Policy Evaluation by Monte-Carlo (MC) Sampling

- **Monte-Carlo Policy Evaluation**
  - Sample episodes $S_0, A_0, R_1, \ldots, R_T \sim \pi$.
  - For each sampled episode:
    - Increment state counter $N(s) \leftarrow N(s) + 1$.
    - Increment total return $S(s) \leftarrow S(s) + G_t$.
  - Estimate mean return $V(s) = S(s)/N(s)$.
- Learns v_π from episodes sampled under policy π, thus **model-free**.
- Updates can be done at first step or at every time step t where state s is visited in episode.
- Converges to v_π for large number of samples.

Incremental Mean

Use definition of incremental mean μ_k s.t.

$$\begin{aligned}
\mu_k &= \frac{1}{k} \sum_{j=1}^{k} x_j \\
&= \frac{1}{k} \left(x_k + \sum_{j=1}^{k-1} x_j \right) \\
&= \frac{1}{k} \left(x_k + (k-1)\mu_{k-1} \right) \\
&= \mu_{k-1} + \frac{1}{k} \left(x_k - \mu_{k-1} \right).
\end{aligned}$$

Incremental Monte-Carlo Updates

- **Incremental Monte-Carlo Policy Evaluation**
  - For each sampled episode, for each step t:
    - $N(S_t) \leftarrow N(S_t) + 1$,
    - $V(S_t) \leftarrow V(S_t) + \alpha \left(G_t - V(S_t) \right)$.
- Can be seen as **incremental update towards actual return**.
- α can be $\frac{1}{N(S_t)}$ or more general term $\alpha > 0$.

Policy Evaluation by Temporal Difference (TD) Learning

- **TD(0):**
  - For each sampled episode, for each step t:
    - $V(S_t) \leftarrow V(S_t) + \alpha\left(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)\right).$
- **Combines sampling and recursive computation** by updating toward estimated return $R_{t+1} + \gamma V(S_{t+1})$.
- Recall $R_{t+1} + \gamma V(S_{t+1})$ from Bellman Expectation Equation, here called *TD target*.
- $\delta_t = (R_{t+1} + \gamma V(S_{t+1}) - V(S_t))$ is called *TD error*.

TD Learning with n-Step Returns

n-**Step Returns**:

- $G_t^{(1)} = R_{t+1} + \gamma V(S_{t+1})$.
- $G_t^{(2)} = R_{t+1} + \gamma R_{t+2} + \gamma^2 V(S_{t+2})$.
- $\vdots$
- $G_t^{(n)} = R_{t+1} + \gamma R_{t+2} + \ldots + \gamma^{n-1} R_{t+n} + \gamma^n V(S_{t+n})$.

n-**Step TD Learning**:

- $V(S_t) \leftarrow V(S_t) + \alpha \left(G_t^{(n)} - V(S_t) \right)$.

Exercise: How can Incremental Monte Carlo be recovered by TD(n)?

TD Learning with λ-Weighted Returns

λ-**Returns**:

- Average *n*-Step Returns using

$$G_t^{\lambda} = (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} G_t^{(n)},$$

where $\lambda \in [0,1]$.

TD(λ) Learning:

- $V(S_t) \leftarrow V(S_t) + \alpha \left(G_t^{\lambda} - V(S_t) \right).$

Exercise: How can TD(0) be recovered from TD(λ)?

Policy Optimization by Q-Learning

- **Q-Learning** [Watkins and Dayan, 1992]:
- For each sampled episode:
  - Initialize S_t.
  - For each step t:
    - Sample A_t, observe R_{t+1}, S_{t+1}.
    - $Q(S_t, A_t) \leftarrow Q(S_t, A_t)$
      $+\alpha(R_{t+1} + \gamma \max_{a'} Q(S_{t+1,a'}) - Q(S_t, A_t))$.
    - $S_t \leftarrow S_{t+1}$.
- **Q-Learning combines sampling and TD(0)-style recursive computation** for policy optimization.
- Recall $R_{t+1} + \gamma \max_{a'} Q(S_{t+1,a'})$ from Bellman Optimality Equation.

Summary: Monte-Carlo and Temporal-Difference Learning

- **MC** has **zero bias, but high variance** that grows with number of random actions, transitions, rewards.
- **TD** techniques
  - **reduce variance** due to reduction to single random action, transition, reward,
  - can learn from **incomplete episodes** and can use **online updates**,
  - introduce **bias** and use approximations which are exact only in special cases.

Summary: Value-Based/Critic-Only Methods

- All techniques discussed so far, DP, MC, and TD, focus on **value-functions**, not policies.
- Value-function is also called **critic**, thus critic-only methods.
- Value-based techniques are inherently **indirect** in learning approximate value-function and extracting near-optimal policy.
- Overview over DP, MC, and TD in [Sutton and Barto, 1998] and [Szepesvári, 2009].
- Problems:
  - Closeness to optimality cannot be quantified.
  - Continous action spaces have to be discretized in order to fit into MDP model.
  - Focus is on deterministic instead of on stochastic policies.