(x + y, x_0) = x_0$
  - Resolution: "Primal" representation of a linear program for solving MDPs:
    - minimize $\sum_s v_s$
    - $\forall s, a \quad v_s \geq R(s,a) + \gamma \sum_{s'} T(s,a,s') v_{s'}$

Policy Iteration

- Defn
  1. Initialize: $\forall s \;\; Q_0(s) = 0$
     Loop:
  2. Policy improvement: $\forall s \;\; \pi_t(s) = \arg\max_a Q_t(s,a) \;\; (t \geq 0)$
  3. Policy evaluation: $Q_{t+1} = Q^{\pi_t}$
- Convergence to optimal value function is exact and complete in finite time
- Converges at least as fast as VI

- Open question: convergence time?
- **Policy iteration does not get stuck in local optima!**