Reinforcement Learning Algorithms

A. LAZARIC (SequeL Team @INRIA-Lille)
ENS Cachan - Master 2 MVA
In This Lecture

- How do we solve an MDP online?

⇒ RL Algorithms
In This Lecture

- Dynamic programming algorithms require an *explicit* definition of
  - transition probabilities $p(·|x, a)$
  - reward function $r(x, a)$

- This knowledge is often *unavailable* (i.e., wind intensity, human-computer-interaction).

- *Can we relax this assumption?*
In This Lecture

- **Learning with generative model.** A black-box simulator \( f \) of the environment is available. Given \((x, a)\),

\[
f(x, a) = \{y, r\} \text{ with } y \sim p(\cdot | x, a), \ r = r(x, a).
\]

- **Episodic learning.** Multiple trajectories can be repeatedly generated from the same state \( x \) and terminating when a reset condition is achieved:

\[
(x_0^i = x, x_1^i, \ldots, x_{T_i}^i)_{i=1}^n.
\]

- **Online learning.** At each time \( t \) the agent is at state \( x_t \), it takes action \( a_t \), it observes a transition to state \( x_{t+1} \), and it receives a reward \( r_t \). We **assume** that \( x_{t+1} \sim p(\cdot | x_t, a_t) \) and \( r_t = r(x_t, a_t) \) (i.e., MDP assumption).
Outline

Mathematical Tools

The Monte-Carlo Algorithm

The TD(1) Algorithm

The TD(0) Algorithm

The TD(\(\lambda\)) Algorithm

The Q-learning Algorithm
Concentration Inequalities

Let $X$ be a random variable and $\{X_n\}_{n \in \mathbb{N}}$ a sequence of r.v.

- $\{X_n\}$ converges to $X$ almost surely, $X_n \xrightarrow{\text{a.s.}} X$, if
  \[\mathbb{P}(\lim_{n \to \infty} X_n = X) = 1,\]

- $\{X_n\}$ converges to $X$ in probability, $X_n \xrightarrow{\mathbb{P}} X$, if for any $\epsilon > 0$,
  \[\lim_{n \to \infty} \mathbb{P}[|X_n - X| > \epsilon] = 0,\]

- $\{X_n\}$ converges to $X$ in law (or in distribution), $X_n \xrightarrow{D} X$, if for any bounded continuous function $f$
  \[\lim_{n \to \infty} \mathbb{E}[f(X_n)] = \mathbb{E}[f(X)].\]

Remark: $X_n \xrightarrow{\text{a.s.}} X \implies X_n \xrightarrow{\mathbb{P}} X \implies X_n \xrightarrow{D} X$. 
Concentration Inequalities

**Proposition (Markov Inequality)**

Let $X$ be a positive random variable. Then for any $a > 0$,

$$
P(X \geq a) \leq \frac{\mathbb{E}X}{a}.
$$

**Proof.**

$$
P(X \geq a) = \mathbb{E}[\mathbb{I}\{X \geq a\}] = \mathbb{E}[\mathbb{I}\{X/a \geq 1\}] \leq \mathbb{E}[X/a]
$$
Concentration Inequalities

**Proposition (Hoeffding Inequality)**

Let $X$ be a *centered* random variable bounded in $[a, b]$. Then for any $s \in \mathbb{R}$,

$$
\mathbb{E}[e^{sX}] \leq e^{s^2(b-a)^2/8}.
$$
Concentration Inequalities

Proof.
From *convexity* of the exponential function, for any $a \leq x \leq b$, 

$$e^{sx} \leq \frac{x - a}{b - a} e^{sb} + \frac{b - x}{b - a} e^{sa}.$$ 

Let $p = -a/(b - a)$ then (recall that $\mathbb{E}[X] = 0$) 

$$\mathbb{E}[e^{sx}] \leq \frac{b}{b - a} e^{sa} - \frac{a}{b - a} e^{sb}$$

$$= (1 - p + pe^{s(b-a)})e^{-ps(b-a)} = e^{\phi(u)}$$

with $u = s(b-a)$ and $\phi(u) = -pu + \log(1 + pe^u)$ whose derivative is

$$\phi'(u) = -p + \frac{p}{p + (1 - p)e^{-u}},$$

and $\phi(0) = \phi'(0) = 0$ and $\phi''(u) = \frac{p(1-p)e^{-u}}{(p + (1 - p)e^{-u})^2} \leq 1/4$.

Thus from *Taylor's theorem*, the exists a $\theta \in [0, u]$ such that 

$$\phi(\theta) = \phi(0) + \theta \phi'(0) + \frac{u^2}{2} \phi''(\theta) \leq \frac{u^2}{8} = \frac{s^2(b-a)^2}{8}.$$
Concentration Inequalities

Proposition (Chernoff-Hoeffding Inequality)

Let $X_i \in [a_i, b_i]$ be $n$ independent r.v. with mean $\mu_i = \mathbb{E}X_i$. Then

$$\mathbb{P}\left[\left|\sum_{i=1}^{n}(X_i - \mu_i)\right| \geq \epsilon\right] \leq 2 \exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right).$$
Concentration Inequalities

Proof.

\[ P\left(\sum_{i=1}^{n} X_i - \mu_i \geq \epsilon\right) = P(e^s \sum_{i=1}^{n} X_i - \mu_i \geq e^{s\epsilon}) \]

\[ \leq e^{-s\epsilon} E[e^{s \sum_{i=1}^{n} X_i - \mu_i}], \quad \text{Markov inequality} \]

\[ = e^{-s\epsilon} \prod_{i=1}^{n} E[e^{s(X_i - \mu_i)}], \quad \text{independent random variables} \]

\[ \leq e^{-s\epsilon} \prod_{i=1}^{n} e^{s^2(b_i - a_i)^2/8}, \quad \text{Hoeffding inequality} \]

\[ = e^{-s\epsilon + s^2 \sum_{i=1}^{n} (b_i - a_i)^2 / 8} \]

If we choose \( s = 4\epsilon / \sum_{i=1}^{n} (b_i - a_i)^2 \), the result follows.

Similar arguments hold for \( P\left(\sum_{i=1}^{n} X_i - \mu_i \leq -\epsilon\right) \).
Monte-Carlo Approximation of a Mean

Definition

Let $X$ be a random variable with mean $\mu = \mathbb{E}[X]$ and variance $\sigma^2 = \mathbb{V}[X]$ and $x_n \sim X$ be $n$ i.i.d. realizations of $X$. The **Monte-Carlo approximation** of the mean (i.e., the empirical mean) built on $n$ i.i.d. realizations is defined as

$$\mu_n = \frac{1}{n} \sum_{i=1}^{n} x_i.$$
Monte-Carlo Approximation of a Mean

- **Unbiased estimator:** Then $\mathbb{E}[\mu_n] = \mu$ (and $\mathbb{V}[\mu_n] = \frac{\mathbb{V}[X]}{n}$)

- **Weak law of large numbers:** $\mu_n \xrightarrow{p} \mu$.

- **Strong law of large numbers:** $\mu_n \xrightarrow{a.s.} \mu$.

- **Central limit theorem (CLT):** $\sqrt{n}(\mu_n - \mu) \xrightarrow{D} \mathcal{N}(0, \mathbb{V}[X])$.

- **Finite sample guarantee:**

  $$\mathbb{P} \left[ \left| \frac{1}{n} \sum_{t=1}^{n} X_t - \mathbb{E}[X_1] \right| > \epsilon \right] \leq 2 \exp \left( - \frac{2n\epsilon^2}{(b - a)^2} \right)$$

  - **deviation**
  - **accuracy**
  - **confidence**
Monte-Carlo Approximation of a Mean

- **Unbiased estimator**: Then $\mathbb{E}[\mu_n] = \mu$ (and $\mathbb{V}[\mu_n] = \frac{\mathbb{V}[X]}{n}$)

- **Weak law of large numbers**: $\mu_n \xrightarrow{P} \mu$.

- **Strong law of large numbers**: $\mu_n \xrightarrow{a.s.} \mu$.

- **Central limit theorem (CLT)**: $\sqrt{n}(\mu_n - \mu) \xrightarrow{D} \mathcal{N}(0, \mathbb{V}[X])$.

- **Finite sample guarantee**:

  $$
  \mathbb{P}
  \left[
  \left| \frac{1}{n} \sum_{t=1}^{n} X_t - \mathbb{E}[X_1] \right| > (b - a) \sqrt{\frac{\log 2/\delta}{2n}} \right] \leq \delta
  $$
Monte-Carlo Approximation of a Mean

- **Unbiased estimator**: Then $E[\mu_n] = \mu$ (and $\nabla[\mu_n] = \frac{\nabla[X]}{n}$)

- **Weak law of large numbers**: $\mu_n \overset{P}{\rightarrow} \mu$.

- **Strong law of large numbers**: $\mu_n \overset{a.s.}{\rightarrow} \mu$.

- **Central limit theorem (CLT)**: $\sqrt{n}(\mu_n - \mu) \overset{D}{\rightarrow} \mathcal{N}(0, \nabla[X])$.

- **Finite sample guarantee**:

$$
P\left[ \left| \frac{1}{n} \sum_{t=1}^{n} X_t - E[X_1] \right| > \epsilon \right] \leq \delta$$

if $n \geq \frac{(b-a)^2 \log 2/\delta}{2\epsilon^2}$.
Exercise

Simulate $n$ Bernoulli of probability $p$ and verify the correctness and the accuracy of the C-H bounds.
Stochastic Approximation of a Mean

**Definition**

Let $X$ a random variable *bounded in* $[0, 1]$ with mean $\mu = \mathbb{E}[X]$ and $x_n \sim X$ be $n$ i.i.d. realizations of $X$. The *stochastic approximation* of the mean is,

$$\mu_n = (1 - \eta_n)\mu_{n-1} + \eta_n x_n$$

with $\mu_1 = x_1$ and where $(\eta_n)$ is a sequence of *learning steps*.

**Remark**: When $\eta_n = \frac{1}{n}$ this is the *recursive* definition of empirical mean.
Stochastic Approximation of a Mean

Proposition (Borel-Cantelli)

Let \((E_n)_{n \geq 1}\) be a sequence of events such that \(\sum_{n \geq 1} P(E_n) < \infty\), then the probability of the intersection of an infinite subset is 0. More formally,

\[
P\left( \limsup_{n \to \infty} E_n \right) = P\left( \bigcap_{n=1}^{\infty} \bigcup_{k=n}^{\infty} E_k \right) = 0.
\]
Stochastic Approximation of a Mean

Proposition

If for any $n$, $\eta_n \geq 0$ and are such that

$$\sum_{n \geq 0} \eta_n = \infty; \quad \sum_{n \geq 0} \eta_n^2 < \infty,$$

then

$$\mu_n \xrightarrow{a.s.} \mu,$$

and we say that $\mu_n$ is a *consistent* estimator.
Stochastic Approximation of a Mean

Proof. We focus on the case \( \eta_n = n^{-\alpha} \).

In order to satisfy the two conditions we need \( 1/2 < \alpha \leq 1 \). In fact, for instance

\[
\alpha = 2 \Rightarrow \sum_{n \geq 0} \frac{1}{n^2} = \frac{\pi^2}{6} < \infty \quad \text{(see the Basel problem)}
\]

\[
\alpha = 1/2 \Rightarrow \sum_{n \geq 0} \left(\frac{1}{\sqrt{n}}\right)^2 = \sum_{n \geq 0} \frac{1}{n} = \infty \quad \text{(harmonic series)}.
\]
Stochastic Approximation of a Mean

Proof (cont’d).

Case $\alpha = 1$

Let $(\epsilon_k)_k$ a sequence such that $\epsilon_k \to 0$, almost sure convergence corresponds to

$$
\mathbb{P}\left( \lim_{n \to \infty} \mu_n = \mu \right) = \mathbb{P}\left( \forall k, \exists n_k, \forall n \geq n_k, |\mu_n - \mu| \leq \epsilon_k \right) = 1.
$$

From Chernoff-Hoeffding inequality for any fixed $n$

$$
\mathbb{P}\left( |\mu_n - \mu| \geq \epsilon \right) \leq 2e^{-2n\epsilon^2}. \tag{1}
$$

Let $\{E_n\}$ be a sequence of events $E_n = \{|\mu_n - \mu| \geq \epsilon\}$. From C-H

$$
\sum_{n \geq 1} \mathbb{P}(E_n) < \infty,
$$

and from Borel-Cantelli lemma we obtain that with probability 1 there exist only a finite number of $n$ values such that $|\mu_n - \mu| \geq \epsilon$. 

Proof (cont’d).

Case $\alpha = 1$

Then for any $\epsilon_k$ there exist only a finite number of instants were $|\mu_n - \mu| \geq \epsilon_k$, which corresponds to have $\exists n_k$ such that

$$\mathbb{P}(\forall n \geq n_k, |\mu_n - \mu| \leq \epsilon_k) = 1$$

Repeating for all $\epsilon_k$ in the sequence leads to the statement.

Remark: when $\alpha = 1$, $\mu_n$ is the Monte-Carlo estimate and this corresponds to the strong law of large numbers. A more precise and accurate proof is here: http://terrytao.wordpress.com/2008/06/18/the-strong-law-of-large-numbers/
Stochastic Approximation of a Mean

*Proof (cont’d).*

**Case 1/2 < \( \alpha \) < 1.** The stochastic approximation \( \mu_n \) is

\[
\begin{align*}
\mu_1 &= x_1 \\
\mu_2 &= (1 - \eta_2)\mu_1 + \eta_2 x_2 = (1 - \eta_2)x_1 + \eta_2 x_2 \\
\mu_3 &= (1 - \eta_3)\mu_2 + \eta_3 x_3 = (1 - \eta_2)(1 - \eta_3)x_1 + \eta_2(1 - \eta_3)x_2 + \eta_3 x_3 \\
&\vdots \\
\mu_n &= \sum_{i=1}^{n} \lambda_i x_i,
\end{align*}
\]

with \( \lambda_i = \eta_i \prod_{j=i+1}^{n} (1 - \eta_j) \) such that \( \sum_{i=1}^{n} \lambda_i = 1.\)

By C-H inequality

\[
\mathbb{P}(\left| \sum_{i=1}^{n} \lambda_i x_i - \sum_{i=1}^{n} \lambda_i \mathbb{E}[x_i] \right| \geq \epsilon) = \mathbb{P}(\left| \mu_n - \mu \right| \geq \epsilon) \leq e^{-\frac{2\epsilon^2}{\sum_{i=1}^{n} \lambda_i^2}}.
\]
Stochastic Approximation of a Mean

Proof (cont’d).

Case $1/2 < \alpha < 1$.

From the definition of $\lambda_i$

$$
\log \lambda_i = \log \eta_i + \sum_{j=i+1}^{n} \log(1 - \eta_j) \leq \log \eta_i - \sum_{j=i+1}^{n} \eta_j
$$

since $\log(1 - x) < -x$. Thus $\lambda_i \leq \eta_i e^{-\sum_{j=i+1}^{n} \eta_j}$ and for any $1 \leq m \leq n$,

$$
\sum_{i=1}^{n} \lambda_i^2 \leq \sum_{i=1}^{n} \eta_i^2 e^{-2 \sum_{j=i+1}^{n} \eta_j}
$$

(a) \hspace{1cm} \leq \sum_{i=1}^{m} e^{-2 \sum_{j=i+1}^{n} \eta_j} + \sum_{i=m+1}^{n} \eta_i^2

(b) \hspace{1cm} \leq me^{-2(n-m)\eta_n} + (n - m)\eta_m^2

(c) \hspace{1cm} = me^{-2(n-m)n^{-\alpha}} + (n - m)m^{-2\alpha}.
Stochastic Approximation of a Mean

Proof (cont’d).

Case $1/2 < \alpha < 1$.

Let $m = n^\beta$ with $\beta = (1 + \alpha/2)/2$ (i.e. $1 - 2\alpha\beta = 1/2 - \alpha$):

$$\sum_{i=1}^{n} \lambda_i^2 \leq ne^{-2(1-n^{-1/4})n^{1-\alpha}} + n^{1/2-\alpha} \leq 2n^{1/2-\alpha}$$

for $n$ big enough, which leads to

$$\mathbb{P}(|\mu_n - \mu| \geq \epsilon) \leq e^{-\frac{\epsilon^2}{n^{1/2-\alpha}}}.$$

From this point we follow the same steps as for $\alpha = 1$ (application of the Borel-Cantelli lemma) and obtain the convergence result for $\mu_n$. 

Stochastic Approximation of a Fixed Point

**Definition**

Let $T : \mathbb{R}^N \to \mathbb{R}^N$ be a contraction in some norm $\| \cdot \|$ with fixed point $V$. For any function $W$ and state $x$, a noisy observation $\hat{T}W(x) = TW(x) + b(x)$ is available. For any $x \in X = \{1, \ldots, N\}$, we defined the stochastic approximation

$$V_{n+1}(x) = (1 - \eta_n(x))V_n(x) + \eta_n(x)(\hat{T}V_n(x))$$

$$= (1 - \eta_n(x))V_n(x) + \eta_n(x)(TW_n(x) + b_n),$$

where $\eta_n$ is a sequence of learning steps.
**Proposition**

Let $\mathcal{F}_n = \{V_0, \ldots, V_n, b_0, \ldots, b_{n-1}, \eta_0, \ldots, \eta_n\}$ the filtration of the algorithm and assume that

$$
\mathbb{E}[b_n(x)|\mathcal{F}_n] = 0 \quad \text{and} \quad \mathbb{E}[b_n^2(x)|\mathcal{F}_n] \leq c(1 + \|V_n\|^2)
$$

for a constant $c$.

If the learning rates $\eta_n(x)$ are positive and satisfy the stochastic approximation conditions

$$
\sum_{n \geq 0} \eta_n = \infty, \quad \sum_{n \geq 0} \eta_n^2 < \infty,
$$

then for any $x \in X$

$$
V_n(x) \xrightarrow{a.s.} V(x).
$$
Stochastic Approximation of a Zero

Robbins-Monro (1951) algorithm. Given a noisy function $f$, find $x^*$ such that $f(x^*) = 0$.

In each $x_n$, observe $y_n = f(x_n) + b_n$ (with $b_n$ a zero-mean independent noise) and compute

$$x_{n+1} = x_n - \eta_n y_n.$$

If $f$ is an increasing function, then under the same assumptions on the learning step

$$x_n \xrightarrow{\text{a.s.}} x^*$$
Stochastic Approximation of a Minimum

**Kiefer-Wolfowitz (1952) algorithm.** Given a function $f$ and noisy observations of its gradient, find $x^* = \arg\min f(x)$. In each $x_n$, observe $g_n = \nabla f(x_n) + b_n$ (with $b_n$ a zero-mean independent noise) and compute

$$x_{n+1} = x_n - \eta_n g_n.$$

If the Hessian $\nabla^2 f$ is *positive*, then under the same assumptions on the learning step

$$x_n \xrightarrow{a.s.} x^*$$

*Remark:* this is often referred to as the stochastic gradient algorithm.
Outline

Mathematical Tools

The Monte-Carlo Algorithm

The TD(1) Algorithm

The TD(0) Algorithm

The TD(\(\lambda\)) Algorithm

The Q-learning Algorithm
Policy Evaluation

We consider the problem of evaluating the performance of a policy $\pi$ in the *undiscounted infinite horizon* setting. For any (proper) policy $\pi$ the value function is

$$V^\pi(x) = \mathbb{E}\left[ \sum_{t=0}^{T-1} r^\pi(x_t) \mid x_0 = x; \pi \right],$$

where $r^\pi(x_t) = r(x_t, \pi(x_t))$ and $T$ is the *random* time when the *terminal state* is achieved.
Question

How can we estimate the value function if an episodic interaction with the environment is possible?

⇒ Monte-Carlo approximation of a mean!
The Monte-Carlo Algorithm

<table>
<thead>
<tr>
<th>Algorithm Definition (Monte-Carlo)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Let ((x_0^i = x, x_1^i, \ldots, x_{T_i}^i = 0)<em>{i \leq n}) be a set of (n) independent trajectories starting from (x) and terminating after (T_i) steps. For any (t &lt; T_i), we denote by (\hat{R}^i(x_t^i) = [r^\pi(x_t^i) + r^\pi(x</em>{t+1}^i) + \cdots + r^\pi(x_{T_i-1}^i)]) the return of the (i)-th trajectory at state (x_t^i). Then the Monte-Carlo estimator of (V^\pi(x)) is (V_n(x) = \frac{1}{n} \sum_{i=1}^{n} [r^\pi(x_0^i) + r^\pi(x_1^i) + \cdots + r^\pi(x_{T_i-1}^i)] = \frac{1}{n} \sum_{i=1}^{n} \hat{R}^i(x)).</td>
</tr>
</tbody>
</table>
All the returns are unbiased estimators of $V^\pi(x)$ since

$$E[\hat{R}^i(x)] = E\left[\sum_{t} r^\pi(x_t^i) + r^\pi(x_{t+1}^i) + \cdots + r^\pi(x_{T^i-1}^i)\right] = V^\pi(x)$$

then

$$V_n(x) \xrightarrow{a.s.} V^\pi(x).$$
First-visit and Every-Visit Monte-Carlo

Remark: any trajectory \((x_0, x_1, x_2, \ldots, x_T)\) contains also the sub-trajectory \((x_t, x_{t+1}, \ldots, x_T)\) whose return \(\hat{R}(x_t) = r^\pi(x_t) + \cdots + r^\pi(x_{T-1})\) could be used to build an estimator of \(V^\pi(x_t)\).

- **First-visit MC.** For each state \(x\) we only consider the sub-trajectory when \(x\) is first achieved. *Unbiased estimator*, *only one sample per trajectory*.

- **Every-visit MC.** Given a trajectory \((x_0 = x, x_1, x_2, \ldots, x_T)\), we list all the \(m\) sub-trajectories starting from \(x\) up to \(x_T\) and we average them all to obtain an estimate. *More than one sample per trajectory, biased estimator*.
Question

More samples or no bias?

⇒ Sometimes a biased estimator is preferable if consistent!
**Example**: 2-state Markov Chain

The reward is 1 while in state 1 (while is 0 in the terminal state). All trajectories are \((x_0 = 1, x_1 = 1, \ldots, x_T = 0)\). By Bellman equations

\[ V(1) = 1 + (1 - p)V(1) + 0 \cdot p = \frac{1}{p}, \]

since \(V(0) = 0\).
First-visit vs Every-Visit Monte-Carlo

We measure the mean squared error (MSE) of $\hat{V}$ w.r.t. $V$

$$
\mathbb{E}[(\hat{V} - V)^2] = (\mathbb{E}[\hat{V}] - V)^2 + \mathbb{E}[(\hat{V} - \mathbb{E}[\hat{V}])^2]
$$

Bias$^2$ + Variance
First-visit vs Every-Visit Monte-Carlo

**First-visit Monte-Carlo.** All the trajectories start from state 1, then the return over one single trajectory is exactly $T$, i.e., $\hat{V} = T$. The time-to-end $T$ is a geometric r.v. with expectation

$$E[\hat{V}] = E[T] = \frac{1}{p} = V^\pi(1) \Rightarrow \text{unbiased estimator.}$$

Thus the MSE of $\hat{V}$ coincides with the variance of $T$, which is

$$E\left[\left(T - \frac{1}{p}\right)^2\right] = \frac{1}{p^2} - \frac{1}{p}.$$
First-visit vs Every-Visit Monte-Carlo

**Every-visit Monte-Carlo.** Given one trajectory, we can construct $T - 1$ sub-trajectories (number of times state 1 is visited), where the $t$-th trajectory has a return $T - t$.

$$
\hat{V} = \frac{1}{T} \sum_{t=0}^{T-1} (T - t) = \frac{1}{T} \sum_{t'=1}^{T} t' = \frac{T + 1}{2}.
$$

The corresponding expectation is

$$
\mathbb{E} \left[ \frac{T + 1}{2} \right] = \frac{1 + p}{2p} \neq V^\pi(1) \Rightarrow \text{biased estimator}.
$$
First-visit vs Every-Visit Monte-Carlo

Let’s consider \( n \) independent trajectories, each of length \( T_i \). Total number of samples \( \sum_{i=1}^{n} T_i \) and the estimator \( \hat{V}_n \) is

\[
\hat{V}_n = \frac{\sum_{i=1}^{n} \sum_{t=0}^{T_i-1} (T_i - t)}{\sum_{i=1}^{n} T_i} = \frac{\sum_{i=1}^{n} T_i(T_i + 1)}{2 \sum_{i=1}^{n} T_i}
\]

\[
= \frac{1}{n} \frac{\sum_{i=1}^{n} T_i(T_i + 1)}{2/n \sum_{i=1}^{n} T_i}
\]

\[
a.s. \quad \frac{\mathbb{E}[T^2] + \mathbb{E}[T]}{2 \mathbb{E}[T]} = \frac{1}{p} = V^\pi(1) \Rightarrow \text{consistent estimator}.
\]

The MSE of the estimator

\[
\mathbb{E} \left[ \left( \frac{T + 1}{2} - \frac{1}{p} \right)^2 \right] = \frac{1}{2p^2} - \frac{3}{4p} + \frac{1}{4} \leq \frac{1}{p^2} - \frac{1}{p}.
\]
First-visit vs Every-Visit Monte-Carlo

In general

- **Every-visit MC**: biased but consistent estimator.
- **First-visit MC**: unbiased estimator with potentially bigger MSE.

Remark: when the state space is large the probability of visiting multiple times the same state is low, then the performance of the two methods tends to be the same.
Outline

Mathematical Tools

The Monte-Carlo Algorithm

The TD(1) Algorithm

The TD(0) Algorithm

The TD(λ) Algorithm

The Q-learning Algorithm
Policy Evaluation

We consider the problem of evaluating the performance of a policy \( \pi \) in the *undiscounted infinite horizon* setting. For any (proper) policy \( \pi \) the value function is

\[
V^\pi(x) = \mathbb{E} \left[ \sum_{t=0}^{T-1} r^\pi(x_t) \mid x_0 = x; \pi \right],
\]

where \( r^\pi(x_t) = r(x_t, \pi(x_t)) \) and \( T \) is the *random* time when the *terminal state* is achieved.
Question

MC requires all the trajectories to be available at once, can we update the estimator online?

⇒ $TD(1)!$
The TD(1) Algorithm

Algorithm Definition (TD(1))

Let \((x^n_0 = x, x^n_1, \ldots, x^n_T)\) be the \(n\)-th trajectory and \(\hat{R}^n\) be the corresponding return. For all \(x_t\) with \(t \leq T - 1\) observed along the trajectory, we update the value function estimate as

\[
V_n(x^n_t) = (1 - \eta_n(x^n_t))V_{n-1}(x^n_t) + \eta_n(x^n_t)\hat{R}^n(x^n_t).
\]
The TD(1) Algorithm

Each sample is an unbiased estimator of the value function

$$\mathbb{E} [ r^\pi(x_t) + r^\pi(x_{t+1}) + \cdots + r^\pi(x_{T-1}) | x_t ] = V^\pi(x_t),$$

then the convergence result of stochastic approximation of a mean applies and if all the states are visited in an infinite number of trajectories and for all $x \in X$

$$\sum_n \eta_n(x) = \infty, \quad \sum_n \eta_n(x)^2 < \infty,$$

then

$$V_n(x) \xrightarrow{a.s.} V^\pi(x)$$
Outline

Mathematical Tools

The Monte-Carlo Algorithm

The TD(1) Algorithm

The TD(0) Algorithm

The TD(λ) Algorithm

The Q-learning Algorithm
We consider the problem of evaluating the performance of a policy $\pi$ in the *undiscounted infinite horizon* setting. For any (proper) policy $\pi$ the value function is

$$V^\pi(x) = r(x, \pi(x)) + \sum_{y \in X} p(y|x, \pi(x)) V^\pi(x) = \mathcal{T}^\pi V^\pi(x).$$

⇒ use *stochastic approximation for fixed point*. 
The TD(0) Algorithm

- **Noisy** observation of the operator $\mathcal{T}^\pi$:
  \[
  \hat{T}^\pi V(x_t) = r^\pi(x_t) + V(x_{t+1}), \text{ with } x_t = x,
  \]

- **Unbiased** estimator of $\mathcal{T}^\pi V(x)$ since
  \[
  \mathbb{E}[\hat{T}^\pi V(x_t)|x_t = x] = \mathbb{E}[r^\pi(x_t) + V(x_{t+1})|x_t = x] \\
  = r(x, \pi(x)) + \sum_y p(y|x, \pi(x)) V(y) = \mathcal{T}^\pi V(x).
  \]

- **Bounded** noise since
  \[
  |\hat{T}^\pi V(x) - \mathcal{T}^\pi V(x)| \leq \|V\|_\infty.
  \]
The TD(0) Algorithm

Algorithm Definition (TD(0))

Let \((x_0^n = x, x_1^n, \ldots, x_{T_n}^n)\) be the \(n\)-th trajectory, and \(\{\hat{T}^\pi V_{n-1}(x_t^n)\}_t\) the noisy observation of the operator \(T^\pi\). For all \(x_t^n\) with \(t \leq T^n - 1\), we update the value function estimate as

\[
V_n(x_t^n) = (1 - \eta_n(x_t^n))V_{n-1}(x_t^n) + \eta_n(x_t^n)\hat{T}^\pi V_{n-1}(x_t^n)
= (1 - \eta_n(x_t^n))V_{n-1}(x_t^n) + \eta_n(x_t^n)(r^\pi(x_t) + V_{n-1}(x_{t+1})).
\]
The TD(0) Algorithm

if all the states are visited in an infinite number of trajectories and for all \( x \in X \)

\[
\sum_n \eta_n(x) = \infty, \quad \sum_n \eta_n(x)^2 < \infty,
\]

then

\[
V_n(x) \xrightarrow{\text{a.s.}} V^\pi(x)
\]
The TD(0) Algorithm

Definition

At iteration $n$, given the estimator $V_{n-1}$ and a transition from state $x_t$ to state $x_{t+1}$ we define the temporal difference

$$d_t = (r^\pi(x_t) + V_{n-1}(x_{t+1})) - V_{n-1}(x_t).$$

Remark: Recalling the definition of Bellman equation for state value function, the temporal difference $d^n_t$ provides a measure of coherence of the estimator $V_{n-1}$ w.r.t. the transition $x_t \rightarrow x_{t+1}$. 
The TD(0) Algorithm

Algorithm Definition (TD(0))

Let \((x^n_0 = x, x^n_1, \ldots, x^n_{T^n})\) be the \(n\)-th trajectory, and \(\{d^n_t\}_t\) the temporal differences. For all \(x^n_t\) with \(t \leq T^n - 1\), we update the value function estimate as

\[
V_n(x^n_t) = V_{n-1}(x^n_t) + \eta_n(x^n_t)d^n_t.
\]
Outline

Mathematical Tools

The Monte-Carlo Algorithm

The TD(1) Algorithm

The TD(0) Algorithm

The TD(λ) Algorithm

The Q-learning Algorithm
Comparison between TD(1) and TD(0)

- **TD(1)**
  \[
  V_n(x_t) = V_{n-1}(x_t) + \eta_n(x_t)[d^n_t + d^n_{t+1} + \cdots + d^n_{T-1}].
  \]

- **TD(0)**
  \[
  V_n(x_t^n) = V_{n-1}(x_t^n) + \eta_n(x_t^n)d^n_t.
  \]
Question

Is it possible to take the best of both?

⇒ $TD(\lambda)!$
The $\mathcal{T}^\pi_\lambda$ Bellman operator

**Definition**

Given $\lambda < 1$, then the Bellman operator $\mathcal{T}^\pi_\lambda$ is

$$
\mathcal{T}^\pi_\lambda = (1 - \lambda) \sum_{m \geq 0} \lambda^m (\mathcal{T}^\pi)^{m+1}.
$$

**Remark:** convex combination of the $m$-step Bellman operators $$(\mathcal{T}^\pi)^m$$ weighted by a sequences of coefficients defined as a function of a $\lambda$. 
The TD(λ) Algorithm

Proposition

If π is a proper policy and \( T^\pi \) is a \( \beta \)-contraction in \( L_{\mu,\infty} \)-norm, then \( T^\pi_\lambda \) is a contraction of factor

\[
\frac{(1 - \lambda)\beta}{1 - \beta\lambda} \in [0, \beta].
\]
The TD(\(\lambda\)) Algorithm

Proof. Let \(P^\pi\) be the transition matrix of the Markov chain then

\[
\mathcal{T}_\lambda^\pi V = (1-\lambda) \left[ \sum_{m \geq 0} \lambda^m \sum_{i=0}^m (P^\pi)^i \right] r^\pi + (1-\lambda) \sum_{m \geq 0} \lambda^m (P^\pi)^{m+1} V
\]

\[
= \left[ \sum_{m \geq 0} \lambda^m (P^\pi)^m \right] r^\pi + (1-\lambda) \sum_{m \geq 0} \lambda^m (P^\pi)^{m+1} V
\]

\[
= (I - \lambda P^\pi)^{-1} r^\pi + (1-\lambda) \sum_{m \geq 0} \lambda^m (P^\pi)^{m+1} V.
\]

Since \(\mathcal{T}^\pi\) is a \(\beta\)-contraction then \(\|(P^\pi)^m V\|_\mu \leq \beta^m \|V\|_\mu\). Thus

\[
\|(1-\lambda) \sum_{m \geq 0} \lambda^m (P^\pi)^{m+1} V\|_\mu \leq (1-\lambda) \sum_{m \geq 0} \lambda^m \|(P^\pi)^{m+1} V\|_\mu \leq \frac{(1-\lambda)\beta}{1-\beta\lambda} \|V\|_\mu,
\]

which implies that \(\mathcal{T}_\lambda^\pi\) is a contraction in \(L_{\mu,\infty}\) as well.
The TD(\(\lambda\)) Algorithm

Algorithm Definition (Sutton, 1988)

Let \((x^n_0 = x, x^n_1, \ldots, x^n_T)\) be the \(n\)-th trajectory, and \(\{d^n_t\}_t\) the temporal differences. For all \(x_t\) with \(t \leq T - 1\), we update the value function estimate as

\[
V_n(x^n_t) = V_{n-1}(x^n_t) + \eta_n(x^n_t) \sum_{s=t}^{T_n-1} \lambda^{s-t} d^n_s.
\]
The TD(\(\lambda\)) Algorithm

We need to show that the temporal difference samples are *unbiased* estimators. For any \(s \geq t\)

\[
\mathbb{E}[d_s | x_t = x] = \mathbb{E}[r^\pi(x_s) + V_{n-1}(x_{s+1}) - V_{n-1}(x_s) | x_t = x]
\]

\[
= \mathbb{E} \left[ \sum_{i=t}^{s} r^\pi(x_i) + V_{n-1}(x_{s+1}) | x_t = x \right] - \mathbb{E} \left[ \sum_{i=k}^{s-1} r^\pi(x_i) + V_{n-1}(x_s) | x_t = x \right]
\]

\[
= (T^\pi)^{s-t+1} V_{n-1}(x) - (T^\pi)^{s-t} V_{n-1}(x).
\]
The TD($\lambda$) Algorithm

\[
\mathbb{E}\left[\sum_{s=t}^{T-1} \lambda^{s-t} d_s | x_t = x\right] = \sum_{s=t}^{T-1} \lambda^{s-t} \left[ (T^\pi)_n^{s-t+1} V_{n-1}(x) - (T^\pi)_n^{s-t} V_{n-1}(x) \right]
\]

\[
= \sum_{m \geq 0} \lambda^m \left[ (T^\pi)_n^{m+1} V_{n-1}(x) - (T^\pi)_n^m V_{n-1}(x) \right]
\]

\[
= \sum_{m \geq 0} \lambda^m (T^\pi)_n^{m+1} V_{n-1}(x) - \left[ V_{n-1}(x) + \sum_{m > 0} \lambda^m (T^\pi)_n^m V_{n-1}(x) \right]
\]

\[
= \sum_{m \geq 0} \lambda^m (T^\pi)_n^{m+1} V_{n-1}(x) - \left[ V_{n-1}(x) + \lambda \sum_{m > 0} \lambda^{m-1} (T^\pi)_n^m V_{n-1}(x) \right]
\]

\[
= \sum_{m \geq 0} \lambda^m (T^\pi)_n^{m+1} V_{n-1}(x) - \left[ V_{n-1}(x) + \lambda \sum_{m \geq 0} \lambda^m (T^\pi)_n^{m+1} V_{n-1}(x) \right]
\]

\[
= (1 - \lambda) \sum_{m \geq 0} \lambda^m (T^\pi)_n^{m+1} V_{n-1}(x) - V_{n-1}(x) = \sum_{m \geq 0} \lambda^m (T^\pi)_n^{m+1} V_{n-1}(x) - V_{n-1}(x).
\]

Then

\[ V_n \xrightarrow{a.s.} V^\pi \]
Sensitivity to $\lambda$

*Linear chain example*

The MSE of $V_n$ w.r.t. $V^{\pi}$ after $n = 100$ trajectories:
Sensitivity to $\lambda$

- $\lambda < 1$: smaller variance w.r.t. $\lambda = 1$ (MC/TD(1)).
- $\lambda > 0$: faster propagation of rewards w.r.t. $\lambda = 0$. 
Question

*Is it possible to update the V estimate at each step?*

⇒ *Online implementation!*
Online Implementation of TD algorithm: Eligibility Traces

Remark: since the update occurs at each step, now we drop the dependency on $n$.

- **Eligibility** traces $z \in \mathbb{R}^N$
- For every transition $x_t \rightarrow x_{t+1}$
  1. Compute the temporal difference
     $$d_t = r^\pi(x_t) + V(x_{t+1}) - V(x_t)$$
  2. Update the eligibility traces
     $$z(x) = \begin{cases} 
     \lambda z(x) & \text{if } x \neq x_t \\
     1 + \lambda z(x) & \text{if } x = x_t \\
     0 & \text{if } x_t = 0 \text{ (reset the traces)}
     \end{cases}$$
  3. For all state $x \in X$
     $$V(x) \leftarrow V(x) + \eta_t(x)z(x)d_t.$$
TD(\(\lambda\)) in discounted reward MDPs

The Bellman operator \(T_\pi^\lambda\) is defined as

\[
T_\pi^\lambda V(x_0) = (1 - \lambda)\mathbb{E}\left[ \sum_{t \geq 0} \lambda^t \left( \sum_{i=0}^{t} \gamma^i r_\pi(x_i) + \gamma^{t+1} V(x_{t+1}) \right) \right]
\]

\[
= \mathbb{E}\left[ (1 - \lambda) \sum_{i \geq 0} \gamma^i r_\pi(x_i) \sum_{t \geq i} \lambda^t + \sum_{t \geq 0} \gamma^{t+1} V(x_{t+1})(\lambda^t - \lambda^{t+1}) \right]
\]

\[
= \mathbb{E}\left[ \sum_{i \geq 0} \lambda^i \left( \gamma^i r_\pi(x_i) + \gamma^{i+1} V(x_{i+1}) - \gamma^i V(x_i) \right) \right] + V_n(x_0)
\]

\[
= \mathbb{E}\left[ \sum_{i \geq 0} (\gamma \lambda)^i d_i \right] + V(x_0),
\]

with the temporal difference \(d_i = r_\pi(x_i) + \gamma V(x_{i+1}) - V(x_i)\).

The corresponding TD(\(\lambda\)) algorithm becomes

\[
V_{n+1}(x_t) = V_n(x_t) + \eta_n(x_t) \sum_{s \geq t} (\gamma \lambda)^{s-t} d_t.
\]
Outline

Mathematical Tools

The Monte-Carlo Algorithm

The TD(1) Algorithm

The TD(0) Algorithm

The TD(\lambda) Algorithm

The Q-learning Algorithm
Question

*How do we compute the optimal policy online?*

⇒ *Q-learning!*
Q-learning

Remark: if we use TD algorithms to compute $V_n \approx V^{\pi_k}$, then we could compute the greedy policy as

$$\pi_{k+1}(x) \in \arg \max_a \left[ r(x, a) + \sum_y p(y|x, a) V_n(y) \right].$$

Problem: the transition $p$ is unknown!!
Solution: use Q-functions and compute

$$\pi_{k+1}(x) \in \arg \max_a Q_n(x, a)$$
Q-learning

Algorithm Definition (Watkins, 1989)

We build a sequence \( \{Q_n\} \) in such a way that for every observed transition \((x, a, y, r)\)

\[
Q_{n+1}(x, a) = (1 - \eta_n(x, a))Q_n(x, a) + \eta_n(x, a)\left[ r + \max_{b \in A} Q_n(y, b) \right].
\]
Q-learning

Proposition

[Watkins et Dayan, 1992] Let assume that all the policies $\pi$ are proper and that all the state-action pairs are visited infinitely often. If

$$\sum_{n \geq 0} \eta_n(x, a) = \infty, \quad \sum_{n \geq 0} \eta_n^2(x, a) < \infty$$

then for any $x \in X$, $a \in A$,

$$Q_n(x, a) \xrightarrow{a.s.} Q^*(x, a).$$
The Q-learning Algorithm

Q-learning

Proof.
Optimal Bellman operator $\mathcal{T}$

$$
\mathcal{T}W(x, a) = r(x, a) + \sum_y p(y|x, a) \max_{b \in A} W(y, b),
$$

with unique fixed point $Q^*$. Since all the policies are proper $\mathcal{T}$ is a contraction in the $L_{\mu,\infty}$-norm.

Q-learning can be written as

$$
Q_{n+1}(x, a) = (1 - \eta_n(x, a))Q_n(x, a) + \eta_n[\mathcal{T}Q_n(x, a) + b_n(x, a)],
$$

where $b_n(x, a)$ is a zero-mean random variable such that

$$
\mathbb{E}[b_n^2(x, a)] \leq c(1 + \max_{y,b} Q_n^2(y, b))
$$

The statement follows from convergence of stochastic approximation of fixed point operators.
The Q-learning Algorithm

Reinforcement Learning

Alessandro Lazaric
alessandro.lazaric@inria.fr
sequel.lille.inria.fr