CS533
Intelligent Agents and Decision Making
Infinite Horizon Markov Decision Problems

1. Some MDP formulations use a reward function $R(s, a)$ that depends on the action taken in a state or a reward function $R(s, a, s')$ that also depends on the result state s' (we get reward $R(s, a, s')$ when we take action a in s and then transition to s'). Write the Bellman optimality equation with discount factor β for each of these two formulations.

2. In this exercise you will prove that the Bellman Backup operator is a contraction operator.

   (a) Prove that, for any two functions f and g,

   $$|\max_a f(a) - \max_a g(a)| \leq \max_a |f(a) - g(a)|.$$

   (b) Use the above result in order to prove that the Bellman Backup operator $B[\cdot]$ is a contraction mapping. That is, prove that for any two value function V and V',

   $$||B[V] - B[V']|| \leq \beta ||V - V'||$$

   where B is the Bellman backup operator, β is the discount factor, and $|| \cdot ||$ is the max norm. By the definition of the max norm, this is equivalent to proving that for any state s,

   $$|B[V](s) - B[V'](s)| \leq \beta ||V - V'||.$$

3. Consider a trivially simple MDP with two states $S = \{s_0, s_1\}$ and a single action $A = \{a\}$. The reward function is $R(s_0) = 0$ and $R(s_1) = 1$. The transition function is $T(s_0, a, s_1) = 1$ and $T(s_1, a, s_1) = 1$. Note that there is only a single policy π for this MDP that takes action a in both states.

   (a) Using a discount factor $\beta = 1$ (i.e. no discounting), write out the linear equations for evaluating the policy and attempt to solve the linear system. What happens and why?

   (b) Repeat the previous question using a discount factor of $\beta = 0.9$.

4. The Bellman Backup operator satisfies the monotonicity property, which states that for any two value functions V and V', if $V \leq V'$, then $B[V] \leq B[V']$. Prove this monotonicity property of B.

5. In class we presented the policy iteration algorithm, which used a "greedy" policy improvement operation. That is, the improved policy π' at each iteration selected the action that maximized the one-step-look ahead value:

   $$\pi'(s) = \arg\max_{a \in A} \sum_{s' \in S} T(s, a, s') V_\pi(s')$$

   where π is the current policy.

   Consider a version of policy iteration, which uses a non-greedy policy improvement operator. This operator returns a policy π' that selects an action in each state that improves over the current action selected by π if possible. But we do not require that π' return the best action.

More formally, the non-greedy policy improvement operators returns a policy π' such that for any state s,

$$\sum_{s' \in S} T(s, \pi'(s), s') V_\pi(s') \geq \sum_{s' \in S} T(s, \pi(s), s') V_\pi(s')$$

with strict inequality when possible.

Prove that the non-greedy policy improvement operator guarantees that $V_{\pi'} \geq V_\pi$ with strict inequality when π is not optimal.