Implementation: Iterative Policy Evaluation

The pseudocode for **iterative policy evaluation** can be found below.

Iterative Policy Evaluation

Input: MDP, policy π, small positive number θ
Output: $V \approx v_\pi$
Initialize V arbitrarily (e.g., $V(s) = 0$ for all $s \in \mathcal{S}^+$)
repeat
  $\Delta \leftarrow 0$
  for $s \in \mathcal{S}$ **do**
    $v \leftarrow V(s)$
    $V(s) \leftarrow \sum_{a\in\mathcal{A}(s)} \pi(a|s) \sum_{s'\in\mathcal{S}, r\in\mathcal{R}} p(s', r|s, a)(r + \gamma V(s'))$
    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
  end
until $\Delta < \theta$;
return V

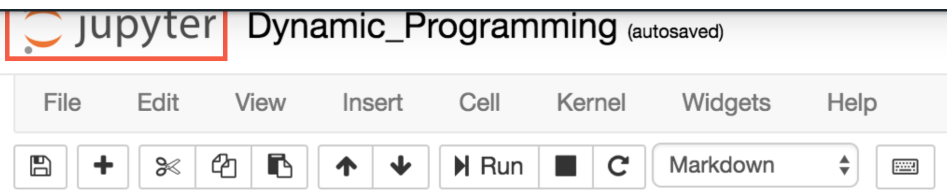
Note that policy evaluation is guaranteed to converge to the state-value function v_π corresponding to a policy π, as long as $v_\pi(s)$ is finite for each state $s \in \mathcal{S}$. For a finite Markov decision process (MDP), this is guaranteed as long as either:

- $\gamma < 1$, or
- if the agent starts in any state $s \in \mathcal{S}$, it is guaranteed to eventually reach a terminal state if it follows policy π.

Please use the next concept to complete **Part 0: Explore FrozenLakeEnv** and **Part 1: Iterative Policy Evaluation** of `Dynamic_Programming.ipynb`. Remember to save your work!

If you'd like to reference the pseudocode while working on the notebook, you are encouraged to open this sheet in a new window.

Feel free to check your solution by looking at the corresponding sections in `Dynamic_Programming_Solution.ipynb`. (*In order to access this file, you need only click on "jupyter" in the top left corner to return to the Notebook dashboard.*)



To find `Dynamic_Programming_Solution.ipynb`, return to the Notebook dashboard.

(Optional) Additional Note on the Convergence Conditions

To see intuitively *why* the conditions for convergence make sense, consider the case that neither of the conditions are satisfied, so:

- $\gamma = 1$, and
- there is some state $s \in \mathcal{S}$ where if the agent starts in that state, it will never encounter a terminal state if it follows policy π.

In this case,

- reward is not discounted, and
- an episode may never finish.

Then, it is possible that iterative policy evaluation will not converge, and this is because the state-value function may not be well-defined! To see this, note that in this case, calculating a state value could involve adding up an infinite number of (expected) rewards, where the sum may not converge.

In case it would help to see a concrete example, consider an MDP with:

- two states s_1 and s_2, where s_2 is a terminal state
- one action a (*Note: An MDP with only one action can also be referred to as a Markov Reward Process (MRP).*)
- $p(s_1, 1|s_1, a) = 1$

In this case, say the agent's policy π is to "select" the only action that's available, so $\pi(s_1) = a$. Say $\gamma = 1$. According to the one-step dynamics, if the agent starts in state s_1, it will stay in that state forever and never encounter the terminal state s_2.

In this case, $v_\pi(s_1)$ **is not well-defined**. To see this, remember that $v_\pi(s_1)$ is the (expected) return after visiting state s_1, and we have that

$$v_\pi(s_1) = 1 + 1 + 1 + 1 + \ldots$$

well-defined. As a **very optional** next step, if you want to verify this mathematically, you may find it useful to review geometric series and the negative binomial

NEXT