Computational Limits &
The LP formulation
Announcements

• HW0: due this Thursday 11:59pm
• Gradescope (please self-enroll)

• Norms for class?
  • Video:
  • Questions:
Today:

• Recap:
  • value/policy iteration + contraction

• Today: computational complexity & the linear programming approach

Question: Given an MDP $\mathcal{M} = (S, A, P, r, \gamma)$ can we exactly compute $Q^*$ (or find $\pi^*$) in polynomial time?
Recap
Define Bellman Operator $\mathcal{T}$:

Given a function $f : S \times A \mapsto \mathbb{R},$

$$\mathcal{T}f : S \times A \mapsto \mathbb{R},$$

$$(\mathcal{T}f)(s, a) := r(s, a) + \gamma \mathbb{E}_{s' \sim P(.|s, a)} \max_{a' \in A} f(s', a'), \forall s, a \in S \times A$$
Value Iteration Algorithm:

1. Initialization: $Q^0 : \|Q^0\|_\infty \in (0, \frac{1}{1 - \gamma})$

2. Iterate until convergence: $Q^{t+1} = \mathcal{T} Q^t$
Policy Iteration Algorithm:

1. Initialization: $\pi^0 : S \mapsto \Delta(A)$

2. Policy Evaluation: $Q^{\pi^t}(s, a), \forall s, a$

3. Policy Improvement $\pi^{t+1}(s) = \arg\max_a Q^{\pi^t}(s, a), \forall s$

Closed-form for PE (see 1.1.3 in Monograph)
Final Quality of the Policy (for VI):

\[ \pi^t : \pi^t(s) = \arg \max_a Q^t(s, a) \]

**Theorem:**

\[ V^{\pi^t}(s) \geq V^*(s) - \frac{2\gamma^t}{1 - \gamma} \| Q^0 - Q^* \|_\infty \forall s \in S \]

\[ \Rightarrow \]

Set \( Q^0 = 0 \). After \( t \geq \frac{\log \frac{2}{\epsilon(1 - \gamma)^2}}{1 - \gamma} \) iterations, we have:

\[ V^{\pi^t}(s) \geq V^*(s) - \epsilon \quad \forall s \in S \]

Same rate for PI.
Today
Polynomial Time & Strongly Polynomial Complexity

- Complexity to compute an exact solution given $M$. (Aside: Why?)
- Assume that basic arithmetic operations (+,-,x,÷) take unit time.
Polynomial Time & Strongly Polynomial Complexity

- Complexity to compute an exact solution given $\mathcal{M}$. (Aside: Why?
- Assume that basic arithmetic operations (+,-,x,/) take unit time.
- Polytime computation: Suppose that $(P, r, \gamma)$ in our MDP $\mathcal{M}$ is specified with rational entries, where $L(P, r, \gamma)$ is total bit-size required to specify $(P, r, \gamma)$.

\[ P \left( S = 4 \mid s \in 2, a \leq 8 \right) = \frac{10}{11} \quad (\gamma) \approx 10^{-4} \]

Can we (exactly) compute $Q^*$ in time that is polynomial in $L(P, r, \gamma)$, # states $S$, and # actions $A$. 
Polynomial Time &
Strongly Polynomial Complexity

• Complexity to compute an exact solution given $\mathcal{M}$. (Aside: Why?
  • Assume that basic arithmetic operations (+,-,x,$\div$) take unit time.
  • Polytime computation: Suppose that $(P, r, \gamma)$ in our MDP $\mathcal{M}$ is specified with rational entries, where $L(P, r, \gamma)$ is total bit-size required to specify $(P, r, \gamma)$.
    Can we (exactly) compute $Q^*$ in time that is polynomial in $L(P, r, \gamma)$, # states $S$, and # actions $A$.
• Strongly polynomial time: Suppose $(P, r, \gamma)$ is specified with real numbers. Can we compute $Q^*$ in $\text{poly}(S, A, \log(1/(1 - \gamma)))$, with no dependence on $L(P, r, \gamma)$?
Computational Complexities of our Iterative Algorithms
Value Iteration

• When the gap in the current objective value and the optimal objective value is smaller than $2^{-L(P,r,\gamma)}$, then the greedy policy will be optimal. (this is a standard argument in optimization)
Value Iteration

• When the gap in the current objective value and the optimal objective value is smaller than $2^{-L(P,r;\gamma)}$, then the greedy policy will be optimal. (this is a standard argument in optimization)
• VI:
  • needs $\log\left(\frac{1}{\epsilon(1 - \gamma)}\right)/(1 - \gamma)$ iterations to obtain an $\epsilon$ accurate solution.
• Per iteration complexity: $S^2A$
Value Iteration

- When the gap in the current objective value and the optimal objective value is smaller than $2^{-L(P, r, \gamma)}$, then the greedy policy will be optimal. (this is a standard argument in optimization)
- VI:
  - needs $\log \left( \frac{1}{(\epsilon(1 - \gamma))} \right) / (1 - \gamma)$ iterations to obtain an $\epsilon$ accurate solution.
  - Per iteration complexity: $S^2 A$
- Poly runtime? For fixed $\gamma$, VI is poly:

$$O\left( S^2 A \frac{L(P, r, \gamma) \log(1/(1 - \gamma))}{1 - \gamma} \right)$$
Value Iteration

- When the gap in the current objective value and the optimal objective value is smaller than $2^{-L(P,r,\gamma)}$, then the greedy policy will be optimal. (this is a standard argument in optimization)
- VI:
  - needs $\log\left(\frac{1}{\epsilon(1 - \gamma)}\right) / (1 - \gamma)$ iterations to obtain an $\epsilon$ accurate solution.
- Per iteration complexity: $S^2A$
- Poly runtime? For fixed $\gamma$, VI is poly:
  $$S^2A \frac{L(P, r, \gamma) \log(1/(1 - \gamma))}{1 - \gamma}$$
- Strongly poly? No
Policy Iteration

- PI Per iteration complexity: $S^3 + S^2 A$
- PI is more costly than VI per iteration.
- PI is observed to be much faster than VI to obtain an exact opt policy.
Policy Iteration

• PI Per iteration complexity: $S^3 + S^2A$
  • PI is more costly than VI per iteration.
  • PI is observed to be much faster than VI to obtain an exact opt policy.
• Poly runtime? For fixed $\gamma$, VI is poly:

$$\frac{(S^3 + S^2A) \cdot L(P, r, \gamma) \cdot \log(1/(1 - \gamma))}{1 - \gamma}$$
Policy Iteration

- PI Per iteration complexity: $S^3 + S^2A$
  - PI is more costly than VI per iteration.
  - PI is observed to be much faster than VI to obtain an exact optimal policy.
- Poly runtime? For fixed $\gamma$, VI is poly:
  $$(S^3 + S^2A) \frac{L(P, r, \gamma) \log(1/(1 - \gamma))}{1 - \gamma}$$
- Does PI compute an optimal policy in time independent of $L(P, r, \gamma)$?
  Yes! finite # of policies
Is PI a strongly poly algo?

• Does PI compute an optimal policy in time independent of $L(P, r, \gamma)$?
  • Yes: after $A^S$ iterations ($A^S$ is the number of policies)
Is PI a strongly poly algo?

- Does PI compute an optimal policy in time independent of $L(P, r, \gamma)$?
  - Yes: after $A^S$ iterations ($A^S$ is the number of policies)

- Is PI strongly polynomial?
  For fixed $\gamma$, yes:
  
  [Ye ’12] PI halts after $\frac{S^2A \log(S^2/(1 - \gamma))}{1 - \gamma}$ iterations.
## Summary Table

<table>
<thead>
<tr>
<th></th>
<th>Value Iteration</th>
<th>Policy Iteration</th>
<th>LP-based Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poly.</td>
<td>$S^2 A \frac{L(P,r,\gamma) \log \frac{1}{1-\gamma}}{1-\gamma}$</td>
<td>$(S^3 + S^2 A) \frac{L(P,r,\gamma) \log \frac{1}{1-\gamma}}{1-\gamma}$</td>
<td>?</td>
</tr>
<tr>
<td>Strongly Poly.</td>
<td>X</td>
<td>$(S^3 + S^2 A) \cdot \min \left{ \frac{A^S}{S}, \frac{S^2 A \log \frac{S^2}{1-\gamma}}{1-\gamma} \right}$</td>
<td>?</td>
</tr>
</tbody>
</table>

- VI Per iteration complexity: $S^2 A$
- PI Per iteration complexity: $S^3 + S^2 A$
Are VI and PI Polynomial Time algorithms? (technically, no)

Is there a polytime (and strongly polytime) algo for an MDP??

YES! Linear Programming
The Primal Linear Program

• We can write the Bellman equations with values rather than Q-values:

\[
V(s) = \max_a \left\{ r(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot | s, a)} \left[ V(s') \right] \right\}
\]

• An equivalent way to write the Bellman equations is as a linear program. With variables \( V \in \mathbb{R}^S \), the LP is:

\[
\begin{align*}
\min & \quad V(s_0) \\
\text{s.t.} & \quad V(s) \geq r(s, a) + \mathbb{E}_{s' \sim P(\cdot | s, a)} V(s') \quad \forall s, a \in S \times A
\end{align*}
\]

if \( V \) is feasible

\[
\Rightarrow V \geq V^*
\]
LP Runtimes and Comments

• Using a polytime LP solver, gives us a poly time algorithm.
LP Runtimes and Comments

- Using a polytime LP solver, gives us a poly time algorithm.
- [Ye, ’05]: there is an interior point algorithm (CIPA) which is also strongly polynomial.
LP Runtimes and Comments

• Using a polytime LP solver, gives us a poly time algorithm.
• [Ye, ’05]: there is an interior point algorithm (CIPA) which is also strongly polynomial.

• Relations:
  • VI is best thought of as a fixed point algorithm
  • PI is equivalent to a (block) simplex algorithm
    (Recall the simplex algo, in general, could be exp time.
    But not for MDPS, at least for fixed $\gamma$.)
## Summary Table

<table>
<thead>
<tr>
<th></th>
<th>Value Iteration</th>
<th>Policy Iteration</th>
<th>LP-based Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poly.</td>
<td>$S^2 A \frac{L(P,r,\gamma) \log \frac{1}{1-\gamma}}{1-\gamma}$</td>
<td>$(S^3 + S^2 A) \frac{L(P,r,\gamma) \log \frac{1}{1-\gamma}}{1-\gamma}$</td>
<td>$S^3 A L(P, r, \gamma)$</td>
</tr>
<tr>
<td>Strongly Poly.</td>
<td>X</td>
<td>$(S^3 + S^2 A) \cdot \min \left{ \frac{A^S}{S}, \frac{S^2 A \log \frac{S^2}{1-\gamma}}{1-\gamma} \right}$</td>
<td>$S^4 A^4 \log \frac{S}{1-\gamma}$</td>
</tr>
</tbody>
</table>

- VI Per iteration complexity: $S^2 A$
- PI Per iteration complexity: $S^3 + S^2 A$
- The LP approach is only logarithmic in $1 - \gamma$
What about the Dual LP?

• The linear programming is helpful in understanding the problem. (even though it is not used often)
• Let us now consider the dual LP.
  • It is also very helpful conceptually.
  • In some cases, it also provides a reasonable algorithmic approach

• Let us start by understanding the dual variables and the “state-action polytope”
For a fixed (possibly stochastic) policy \( \pi \), define the state-action visitation distribution \( \nu^\pi \) as:

\[
\nu^\pi(s, a) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \Pr^\pi(s_t = s, a_t = a | s_0)
\]

where \( \Pr^\pi(s_t = s, a_t = a | s_0) \) is the state-action visitation probability when we execute \( \pi \) starting at state \( s_0 \).
State-Action Visitation Measures

• For a fixed (possibly stochastic) policy $\pi$, define the state-action visitation distribution $\nu^\pi$ as:

$$
\nu^\pi(s, a) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \Pr^\pi(s_t = s, a_t = a \mid s_0)
$$

where $\Pr^\pi(s_t = s, a_t = a \mid s_0)$ is the state-action visitation probability when we execute $\pi$ starting at state $s_0$.

• We can verify that have $\nu^\pi$ satisfies, for all states $s \in S$:

$$
\sum_a \nu^\pi(s, a) = (1 - \gamma)I(s = s_0) + \gamma \sum_{s', a'} P(s \mid s', a') \nu^\pi(s', a')
$$
The “State-Action” Polytope

• Let us define the state-action polytope $K$ as follows:

$$K := \left\{ \nu \mid \nu \geq 0 \quad \text{and} \quad \sum_a \nu(s, a) = (1 - \gamma)I(s = s_0) + \gamma \sum_{s', a'} P(s \mid s', a') \nu(s', a') \right\}$$
The “State-Action” Polytope

• Let us define the state-action polytope $K$ as follows:

$$
K := \left\{ \nu \mid \nu \geq 0 \text{ and } \sum_{a} \nu(s, a) = (1 - \gamma)I(s = s_0) + \gamma \sum_{s', a'} P(s' \mid s', a')\nu(s', a') \right\}
$$
The “State-Action” Polytope

• Let us define the state-action polytope $K$ as follows:

$$K := \left\{ \nu \mid \nu \geq 0 \quad \text{and} \quad \sum_a \nu(s, a) = (1 - \gamma)I(s = s_0) + \gamma \sum_{s', a'} P(s' \mid s', a') \nu(s', a') \right\}$$

• This set precisely characterizes all state-action visitation distributions:
The “State-Action” Polytope

• Let us define the state-action polytope $K$ as follows:

$$K := \left\{ \nu \mid \nu \geq 0 \text{ and } \sum_a \nu(s, a) = (1 - \gamma)I(s = s_0) + \gamma \sum_{s', a'} P(s \mid s', a')\nu(s', a') \right\}$$

• This set precisely characterizes all state-action visitation distributions:

Lemma: $K$ is equal to the set of all feasible state-action distributions, i.e. $\nu \in K$ if and only if there exists a (possibly randomized) policy $\pi$ s.t. $\nu^\pi = \nu$
The Dual LP

\[
\max \sum_{s,a} \nu(s, a)r(s, a)
\]

s.t. $\nu \in K$

- One can verify that this is the dual of the primal LP.
- Note that $K$ is a polytope