It is hard to predict, especially about the future.

Niels Bohr

You are what you pretend to be, so be careful what you pretend to be.

Kurt Vonnegut
Convergence rate of TD(0) with function approximation

Prashanth L A†

Joint work with Nathaniel Korda‡ and Rémi Munos*

†Indian Institute of Science  ‡MLRG - Oxford University  *Google DeepMind

March 27, 2015
Background
Markov Decision Processes (MDPs)

**MDP:** Set of States $\mathcal{X}$, Set of Actions $\mathcal{A}$, Rewards $r(x, a)$

**Transition probability:**

$$p(s, a, s') = Pr\left\{s_{t+1} = s' | s_t = s, a_t = a\right\}$$

\[\begin{align*}
  &s_t \quad a_t \quad s_{t+1} \quad r_{t+1} \\
  &\quad t \quad \quad \quad t+1
\end{align*}\]
The Controlled Markov Property

- **Controlled Markov Property:** \( \forall i_0, i_1, \ldots, s, s', b_0, b_1 \ldots, a, \)
  \[
P(s_{t+1} = s' \mid s_t = s, a_t = a, \ldots, s_0 = i_0, a_0 = b_0) = p(s, a, s')
  \]

**Figure:** The Controlled Markov Behaviour
Value function

\[ V^\pi(s) = E \left[ \sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \mid s_0 = s, \pi \right] \]

Value function \quad Reward \quad Policy

\( V^\pi \) is the fixed point of the Bellman Operator \( \mathcal{T}^\pi \):

\[
\mathcal{T}^\pi(V)(s) := r(s, \pi(s)) + \beta \sum_{s'} p(s, \pi(s), s') V(s')
\]
Value function

\[ V^\pi(s) = E\left[ \sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \mid s_0 = s, \pi \right] \]

\( V^\pi \) is the fixed point of the Bellman Operator \( T^\pi \):

\[ T^\pi(V)(s) := r(s, \pi(s)) + \beta \sum_{s'} p(s, \pi(s), s') V(s') \]
Value function

\[ V^\pi(s) = E \left[ \sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \mid s_0 = s, \pi \right] \]

\( V^\pi \) is the fixed point of the Bellman Operator \( T^\pi \):

\[ T^\pi (V)(s) := r(s, \pi(s)) + \beta \sum_{s'} p(s, \pi(s), s') V(s') \]
Value function

\[ V^\pi(s) = E \left[ \sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \mid s_0 = s, \pi \right] \]

\( V^\pi \) is the fixed point of the Bellman Operator \( T^\pi \):

\[ T^\pi(V)(s) := r(s, \pi(s)) + \beta \sum_{s'} p(s, \pi(s), s') V(s') \]
Value function

$$V^\pi(s) = E \left[ \sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \mid s_0 = s, \pi \right]$$

$V^\pi$ is the fixed point of the Bellman Operator $T^\pi$:

$$T^\pi(V)(s) := r(s, \pi(s)) + \beta \sum_{s'} p(s, \pi(s), s') V(s')$$
Policy evaluation using TD

Temporal difference learning

- **Problem:** estimate the value function for a given policy $\pi$
- **Solution:** Use TD(0)

$$V_{t+1}(s_t) = V_t(s_t) + \alpha_t \left( r_{t+1} + \gamma V_t(s_{t+1}) - V_t(s_t) \right).$$

**Why TD(0)?**

- Simulation based algorithms like Monte-Carlo (no model necessary!)
- Update a guess based on another guess (like DP)
- Guaranteed convergence to value function $V^\pi(s)$ under standard assumptions
Policy evaluation using TD

Temporal difference learning

- **Problem**: estimate the value function for a given policy $\pi$
- **Solution**: Use TD(0)

$$V_{t+1}(s_t) = V_t(s_t) + \alpha_t \left( r_{t+1} + \gamma V_t(s_{t+1}) - V_t(s_t) \right).$$

Why TD(0)?

- Simulation based algorithms like Monte-Carlo (no model necessary!)
- Update a guess based on another guess (like DP)
- Guaranteed convergence to value function $V^\pi(s)$ under standard assumptions
**TD with Function Approximation**

Linear Function Approximation.

\[ V^\pi(s) \approx \theta^T \phi(s) \]

Parameter \( \theta \in \mathbb{R}^d \)

Feature \( \phi(s) \in \mathbb{R}^d \)

Note: \( d \ll |S| \)

**TD Fixed Point**

\[ \Phi \theta^* = \Pi \mathcal{T}^\pi (\Phi \theta^*) \]

Feature Matrix

Orthogonal Projection

with rows \( \phi(s)^T, \forall s \in \mathcal{S} \)

to \( \mathcal{B} = \{ \Phi \theta \mid \theta \in \mathbb{R}^d \} \)
TD with Function Approximation

Linear Function Approximation.

$V^\pi(s) \approx \theta^T \phi(s)$

Parameter $\theta \in \mathbb{R}^d$

Feature $\phi(s) \in \mathbb{R}^d$

Note: $d << |S|$

TD Fixed Point

$\Phi \theta^* = \Pi \mathcal{T}^\pi(\Phi \theta^*)$

Feature Matrix with rows $\phi(s)^T$, $\forall s \in \mathcal{S}$

Orthogonal Projection to $\mathcal{B} = \{\Phi \theta \mid \theta \in \mathbb{R}^d\}$
**TD(0) with function approximation**

\[ \theta_{n+1} = \theta_n + \gamma_n (r(s_n, \pi(s_n)) + \beta \theta_n^T \phi(s_{n+1}) - \theta_n^T \phi(s_n)) \phi(s_n) \]

**Step-sizes**

**Fixed-point iteration**

J. N. Tsitsiklis and B.V. Roy. (1997) show that \( \theta_n \to \theta^* \) a.s., where

\[ A\theta^* = b, \text{ where } A = \Phi^T \Psi (I - \beta P) \Phi \text{ and } b = \Phi^T \Psi r. \]

---

Assumptions

**Ergodicity**  Markov chain induced by the policy $\pi$ is irreducible and aperiodic. Moreover, there exists a stationary distribution $\Psi(= \Psi_\pi)$ for this Markov chain.

**Linear independence**  Feature matrix $\Phi$ has full column rank $\Rightarrow$ 
\[ \lambda_{\min}(\Phi^\top \Psi \Phi) \geq \mu > 0 \]

**Bounded rewards**  $|r(s, \pi(s))| \leq 1$, for all $s \in S$.

**Bounded features**  $\|\phi(s)\|_2 \leq 1$, for all $s \in S$. 
Assumptions (contd)

Step sizes satisfy $\sum_n \gamma_n = \infty$, and $\sum_n \gamma_n^2 < \infty$.

**Bounded mixing time**  $\exists$ a non-negative function $B(\cdot)$ such that: $\forall s_0 \in S$ and $m \geq 0$,

$$\sum_{\tau=0}^{\infty} \| \mathbb{E}(\phi(s_\tau) \mid s_0) - \mathbb{E}_\Psi(\phi(s_\tau)) \| \leq B(s_0),$$

$$\sum_{\tau=0}^{\infty} \| \mathbb{E}[\phi(s_\tau)\phi(s_{\tau+m})^T \mid s_0] - \mathbb{E}_\Psi[\phi(s_\tau)\phi(s_{\tau+m})^T] \| \leq B(s_0),$$

where $B(\cdot)$ satisfies:

for any $q > 1$, there exists a $K_q < \infty$ such that $\mathbb{E}[B^q(s) \mid s_0] \leq K_q B^q(s_0)$. 
In the long run we are all dead.  

John Maynard Keynes

Question: What happens in a short run of TD(0) with function approximation?
Concentration Bounds: Non-averaged TD(0)
Non-averaged case: Bound in expectation

Step-size choice

\[ \gamma_n = \frac{c}{2(c + n)}, \quad \text{with} \quad (1 - \beta)^2 \mu c > 1/2 \]

Bound in expectation

\[ \mathbb{E}\|\theta_n - \theta^*\|_2 \leq \frac{K_1(n)}{\sqrt{n + c}}, \quad \text{where} \]

\[ K_1(n) = \frac{2\sqrt{c}\|\theta_0 - \theta^*\|_2}{(n + c)^2(1 - \beta)^2 \mu c - 1/2} + \frac{c(1 - \beta)(3 + 6H)B(s_0)}{\sqrt{2(1 - \beta)^2 \mu c - 1}} \]

\( H \) is an upper bound on \( \|\theta_n\|_2 \), for all \( n \).
Non-averaged case: Bound in expectation

Step-size choice

\[ \gamma_n = \frac{c}{2(c + n)}, \quad \text{with} \quad (1 - \beta)^2 \mu c > 1/2 \]

Bound in expectation

\[ \mathbb{E} \| \theta_n - \theta^* \|_2 \leq \frac{K_1(n)}{\sqrt{n + c}}, \quad \text{where} \]

\[ K_1(n) = \frac{2\sqrt{c} \| \theta_0 - \theta^* \|_2}{(n + c)^2(1 - \beta)^2 \mu c - 1/2} + \frac{c(1 - \beta)(3 + 6H)B(s_0)}{\sqrt{2}(1 - \beta)^2 \mu c - 1} \]

\[ H \text{ is an upper bound on } \| \theta_n \|_2, \text{ for all } n. \]
Non-averaged case: High probability bound

Step-size choice

\[ \gamma_n = \frac{c}{2(c+n)}, \quad \text{with} \quad (\mu(1-\beta)/2 + 3B(s_0))c > 1 \]

High-probability bound

\[
\mathbb{P}\left( \|\theta_n - \theta^*\|_2 \leq \frac{K_2(n)}{\sqrt{n+c}} \right) \geq 1 - \delta, \quad \text{where}
\]

\[
K_2(n) := \frac{(1-\beta)c\sqrt{\ln(1/\delta)(1+9B(s_0)^2)}}{(\mu(1-\beta)/2 + 3B(s_0)^2)c - 1} + K_1(n)
\]

\(K_1(n)\) and \(K_2(n)\) above are \(O(1)\)
Non-averaged case: High probability bound

Step-size choice

$$\gamma_n = \frac{c}{2(c + n)}, \quad \text{with} \quad \left(\mu(1 - \beta)/2 + 3B(s_0)\right)c > 1$$

High-probability bound

$$\mathbb{P}\left(\|\theta_n - \theta^*\|_2 \leq \frac{K_2(n)}{\sqrt{n + c}}\right) \geq 1 - \delta,$$

where

$$K_2(n) := \frac{(1 - \beta)c \sqrt{\ln(1/\delta)(1 + 9B(s_0)^2)}}{(\mu(1 - \beta)/2 + 3B(s_0)^2)c - 1} + K_1(n)$$

$$K_1(n) \text{ and } K_2(n) \text{ above are } O(1)$$
Non-averaged case: High probability bound

Step-size choice

\[ \gamma_n = \frac{c}{2(c + n)}, \quad \text{with} \quad (\mu(1 - \beta)/2 + 3B(s_0)) c > 1 \]

High-probability bound

\[
\Pr \left( \| \theta_n - \theta^* \|_2 \leq \frac{K_2(n)}{\sqrt{n + c}} \right) \geq 1 - \delta, \quad \text{where}
\]

\[
K_2(n) := \frac{(1 - \beta)c\sqrt{\ln(1/\delta)(1 + 9B(s_0)^2)}}{\left(\mu(1 - \beta)/2 + 3B(s_0)^2\right)c - 1} + K_1(n)
\]

\[ K_1(n) \text{ and } K_2(n) \text{ above are } O(1) \]
Why are these bounds problematic?

*Obtaining optimal rate* \( O\left(\frac{1}{\sqrt{n}}\right) \) *with a step-size* \( \gamma_n = \frac{c}{(c + n)} \)

**In expectation:** Require \( c \) to be chosen such that \((1 - \beta)^2 \mu c \in (1/2, \infty)\)

**In high-probability:** \( c \) should satisfy \((\mu(1 - \beta)/2 + 3B(s_0)) c > 1.\)

Optimal rate requires knowledge of the mixing bound \( B(s_0) \)
Even for finite state space settings, \( B(s_0) \) is a constant, albeit one that depends on the transition dynamics!

**Solution**
Iterate averaging
Why are these bounds problematic?

Obtaining optimal rate $O\left(\frac{1}{\sqrt{n}}\right)$ with a step-size $\gamma_n = c/(c + n)$

In expectation: Require $c$ to be chosen such that $(1 - \beta)^2 \mu c \in (1/2, \infty)$

In high-probability: $c$ should satisfy $(\mu(1 - \beta)/2 + 3B(s_0)) c > 1$.

Optimal rate requires knowledge of the mixing bound $B(s_0)$
Even for finite state space settings, $B(s_0)$ is a constant, albeit one that depends on the transition dynamics!

Solution
Iterate averaging
Proof Outline

Let $z_n = \theta_n - \theta^*$. We first bound the deviation of this error from its mean:

$$
\mathbb{P}(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \geq \epsilon) \leq \exp \left( - \frac{\epsilon^2}{2 \sum_{i=1}^{n} L_i^2} \right), \quad \forall \epsilon > 0,
$$

and then bound the size of the mean itself:

$$
\mathbb{E}\|z_n\|_2 \leq \left[ \frac{2 \exp(\epsilon) \|z_0\|_2}{s_0^{\gamma_n}} \right] + \left( \sum_{k=1}^{n-1} (3 + 6H)^2 B(s_0)^2 \gamma_{k+1}^2 \exp(-2(1 - \beta)\mu(\Gamma_n - \Gamma_{k+1})) \right)^{1/2},
$$

Note that $L_i := \gamma_i \left[ \prod_{j=i+1}^{n} \left( 1 - 2\gamma_j \left( \mu \left( 1 - \beta - \frac{\gamma_j}{2} \right) + [1 + \beta(3 - \beta)] B(s_0) \right) \right) \right]^{1/2}$.
Proof Outline

Let $z_n = \theta_n - \theta^*$. We first bound the deviation of this error from its mean:

$$
\mathbb{P}(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \geq \epsilon) \leq \exp\left(-\frac{\epsilon^2}{2 \sum_{i=1}^{n} L_i^2}\right), \quad \forall \epsilon > 0,
$$

and then bound the size of the mean itself:

$$
\mathbb{E}\|z_n\|_2 \leq \left[2 \exp(- (1 - \beta) \mu \Gamma_n) \|z_0\|_2 \right. \\
\left. + \left(\sum_{k=1}^{n-1} (3 + 6H)^2 B(s_0)^2 \gamma_{k+1}^2 \exp(-2(1 - \beta) \mu (\Gamma_n - \Gamma_{k+1})\right)\right]^{1/2},
$$

Note that $L_i := \gamma_i \left[\prod_{j=i+1}^{n} \left(1 - 2 \gamma_j \left(\mu \left(1 - \beta - \frac{\gamma_j}{2}\right) + [1 + \beta(3 - \beta)] B(s_0)\right)\right)\right]^{1/2}$
Proof Outline: Bound in Expectation

Let \( f_X(n)(\theta) := [r(s_n, \pi(s_n)) + \beta \theta^T_{n-1} \phi(s_{n+1}) - \theta^T_{n-1} \phi(s_n)] \phi(s_n) \). Then, TD update is equivalent to

\[
\theta_{n+1} = \theta_n + \gamma_n [\mathbb{E}_\Psi (f_X(n)(\theta_n)) + \epsilon_n + \Delta M_n]
\]  

(1)

Mixing error \( \epsilon_n := \mathbb{E}(f_X(n)(\theta_n) | s_0) - \mathbb{E}_\Psi(f_X(n)(\theta_n)) \)

Martingale sequence \( \Delta M_n := f_X(n)(\theta_n) - \mathbb{E}(f_X(n)(\theta_n) | s_0) \)

Unrolling (1), we obtain:

\[
z_{n+1} = (I - \gamma_nA)z_n + \gamma_n (\epsilon_n + \Delta M_n) \\
= \Pi_n z_0 + \sum_{k=1}^{n} \gamma_k \Pi_n \Pi_k^{-1} (\epsilon_k + \Delta M_k)
\]

Here \( A := \Phi^T \Psi(I - \beta P) \Phi \) and \( \Pi_n := \prod_{k=1}^{n} (I - \gamma_k A) \).
Proof Outline: Bound in Expectation

Let \( f_X_n(\theta) := [r(s_n, \pi(s_n)) + \beta \theta_{n-1}^T \phi(s_{n+1}) - \theta_{n-1}^T \phi(s_n)] \phi(s_n) \). Then, TD update is equivalent to

\[
\theta_{n+1} = \theta_n + \gamma_n \left[ \mathbb{E}_\Psi (f_X_n(\theta_n)) + \epsilon_n + \Delta M_n \right]
\]  

(1)

Mixing error \( \epsilon_n := \mathbb{E}(f_X_n(\theta_n) \mid s_0) - \mathbb{E}_\Psi (f_X_n(\theta_n)) \)

Martingale sequence \( \Delta M_n := f_X_n(\theta_n) - \mathbb{E}(f_X_n(\theta_n) \mid s_0) \)

Unrolling (1), we obtain:

\[
z_{n+1} = (I - \gamma_n A) z_n + \gamma_n (\epsilon_n + \Delta M_n)
= \Pi_n z_0 + \sum_{k=1}^{n} \gamma_k \Pi_n \Pi_k^{-1} (\epsilon_k + \Delta M_k)
\]

Here \( A := \Phi^T \Psi (I - \beta P) \Phi \) and \( \Pi_n := \prod_{k=1}^{n} (I - \gamma_k A) \).
Proof Outline: Bound in Expectation

By Jensen’s inequality, we obtain

\[ \mathbb{E}(\|z_n\|^2 | s_0) \leq \left( \mathbb{E}(\langle z_n, z_n \rangle) | s_0 \right)^{\frac{1}{2}} \]

\[ \leq \left( 2 \left\| \Pi_n z_0 \right\|_2^2 + 3 \sum_{k=1}^{n} \gamma_k^2 \left\| \Pi_n \Pi_k^{-1} \right\|_2^2 \mathbb{E}(\|\epsilon_k\|_2^2 | s_0) \right) + 2 \sum_{k=1}^{n} \gamma_k^2 \left\| \Pi_n \Pi_k^{-1} \right\|_2^2 \mathbb{E}(\|\Delta M_k\|_2^2 | s_0) \right)^{\frac{1}{2}} \]

Rest of the proof amounts to bounding each of the terms on RHS above.
Proof Outline: High Probability Bound

Recall $z_n = \theta_n - \theta^*$.  

**Step 1: (Error decomposition)**

$$
\|z_n\|_2^2 - \mathbb{E}\|z_n\|_2^2 = \sum_{i=1}^{n} g_i - \mathbb{E}[g_i | \mathcal{F}_{i-1}] = \sum_{i=1}^{n} D_i,
$$

where $D_i := g_i - \mathbb{E}[g_i | \mathcal{F}_{i-1}]$, $g_i := \mathbb{E}[\|z_n\|_2 | \theta_i]$, and $\mathcal{F}_i = \sigma(\theta_1, \ldots, \theta_n)$.

**Step 2: (Lipschitz continuity)**

Functions $g_i$ are Lipschitz continuous with Lipschitz constants $L_i$.

**Step 3: (Concentration inequality)**

$$
P(\|z_n\|_2^2 - \mathbb{E}\|z_n\|_2^2 \geq \epsilon) = P\left(\sum_{i=1}^{n} D_i \geq \epsilon\right) \leq \exp(-\lambda \epsilon) \exp\left(\frac{\alpha^2}{2} \sum_{i=1}^{n} L_i^2\right).
$$
Proof Outline: High Probability Bound

Recall \( z_n = \theta_n - \theta^* \).

**Step 1:** (Error decomposition)

\[
\|z_n\|_2 - \mathbb{E}\|z_n\|_2 = \sum_{i=1}^{n} g_i - \mathbb{E}[g_i | \mathcal{F}_{i-1}] = \sum_{i=1}^{n} D_i,
\]

where \( D_i := g_i - \mathbb{E}[g_i | \mathcal{F}_{i-1}] \), \( g_i := \mathbb{E}[\|z_n\|_2 | \theta_i] \), and \( \mathcal{F}_i = \sigma(\theta_1, \ldots, \theta_n) \).

**Step 2:** (Lipschitz continuity)

Functions \( g_i \) are Lipschitz continuous with Lipschitz constants \( L_i \).

**Step 3:** (Concentration inequality)

\[
\mathbb{P}(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \geq \epsilon) = \mathbb{P}\left( \sum_{i=1}^{n} D_i \geq \epsilon \right) \leq \exp(-\lambda \epsilon) \exp\left( \frac{\alpha \lambda^2}{2} \sum_{i=1}^{n} L_i^2 \right).
\]
Proof Outline: High Probability Bound

Recall \( z_n = \theta_n - \theta^* \).

**Step 1: (Error decomposition)**

\[
\|z_n\|_2 - \mathbb{E}\|z_n\|_2 = \sum_{i=1}^{n} g_i - \mathbb{E}[g_i | \mathcal{F}_{i-1}] = \sum_{i=1}^{n} D_i,
\]

where \( D_i := g_i - \mathbb{E}[g_i | \mathcal{F}_{i-1}] \), \( g_i := \mathbb{E}[\|z_n\|_2 | \theta_i] \), and \( \mathcal{F}_i = \sigma(\theta_1, \ldots, \theta_n) \).

**Step 2: (Lipschitz continuity)**

Functions \( g_i \) are Lipschitz continuous with Lipschitz constants \( L_i \).

**Step 3: (Concentration inequality)**

\[
\mathbb{P}(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \geq \epsilon) = \mathbb{P}\left(\sum_{i=1}^{n} D_i \geq \epsilon\right) \leq \exp(-\lambda \epsilon) \exp\left(\frac{\alpha \lambda^2}{2} \sum_{i=1}^{n} L_i^2\right).
\]
Concentration Bounds: Iterate Averaged TD(0)
Polyak-Ruppert averaging: Bound in expectation

**Bigger step-size + Averaging**

\[
\gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha
\]

\[
\bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n
\]

with \( \alpha \in (1/2, 1) \) and \( c > 0 \)

**Bound in expectation**

\[
E \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{IA}^A(n)}{(n + c)^{\alpha/2}}, \text{ where}
\]

\[
K_{IA}^A(n) := \sqrt{1 + 9B(s_0)^2} \left[ \frac{\|\theta_0 - \theta^*\|_2}{(n + c)^{(1-\alpha)/2}} + \frac{2\beta(1 - \beta)c^\alpha HB(s_0)}{(\mu c^\alpha (1 - \beta)^2)^{\alpha 1+2\alpha/(2(1-\alpha))}} \right]
\]
Polyak-Ruppert averaging: Bound in expectation

Bigger step-size + Averaging

\[ \gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha \]

\[ \bar{\theta}_{n+1} := \frac{(\theta_1 + \ldots + \theta_n)}{n} \]

with \( \alpha \in (1/2, 1) \) and \( c > 0 \)

Bound in expectation

\[ \mathbb{E} \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{1A}^A(n)}{(n + c)^{\alpha/2}}, \text{ where} \]

\[ K_{1A}^A(n) := \sqrt{1 + 9B(s_0)^2} \left[ \frac{\left\| \theta_0 - \theta^* \right\|_2}{(n + c)^{(1 - \alpha)/2}} + \frac{2\beta(1 - \beta)c^\alpha HB(s_0)}{\left(\mu c^\alpha (1 - \beta)^2\right)^{\frac{1 + 2\alpha}{2(1 - \alpha)}}} \right] \]
Iterate averaging: High probability bound

Bigger step-size + Averaging

\[
\gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha
\]

\[
\bar{\theta}_{n+1} := \frac{\theta_1 + \ldots + \theta_n}{n}
\]

High-probability bound

\[
P \left( \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{2A}^I(n)}{(n + c)^{\alpha/2}} \right) \geq 1 - \delta, \text{ where}
\]

\[
K_{2A}^I(n) := \sqrt{\left(1 + 9B(s_0)^2\right) \left( \frac{2\alpha}{\mu \left[ \frac{1 - \beta}{2} + B(s_0) \right] c^\alpha + \frac{2(3\alpha)}{\alpha}} \right) + K_1(n)}
\]
Iterate averaging: High probability bound

**Bigger step-size + Averaging**

\[
\gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha
\]

\[
\bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n
\]

**High-probability bound**

\[
P \left( \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{IA}^2(n)}{(n + c)^{\alpha/2}} \right) \geq 1 - \delta, \text{ where}
\]

\[
K_{IA}^2(n) := \sqrt{(1 + 9B(s_0)^2) \left( \frac{2\alpha}{\mu \left[ \frac{1 - \beta}{2} + B(s_0) \right] c^\alpha} + \frac{2(3\alpha)}{\alpha} \right)} + K_1(n)
\]
Iterate averaging: High probability bound

Bigger step-size + Averaging

\[ \gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha \]

\[ \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n \]

High-probability bound

\[ \mathbb{P} \left( \| \bar{\theta}_n - \hat{\theta}_T \|_2 \leq \frac{K_{IA}^2(n)}{(n + c)^{\alpha/2}} \right) \geq 1 - \delta, \text{ where} \]

\[ \alpha \text{ can be chosen arbitrarily close to 1, resulting in a rate } O \left( \frac{1}{\sqrt{n}} \right). \]
Proof Outline

Let $\bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n$ and $z_n = \bar{\theta}_{n+1} - \theta^*$. Then,

$$\mathbb{P}(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \geq \epsilon) \leq \exp \left( - \frac{\epsilon^2}{2 \sum_{i=1}^{n} L_i^2} \right), \quad \forall \epsilon > 0,$$

where $L_i := \frac{\gamma_i}{n} \left( 1 + \sum_{l=i+1}^{n-1} \prod_{j=i}^{l} (1 - 2\gamma_j \left( \mu \left( 1 - \beta - \frac{\gamma_j}{2} \right) + [1 + \beta(3 - \beta)] B(s_0) \right)) \right)$. With $\gamma_n = (1 - \beta)(c/(c + n))^\alpha$, we obtain

$$\sum_{i=1}^{n} L_i^2 \leq \left[ \frac{2\alpha}{\mu \left( \frac{1-\beta}{2} + B(s_0) \right) c^\alpha} + \frac{5\alpha}{\alpha} \right]^2 \times \frac{1}{n}$$
Proof Outline

Let $\bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n$ and $z_n = \bar{\theta}_{n+1} - \theta^*$. Then,

$$
P(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \geq \epsilon) \leq \exp \left( - \frac{\epsilon^2}{2 \sum_{i=1}^{n} L_i^2} \right), \quad \forall \epsilon > 0,
$$

where $L_i := \frac{\gamma_i}{n} \left( 1 + \sum_{l=i+1}^{n-1} \prod_{j=i}^{l} \left( 1 - 2\gamma_j \left( \mu \left( 1 - \beta - \frac{\gamma_j}{2} \right) + [1 + \beta(3 - \beta)] B(s_0) \right) \right) \right)$.

With $\gamma_n = (1 - \beta)(c/(c + n))^\alpha$, we obtain

$$
\sum_{i=1}^{n} L_i^2 \leq \frac{2\alpha}{\mu \left[ \frac{1 - \beta}{2} + B(s_0) \right] c^\alpha} + \frac{5^\alpha}{\alpha} \times \frac{1}{n} 
$$
Proof outline: Bound in expectation

To bound the expected error we directly average the errors of the non-averaged iterates:

\[
\mathbb{E} \left\| \bar{\theta}_{n+1} - \theta^* \right\|_2 \leq \frac{1}{n} \sum_{k=1}^{n} \mathbb{E} \left\| \theta_k - \theta^* \right\|_2,
\]

and then specialise to the choice of step-size: \( \gamma_n = (1 - \beta) \left( c / (c + n) \right)^\alpha \)

\[
\mathbb{E} \left\| \bar{\theta}_{n+1} - \theta^* \right\|_2 \leq \frac{\sqrt{1 + 9B(s_0)}}{n} \left( \sum_{n=1}^{\infty} \exp(-\mu c(n + c)^{1-\alpha}) \left\| \theta_0 - \theta^* \right\|_2 \right)
\]

\[+ 2\beta H c^\alpha (1 - \beta) \left( \mu c^\alpha (1 - \beta)^2 \right)^{-\alpha \frac{1+2\alpha}{2(1-\alpha)}} (n + c)^{-\frac{\alpha}{2}} \]
Centered TD (CTD)
The Variance Problem

Why does iterate averaging work?

- in TD(0), each iterate introduces a high variance, which must be controlled by the step-size choice
- averaging the iterates reduces the variance of the final estimator
- reduced variance allows for more exploration within the iterates through larger step sizes
Centering: another approach to variance reduction

- instead of averaging iterates one can use an average to guide the iterates
- now all iterates are informed by their history
- constructing this average in epochs allows a constant step-size choice
Centering: The Idea

Recall that for $TD(0)$,

$$
\theta_{n+1} = \theta_n + \gamma_n \left( r(s_n, \pi(s_n)) + \beta \theta_n^T \phi(s_{n+1}) - \theta_n^T \phi(s_n) \right) \phi(s_n) = f_n(\theta_n)
$$

and that $\theta_n \to \theta^*$, the solution of $F(\theta) := \Pi T^\pi (\Phi \theta) - \Phi \theta = 0$.

Centering each iterate:

$$
\theta_{n+1} = \theta_n + \gamma \left( f_n(\theta_n) - f_n(\bar{\theta}_n) + F(\bar{\theta}_n) \right)
$$
Centering: The Idea

\[ \theta_{n+1} = \theta_n + \gamma \left( f_n(\theta_n) - f_n(\bar{\theta}_n) + F(\bar{\theta}_n) \right) \]

Why Centering helps?

- No updates after hitting \( \theta^* \)
- An average guides the updates, resulting in low variance of term (*)
- Allows using a (large) **constant step-size**
- \( O(d) \) update - same as TD(0)
- Working with epochs \( \Rightarrow \) need to store only the averaged iterate \( \bar{\theta}_n \) and an estimate of \( \hat{F}(\bar{\theta}_n) \)
Centering: The Idea

Centered update:

\[ \theta_{n+1} = \theta_n + \gamma \left( f_n(\theta_n) - f_n(\bar{\theta}_n) + F(\bar{\theta}_n) \right) \]

Challenges compared to gradient descent with a accessible cost function

- \( F \) is \textit{unknown} and \textit{inaccessible} in our setting
- To prove convergence bounds one has to cope with the error due to \textit{incomplete mixing}
Centering: The Idea

Centered update:

\[ \theta_{n+1} = \theta_n + \gamma (f_n(\theta_n) - f_n(\bar{\theta}_n) + F(\bar{\theta}_n)) \]

Challenges compared to gradient descent with a accessible cost function

- \( F \) is **unknown** and **inaccessible** in our setting
- To prove convergence bounds one has to cope with the error due to **incomplete mixing**
### Beginning of each epoch

an iterate $\bar{\theta}^{(m)}$ is chosen uniformly at random from the previous epoch

### Epoch run

Set $\theta_{mM} := \bar{\theta}^{(m)}$, and, for $n = mM, \ldots, (m+1)M = 1$

\[
\theta_{n+1} = \theta_n + \gamma \left( f_{X_{in}}(\theta_n) - f_{X_{in}}(\bar{\theta}^{(m)}) + \hat{F}^{(m)}(\bar{\theta}^{(m)}) \right),
\]

where $\hat{F}^{(m)}(\theta) := \frac{1}{M} \sum_{i=(m-1)M}^{mM} f_{X_i}(\theta)$

(2)
Beginning of each epoch, an iterate $\bar{\theta}^{(m)}$ is chosen uniformly at random from the previous epoch

**Epoch run**

Set $\theta_{mM} := \bar{\theta}^{(m)}$, and, for $n = mM, \ldots, (m + 1)M - 1$

$$\theta_{n+1} = \theta_n + \gamma \left( f_{X_{in}}(\theta_n) - f_{X_{in}}(\bar{\theta}^{(m)}) + \hat{F}^{(m)}(\bar{\theta}^{(m)}) \right),$$

where

$$\hat{F}^{(m)}(\theta) := \frac{1}{M} \sum_{i=(m-1)M}^{mM} f_{X_i}(\theta).$$

(2)
Centering: Results

Epoch length and step size choice

Choose $M$ and $\gamma$ such that $C_1 < 1$, where

$$C_1 := \left( \frac{1}{2\mu \gamma M ((1 - \beta) - d^2 \gamma)} + \frac{\gamma d^2}{2((1 - \beta) - d^2 \gamma)} \right)$$

Error bound

$$\| \Phi(\bar{\theta}(m) - \theta^*) \|^2_\Psi \leq C_1^m \left( \| \Phi(\bar{\theta}(0) - \theta^*) \|^2_\Psi \right) + C_2 H (5\gamma + 4) \sum_{k=1}^{m-1} C_1^{(m-2)-k} B_{(k-1)M}^{kM}(s_0),$$

where $C_2 = \gamma / (2M ((1 - \beta) - d^2 \gamma))$ and $B_{(k-1)M}^{kM}$ is an upper bound on the partial sums $\sum_{i=(k-1)M}^{kM} (\mathbb{E}(\phi(s_i) | s_0) - \mathbb{E}_\Psi(\phi(s_i)))$ and $\sum_{i=(k-1)M}^{kM} (\mathbb{E}(\phi(s_i) \phi(s_{i+l}) | s_0) - \mathbb{E}_\Psi(\phi(s_i) \phi(s_{i+l})^T))$, for $l = 0, 1.$
Centering: Results

**Epoch length and step size choice**

Choose $M$ and $\gamma$ such that $C_1 < 1$, where

$$C_1 := \left( \frac{1}{2\mu\gamma M((1 - \beta) - d^2\gamma)} + \frac{\gamma d^2}{2((1 - \beta) - d^2\gamma)} \right)$$

**Error bound**

$$\| \Phi(\bar{\theta}(m) - \theta^*) \|_\Psi^2 \leq C_1^m \left( \| \Phi(\bar{\theta}(0) - \theta^*) \|_\Psi^2 \right)$$

$$+ C_2 H(5\gamma + 4) \sum_{k=1}^{m-1} C_1^{(m-2)-k} B_{(k-1)M}^k(s_0),$$

where $C_2 = \gamma/(2M((1 - \beta) - d^2\gamma))$ and $B_{(k-1)M}^k$ is an upper bound on the partial sums $\sum_{i=(k-1)M}^{kM} (\mathbb{E}(\phi(s_i) \mid s_0) - \mathbb{E}_\Psi(\phi(s_i)))$

and $\sum_{i=(k-1)M}^{kM} (\mathbb{E}(\phi(s_i)\phi(s_{i+l}) \mid s_0) - \mathbb{E}_\Psi(\phi(s_i)\phi(s_{i+l})^\top))$, for $l = 0, 1$. 
Centering: Results cont.

The effect of mixing error
If the Markov chain underlying policy $\pi$ satisfies the following property:

$$|P(s_t = s \mid s_0) - \psi(s)| \leq C\rho^{t/M},$$

then

$$\|\Phi(\bar{\theta}^{(m)} - \theta^*)\|_{\psi}^2 \leq C_1^m \left(\|\Phi(\bar{\theta}^{(0)} - \theta^*)\|_{\psi}^2\right) + CMC_2H(5\gamma + 4) \max\{C_1, \rho^M\}^{(m-1)}$$

When the MDP mixes exponentially fast (e.g. finite state-space MDPs) we get the exponential convergence rate (* only in the first term) otherwise the decay of the error is dominated by the mixing rate.
Centering: Results cont.

**The effect of mixing error**

If the Markov chain underlying policy $\pi$ satisfies the following property:

$$|P(s_t = s \mid s_0) - \psi(s)| \leq C \rho^{t/M},$$

then

$$\|\Phi(\bar{\theta}^{(m)} - \theta^*)\|^2_{\Psi} \leq C_1^m \left(\|\Phi(\bar{\theta}^{(0)} - \theta^*)\|^2_{\Psi}\right) + CMC_2H(5\gamma + 4) \max\{C_1, \rho^M\}^{(m-1)}$$

When the MDP mixes exponentially fast (e.g. finite state-space MDPs) we get the exponential convergence rate

(* only in the first term)

Otherwise the decay of the error is dominated by the mixing rate
Centering: Results cont.

The effect of mixing error
If the Markov chain underlying policy $\pi$ satisfies the following property:

$$|P(s_t = s | s_0) - \psi(s)| \leq C \rho^{t/M},$$

then

$$\|\Phi(\bar{\theta}^{(m)} - \theta^*)\|^2_{\Psi} \leq C_1^m \left(\|\Phi(\bar{\theta}^{(0)} - \theta^*)\|^2_{\Psi}\right) + CMC_2H(5\gamma + 4) \max\{C_1, \rho^M\}^{m-1}$$

When the MDP mixes exponentially fast (e.g. finite state-space MDPs) we get the exponential convergence rate

(* only in the first term)

Otherwise the decay of the error is dominated by the mixing rate
Proof Outline

Let $\overline{f}_{X_{in}}(\theta_n) := f_{X_{in}}(\theta_n) - f_{X_{in}}(\bar{\theta}^{(m)}) + \mathbb{E}_{\Psi}(f_{X_{in}}(\bar{\theta}^{(m)})).$

**Step 1: (Rewriting CTD update)**

$$\theta_{n+1} = \theta_n + \gamma \left( \overline{f}_{X_{in}}(\theta_n) + \epsilon_n \right) \text{ where } \epsilon_n := \mathbb{E}(f_{X_{in}}(\bar{\theta}^{(m)}) \mid F_{mM}) - \mathbb{E}_{\Psi}(f_{X_{in}}(\bar{\theta}^{(m)}))$$

**Step 2: (Bounding the variance of centered updates)**

$$\mathbb{E}_{\Psi} \left( \| \overline{f}_{X_{in}}(\theta_n) \|_2^2 \right) \leq d^2 \left( \| \Phi(\theta_n - \theta^*) \|_{\Psi}^2 + \| \Phi(\bar{\theta}^{(m)} - \theta^*) \|_{\Psi}^2 \right)$$
Proof Outline

Let \( \bar{f}_{X_n}(\theta_n) := f_{X_n}(\theta_n) - f_{X_n}(\tilde{\theta}(m)) + \mathbb{E}_{\Psi}(f_{X_n}(\tilde{\theta}(m))) \).

**Step 1: (Rewriting CTD update)**

\[
\theta_{n+1} = \theta_n + \gamma \left( \bar{f}_{X_n}(\theta_n) + \epsilon_n \right) \text{ where } \epsilon_n := \mathbb{E}(f_{X_n}(\tilde{\theta}(m)) | \mathcal{F}_m) - \mathbb{E}_{\Psi}(f_{X_n}(\tilde{\theta}(m)))
\]

**Step 2: (Bounding the variance of centered updates)**

\[
\mathbb{E}_{\Psi} \left( ||f_{X_n}(\theta_n)||_2^2 \right) \leq d^2 \left( ||\Phi(\theta_n - \theta^*)||_{\Psi}^2 + ||\Phi(\tilde{\theta}(m) - \theta^*)||_{\Psi}^2 \right)
\]
Proof Outline

Step 3: (Analysis for a particular epoch)

\[
\mathbb{E}_\theta_n \|\theta_{n+1} - \theta^*\|^2 \leq \|\theta_n - \theta^\star\|^2 + 2\gamma (\theta_n - \theta^\star)^\top \mathbb{E}_\theta_n [f_{X_i}(\theta_n)] + \gamma^2 \mathbb{E}_\theta_n \|f_{X_i}(\theta_n)\|^2 \\
\leq \|\theta_n - \theta^*\|^2 - 2\gamma ((1 - \beta) - d^2\gamma) \|\Phi(\theta_n - \theta^\star)\|^2 + \gamma^2 d^2 \left(\|\Phi(\bar{\theta}^{(m)} - \theta^\star)\|^2\right) + \gamma^2 \mathbb{E}_\theta_n \|\epsilon_n\|^2
\]

Summing the above inequality over an epoch and noting that

\[
\mathbb{E}_\Psi,\theta_n \|\theta_{n+1} - \theta^*\|^2 \geq 0 \quad \text{and} \quad (\bar{\theta}^{(m)} - \theta^\star)^\top I(\bar{\theta}^{(m)} - \theta^\star) \leq \frac{1}{\mu}(\bar{\theta}^{(m)} - \theta^\star)^\top I^\top \Psi I(\bar{\theta}^{(m)} - \theta^\star),
\]

we obtain the following by setting \(\theta_0 = \bar{\theta}^{(m)}\):

\[
2\gamma M((1 - \beta) - d^2\gamma) \|\Phi(\bar{\theta}^{(m+1)} - \theta^\star)\|^2 \leq \left(\frac{1}{\mu} + \gamma^2 Md^2\right) \left(\|\Phi(\bar{\theta}^{(m)} - \theta^\star)\|^2\right) \]

\[
+ \gamma^2 \sum_{i=(m-1)M}^{mM} \mathbb{E}_{\theta_i} \|\epsilon_i\|^2
\]

The final step is to unroll (across epochs) the final recursion above to obtain the rate for CTD.
TD(0) on a batch
Dilbert’s boss on big data!

Consultants say three quintillion bytes of data are created every day.

It comes from everywhere. It knows all.

According to the book of Wikipedia, its name is ‘Big Data.’
LSTD - A Batch Algorithm

Given dataset $\mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \ldots, T\}$

LSTD approximates the TD fixed point by

$$\hat{\theta}_T = \bar{A}_T^{-1}\bar{b}_T$$

where

$$\bar{A}_T = \frac{1}{T} \sum_{i=1}^{T} \phi(s_i)(\phi(s'_i) - \beta \phi(s'_i))^T$$

$$\bar{b}_T = \frac{1}{T} \sum_{i=1}^{T} r_i \phi(s_i).$$

$O(d^2T)$ Complexity
LSTD - A Batch Algorithm

Given dataset $\mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \ldots, T\}$

**LSTD** approximates the TD fixed point by

$$\hat{\theta}_T = \bar{A}_T^{-1}\bar{b}_T,$$

where $\bar{A}_T = \frac{1}{T} \sum_{i=1}^{T} \phi(s_i)(\phi(s_i) - \beta \phi(s'_i))^\top$

$$\bar{b}_T = \frac{1}{T} \sum_{i=1}^{T} r_i \phi(s_i).$$

$O(d^2 T)$ Complexity
Complexity of LSTD [1]

**LSTD Complexity**
- $O(d^2 T)$ using the Sherman-Morrison lemma or
- $O(d^{2.807})$ using the Strassen algorithm or $O(d^{2.375})$ the Coppersmith-Winograd algorithm

**Figure**: LSPI - a batch-mode RL algorithm for control
Complexity of LSTD [1]

LSTD Complexity

- $O(d^2T)$ using the Sherman-Morrison lemma or
- $O(d^{2.807})$ using the Strassen algorithm or $O(d^{2.375})$ the Coppersmith-Winograd algorithm

Figure: LSPI - a batch-mode RL algorithm for control
Complexity of LSTD [2]

Problem
Practical applications involve high-dimensional features (e.g. Computer-Go: \( d \sim 10^6 \)) \( \Rightarrow \) solving LSTD is computationally intensive

Related works: GTD \(^1\), GTD2 \(^2\), iLSTD \(^3\)

Solution
Use stochastic approximation (SA)

Complexity \( O(dT) \) \( \Rightarrow \) \( O(d) \) reduction in complexity

Theory SA variant of LSTD does not impact overall rate of convergence

Experiments On traffic control application, performance of SA-based LSTD is comparable to LSTD, while gaining in runtime!

---

1 Sutton et al. (2009) A convergent O(n) algorithm for off-policy temporal difference learning. In: NIPS
3 Geramifard A et al. (2007) iLSTD: Eligibility traces and convergence analysis. In: NIPS
Complexity of LSTD [2]

Problem
Practical applications involve high-dimensional features (e.g. Computer-Go: $d \sim 10^6$) ⇒ solving LSTD is computationally intensive

Related works: GTD ¹, GTD2 ², iLSTD ³

Solution
Use stochastic approximation (SA)

Complexity $O(dT) \Rightarrow O(d)$ reduction in complexity

Theory SA variant of LSTD does not impact overall rate of convergence

Experiments On traffic control application, performance of SA-based LSTD is comparable to LSTD, while gaining in runtime!

³ Geramifard A et al. (2007) iLSTD: Eligibility traces and convergence analysis. In: NIPS
Fast LSTD using Stochastic Approximation

- **Random Sampling**
  - Pick $i_n$ uniformly in $\{1, \ldots, T\}$

- **SA Update**
  - Update $\theta_n$ using $(s_{i_n}, r_{i_n}, s'_{i_n})$

**Update rule:**

$$
\theta_n = \theta_{n-1} + \gamma_n \left( r_{i_n} + \beta \theta_{n-1}^T \phi(s'_{i_n}) - \theta_{n-1}^T \phi(s_{i_n}) \right) \phi(s_{i_n})
$$

- **Step-sizes**
  - Fixed-point iteration

- **Complexity:** $O(d)$ per iteration
Fast LSTD using Stochastic Approximation

\[ \theta_n = \theta_{n-1} + \gamma_n \left( r_{i_n} + \beta \theta_{n-1}^T \phi(s_{i_n}') - \theta_{n-1}^T \phi(s_{i_n}) \right) \phi(s_{i_n}) \]

Step-sizes

Fixed-point iteration

Complexity: \( O(d) \) per iteration
Assumptions

Setting: Given dataset $\mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \ldots, T\}$

(A1) $\|\phi(s_i)\|_2 \leq 1$

(A2) $|r_i| \leq R_{\text{max}} < \infty$

(A3) $\lambda_{\min} \left( \frac{1}{T} \sum_{i=1}^{T} \phi(s_i)\phi(s'_i)^T \right) \geq \mu$. 

- Bounded features
- Bounded rewards
- Co-variance matrix has a min-eigenvalue
Assumptions

Setting: Given dataset \( \mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \ldots, T\} \)

(A1) \[ \| \phi(s_i) \|_2 \leq 1 \]

(A2) \[ |r_i| \leq R_{\text{max}} < \infty \]

(A3) \[ \lambda_{\text{min}} \left( \frac{1}{T} \sum_{i=1}^{T} \phi(s_i) \phi(s_i)^T \right) \geq \mu. \]

Bounded features

Bounded rewards

Co-variance matrix has a min-eigenvalue
Assumptions

Setting: Given dataset $\mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \ldots, T\}$

(A1) $\|\phi(s_i)\|_2 \leq 1$

(Bounded features)

(A2) $|r_i| \leq R_{\text{max}} < \infty$

(Bounded rewards)

(A3) $\lambda_{\text{min}} \left( \frac{1}{T} \sum_{i=1}^{T} \phi(s_i) \phi(s'_i)^T \right) \geq \mu$

(Co-variance matrix has a min-eigenvalue)
Assumptions

Setting: Given dataset \( \mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \ldots, T\} \)

(A1) \( \|\phi(s_i)\|_2 \leq 1 \) \quad \text{Bounded features}

(A2) \( |r_i| \leq R_{\max} < \infty \) \quad \text{Bounded rewards}

(A3) \( \lambda_{\min} \left( \frac{1}{T} \sum_{i=1}^{T} \phi(s_i)\phi(s_i)^T \right) \geq \mu. \) \quad \text{Co-variance matrix has a min-eigenvalue}
Convergence Rate

Step-size choice

\[ \gamma_n = \frac{(1 - \beta)c}{2(c + n)}, \quad \text{with} \quad (1 - \beta)^2 \mu c \in (1.33, 2) \]

Bound in expectation

\[ \mathbb{E} \left\| \theta_n - \hat{\theta}_T \right\|_2^2 \leq \frac{K_1}{\sqrt{n + c}} \]

High-probability bound

\[ \mathbb{P} \left( \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_2}{\sqrt{n + c}} \right) \geq 1 - \delta, \]

By iterate-averaging, the dependency of \( c \) on \( \mu \) can be removed.
Convergence Rate

Step-size choice

\[ \gamma_n = \frac{(1 - \beta)c}{2(c + n)}, \quad \text{with} \quad (1 - \beta)^2 \mu c \in (1.33, 2) \]

Bound in expectation

\[ \mathbb{E} \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_1}{\sqrt{n + c}} \]

High-probability bound

\[ \mathbb{P} \left( \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_2}{\sqrt{n + c}} \right) \geq 1 - \delta, \]

By iterate-averaging, the dependency of \( c \) on \( \mu \) can be removed.
Convergence Rate

Step-size choice

\[ \gamma_n = \frac{(1 - \beta)c}{2(c + n)}, \text{ with } (1 - \beta)^2 \mu c \in (1.33, 2) \]

Bound in expectation

\[ \mathbb{E} \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_1}{\sqrt{n + c}} \]

High-probability bound

\[ \mathbb{P} \left( \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_2}{\sqrt{n + c}} \right) \geq 1 - \delta, \]

By iterate-averaging, the dependency of \( c \) on \( \mu \) can be removed.
Convergence Rate

Step-size choice

\[ \gamma_n = \frac{(1 - \beta)c}{2(c + n)}, \]

with \((1 - \beta)^2 \mu c \in (1.33, 2)\)

Bound in expectation

\[ \mathbb{E} \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_1}{\sqrt{n + c}} \]

High-probability bound

\[ \mathbb{P} \left( \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_2}{\sqrt{n + c}} \right) \geq 1 - \delta, \]

By iterate-averaging, the dependency of \(c\) on \(\mu\) can be removed
The constants

\[ K_1(n) = \frac{\sqrt{c} \left\| \theta_0 - \hat{\theta}_T \right\|_2}{n^{(1 - \beta) \mu c - 1}/2} + \frac{(1 - \beta) c h^2(n)}{2}, \]

\[ K_2(n) = \frac{(1 - \beta) c \sqrt{\log \delta^{-1}}}{2 \sqrt{\left( \frac{4}{3} (1 - \beta)^2 \mu c - 1 \right)}} + K_1(n), \]

where

\[ h(k) := (1 + R_{\max} + \beta)^2 \max \left( \left( \left\| \theta_0 - \hat{\theta}_T \right\|_2 + \ln n + \left\| \hat{\theta}_T \right\|_2 \right)^4, 1 \right) \]

Both \( K_1(n) \) and \( K_2(n) \) are \( O(1) \)
Iterate Averaging

Bigger step-size + Averaging

\[ \gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha \]

\[ \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n \]

Bound in expectation

\[ \mathbb{E} \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2^2 \leq \frac{K_{IA}^1(n)}{(n + c)^{\alpha/2}} \]

High-probability bound

\[ \Pr \left( \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2^2 \leq \frac{K_{IA}^2(n)}{(n + c)^{\alpha/2}} \right) \geq 1 - \delta, \]

Dependency of \( c \) on \( \mu \) is removed dependency at the cost of \((1 - \alpha)/2\) in the rate.
Iterate Averaging

**Bigger step-size + Averaging**

\[ \gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha \]

\[ \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n \]

**Bound in expectation**

\[ \mathbb{E} \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{1A}^A(n)}{(n + c)^{\alpha/2}} \]

**High-probability bound**

\[ \mathbb{P} \left( \left\| \theta_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{2A}^A(n)}{(n + c)^{\alpha/2}} \right) \geq 1 - \delta, \]

Dependency of \( c \) on \( \mu \) is removed dependency at the cost of \((1 - \alpha)/2\) in the rate.
Iterate Averaging

Bigger step-size + Averaging

\[ \gamma_n := \frac{(1 - \beta)}{2} \left( \frac{c}{c + n} \right)^\alpha \]

\[ \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n \]

Bound in expectation

\[ \mathbb{E} \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{1A}^I(n)}{(n + c)^{\alpha/2}} \]

High-probability bound

\[ \mathbb{P} \left( \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_{2A}^I(n)}{(n + c)^{\alpha/2}} \right) \geq 1 - \delta, \]

Dependency of \( c \) on \( \mu \) is removed dependency at the cost of \( (1 - \alpha)/2 \) in the rate.
Iterate Averaging

Bigger step-size + Averaging

\[ \gamma_n := \frac{1 - \beta}{2} \left( \frac{c}{c + n} \right)^\alpha \]

\[ \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n \]

Bound in expectation

\[ \mathbb{E} \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K^{IA}_1(n)}{(n + c)^{\alpha/2}} \]

High-probability bound

\[ \mathbb{P} \left( \left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K^{IA}_2(n)}{(n + c)^{\alpha/2}} \right) \geq 1 - \delta, \]

Dependency of \( c \) on \( \mu \) is removed dependency at the cost of \( (1 - \alpha)/2 \) in the rate.
The constants

\[ K_{1A}^I(n) := \frac{C \left\| \theta_0 - \hat{\theta}_T \right\|_2}{(n + c)^{(1-\alpha)/2}} + \frac{h(n)c^\alpha (1 - \beta)}{(\mu c^\alpha (1 - \beta)^2)^{\frac{1+2\alpha}{2(1-\alpha)}}}, \text{ and} \]

\[ K_{2A}^I(n) := \sqrt{\frac{\log \delta^{-1}}{\mu(1 - \beta)}} \left[ 3^\alpha + \left[ \frac{2\alpha}{\mu c^\alpha (1 - \beta)^2 + \frac{2\alpha}{\alpha}} \right]^2 \right] \frac{1}{(n + c)^{(1-\alpha)/2}} + K_{1A}^I(n). \]

As before, both \( K_{1A}^I(n) \) and \( K_{2A}^I(n) \) are \( O(1) \)
Performance bounds

True value function $v$

Approximate value function $\tilde{v}_n := \Phi \theta_n$

$$\| v - \tilde{v}_n \|_T \leq \frac{\| v - \Pi v \|_T}{\sqrt{1 - \beta^2}} + O \left( \sqrt{\frac{d}{(1 - \beta)^2 \mu T}} \right) + O \left( \sqrt{\frac{1}{(1 - \beta)^2 \mu^2 n \ln \frac{1}{\delta}}} \right)$$

approximation error

estimation error

computational error

---

1 $\| f \|_T^2 := T^{-1} \sum_{i=1}^{T} f(s_i)^2$, for any function $f$.

Performance bounds

\[ \| v - \tilde{v}_n \|_T \leq \frac{\| v - \Pi v \|_T}{\sqrt{1 - \beta^2}} + O\left(\sqrt{\frac{d}{(1 - \beta)^2 \mu T}}\right) + O\left(\sqrt{\frac{1}{(1 - \beta)^2 \mu^2 n \ln \frac{1}{\delta}}}\right) \]

Artifacts of function approximation and least squares methods

Consequence of using SA for LSTD

Setting \( n = \ln(1/\delta)T/(d\mu) \), the convergence rate is unaffected!
Performance bounds

\[ \|v - \tilde{v}_n\|_T \leq \frac{\|v - \Pi v\|_T}{\sqrt{1 - \beta^2}} + O\left(\sqrt{\frac{d}{(1 - \beta)^2 \mu T}}\right) + O\left(\sqrt{\frac{1}{(1 - \beta)^2 \mu^2 n \ln \frac{1}{\delta}}\right) \]

Artifacts of function approximation and least squares methods

Consequence of using SA for LSTD

Setting \( n = \ln(1/\delta)T/(d\mu) \), the convergence rate is unaffected!
Performance bounds

\[ \| v - \tilde{v}_n \|_T \leq \frac{\| v - \Pi v \|_T}{\sqrt{1 - \beta^2}} + O \left( \sqrt{\frac{d}{(1 - \beta)^2 \mu T}} \right) + O \left( \sqrt{\frac{1}{(1 - \beta)^2 \mu^2 n \ln \frac{1}{\delta}}} \right) \]

Artifacts of function approximation and least squares methods

Consequence of using SA for LSTD

Setting \( n = \ln(1/\delta)T/(d\mu) \), the convergence rate is unaffected!
**LSPI - A Quick Recap**

Policy evaluation:

\[
Q^\pi(s, a) = E \left[ \sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \mid s_0 = s, a_0 = a \right]
\]

Policy improvement:

\[
\pi'(s) = \arg \max_{a \in A} \theta^T \phi(s, a)
\]

Prashanth L A
**LSPI - A Quick Recap**

Policy Evaluation

Policy $\pi$

Q-value $Q^\pi$

Policy Improvement

$Q^\pi(s, a) = E \left[ \sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \mid s_0 = s, a_0 = a \right]$

$\pi'(s) = \arg \max_{a \in A} \theta^T \phi(s, a)$
Policy Evaluation: LSTDQ and its SA variant

Given a set of samples $D := \{(s_i, a_i, r_i, s'_i), i = 1, \ldots, T\}$

LSTDQ approximates $Q^\pi$ by

$$\hat{\theta}_T = \bar{A}_T^{-1} \bar{b}_T$$

where

$$\bar{A}_T = \frac{1}{T} \sum_{i=1}^{T} \phi(s_i, a_i) (\phi(s_i, a_i) - \beta \phi(s'_i, \pi(s'_i)))^T,$$

and

$$\bar{b}_T = T^{-1} \sum_{i=1}^{T} r_i \phi(s_i, a_i).$$

Fast LSTDQ using SA:

$$\theta_k = \theta_{k-1} + \gamma_k \left( r_{i_k} + \beta \theta_{k-1}^T \phi(s'_{i_k}, \pi(s'_{i_k})) - \theta_{k-1}^T \phi(s_{i_k}, a_{i_k}) \right) \phi(s_{i_k}, a_{i_k})$$
Policy Evaluation: LSTDQ and its SA variant

Given a set of samples $\mathcal{D} := \{(s_i, a_i, r_i, s'_i), i = 1, \ldots, T\}$

LSTDQ approximates $Q^\pi$ by

$$\hat{\theta}_T = \bar{A}_T^{-1} \bar{b}_T$$

where

$$\bar{A}_T = \frac{1}{T} \sum_{i=1}^{T} \phi(s_i, a_i)(\phi(s_i, a_i) - \beta \phi(s'_i, \pi(s'_i)))^\top,$$

and

$$\bar{b}_T = T^{-1} \sum_{i=1}^{T} r_i \phi(s_i, a_i).$$

Fast LSTDQ using SA:

$$\theta_k = \theta_{k-1} + \gamma_k \left( r_{i_k} + \beta \theta_{k-1}^\top \phi(s'_{i_k}, \pi(s'_{i_k})) - \theta_{k-1}^\top \phi(s_{i_k}, a_{i_k}) \right) \phi(s_{i_k}, a_{i_k})$$
Fast LSPI using SA (fLSPI-SA)

**Input:** Sample set $D := \{s_i, a_i, r_i, s'_i\}_{i=1}^T$

repeat

**Policy Evaluation**

For $k = 1$ to $\tau$

- Get random sample index: $i_k \sim U(\{1, \ldots, T\})$
- Update fLSTD-SA iterate $\theta_k$

$\theta' \leftarrow \theta$, $\Delta = \|\theta - \theta'\|_2$

**Policy Improvement**

Obtain a greedy policy $\pi'(s) = \arg \max_{a \in A} \theta'^T \phi(s, a)$

$\theta \leftarrow \theta'$, $\pi \leftarrow \pi'$

until $\Delta < \epsilon$
Fast LSPI using SA (fLSPI-SA)

**Input:** Sample set \( D := \{s_i, a_i, r_i, s'_i\}_{i=1}^T \)

repeat

**Policy Evaluation**

For \( k = 1 \) to \( \tau \)
- Get random sample index: \( i_k \sim U(\{1, \ldots, T\}) \)
- Update fLSTD-SA iterate \( \theta_k \)

\[ \theta' \leftarrow \theta_\tau, \Delta = \|\theta - \theta'\|_2 \]

**Policy Improvement**

Obtain a greedy policy \( \pi'(s) = \arg\max_{a \in A} \theta'^T \phi(s, a) \)

\[ \theta \leftarrow \theta', \pi \leftarrow \pi' \]

until \( \Delta < \epsilon \)
The traffic control problem
Simulation Results on 7x9-grid network

**Tracking error**

![Tracking error graph]

**Throughput (TAR)**

![Throughput graph]
Runtime Performance on three road networks

- 7x9-Grid ($d = 504$)
  - Runtime: $4,917 \cdot 10^3$
  - LSPI
  - fLSPI-SA

- 14x9-Grid ($d = 1008$)
  - Runtime: $30,144$

- 14x18-Grid ($d = 2016$)
  - Runtime: $1.91 \cdot 10^5$
  - LSPI
  - fLSPI-SA
SGD in Linear Bandits
Complacs News Recommendation Platform

- **NOAM database**: 17 million articles from 2010
- **Task**: Find the best among 2000 news feeds
- **Reward**: Relevancy score of the article
- **Feature dimension**: 80000 (approx)

---

1 In collaboration with Nello Cristianini and Tom Welfare at University of Bristol
Complacs News Recommendation Platform

- **NOAM database**: 17 million articles from 2010
- **Task**: Find the best among 2000 news feeds
- **Reward**: Relevancy score of the article
- **Feature dimension**: 80000 (approx)

---

1. In collaboration with Nello Cristianini and Tom Welfare at University of Bristol
Complacs News Recommendation Platform

- **NOAM database**: 17 million articles from 2010
- **Task**: Find the best among 2000 news feeds
- **Reward**: Relevancy score of the article
- **Feature dimension**: 80000 (approx)

\[\text{In collaboration with Nello Cristianini and Tom Welfare at University of Bristol}\]
Complacs News Recommendation Platform

- **NOAM database**: 17 million articles from 2010
- **Task**: Find the best among 2000 news feeds
- **Reward**: Relevancy score of the article
- **Feature dimension**: 80000 (approx)

\(^1\) In collaboration with Nello Cristianini and Tom Welfare at University of Bristol
More on relevancy score

**Problem:** Find the best news feed for *Crime stories*

**Sample scores:**

- **Five dead in Finnish mall shooting**  
  Score: 1.93

- **Holidays provide more opportunities to drink**  
  Score: −0.48

- **Russia raises price of vodka**  
  Score: 2.67

- **Why Obama Care Must Be Defeated**  
  Score: 0.43

- **University closure due to weather**  
  Score: −1.06
More on relevancy score

**Problem:** Find the best news feed for *Crime stories*

**Sample scores:**

- Five dead in Finnish mall shooting: Score: 1.93
- Holidays provide more opportunities to drink: Score: −0.48
- Russia raises price of vodka: Score: 2.67
- Why Obama Care Must Be Defeated: Score: 0.43
- University closure due to weather: Score: −1.06
More on relevancy score

**Problem:** Find the best news feed for *Crime stories*

**Sample scores:**

- Five dead in Finnish mall shooting  
  Score: 1.93

- Holidays provide more opportunities to drink  
  Score: −0.48

- **Russia raises price of vodka**  
  Score: 2.67

- Why Obama Care Must Be Defeated  
  Score: 0.43

- University closure due to weather  
  Score: −1.06
More on relevancy score

**Problem:** Find the best news feed for *Crime stories*

**Sample scores:**

- *Five dead in Finnish mall shooting*  
  Score: 1.93

- *Holidays provide more opportunities to drink*  
  Score: −0.48

- *Russia raises price of vodka*  
  Score: 2.67

- *Why Obama Care Must Be Defeated*  
  Score: 0.43

- *University closure due to weather*  
  Score: −1.06
More on relevancy score

Problem: Find the best news feed for Crime stories

Sample scores:

- Five dead in Finnish mall shooting: Score: 1.93
- Holidays provide more opportunities to drink: Score: −0.48
- Russia raises price of vodka: Score: 2.67
- Why Obama Care Must Be Defeated: Score: 0.43
- University closure due to weather: Score: −1.06
A linear bandit algorithm

\[ x_n := \arg \max_{x \in D} UCB(x) \]

Rewards \( y_n \)
\[ \text{s.t. } \mathbb{E}[y_n | x_n] = x_n^T \theta^* \]

Choose \( x_n \) \hspace{1cm} Observe \( y_n \)

Estimate UCBs

Regression used to compute
\[ UCB(x) := x^T \hat{\theta}_n + \alpha \sqrt{x^T A_n^{-1} x} \]
A linear bandit algorithm

\[ x_n := \arg \max_{x \in D} UCB(x) \]

Rewards \( y_n \)
\[ \text{s.t. } \mathbb{E}[y_n | x_n] = x_n^T \theta^* \]

Regression used to compute \( UCB(x) := x^T \hat{\theta}_n + \alpha \sqrt{x^TA_n^{-1}x} \)
A linear bandit algorithm

\[ x_n := \arg \max_{x \in D} UCB(x) \]

Estimate UCBs

Rewards \( y_n \)
\[ \text{s.t. } \mathbb{E}[y_n \mid x_n] = x_n^T \theta^* \]

Regression used to compute
\[ UCB(x) := x^T \hat{\theta}_n + \alpha \sqrt{x^T A_n^{-1} x} \]
A linear bandit algorithm

\[ x_n := \arg \max_{x \in D} UCB(x) \]

\[ y_n \]

\[ \text{Choose } x_n \]

\[ \text{Observe } y_n \]

\[ \text{Estimate UCBs} \]

Rewards \( y_n \)

\[ \text{s.t. } \mathbb{E}[y_n | x_n] = x_n^T \theta^* \]

Regression used to compute \( UCB(x) := x^T \hat{\theta}_n + \alpha \sqrt{x^T A_n^{-1} x} \)
UCB values

- Mean-reward estimate

\[ UCB(x) = \hat{\mu}(x) + \alpha \hat{\sigma}(x) \]

- Confidence width

At each round \( t \), select a tap. Optimize the quality of \( n \) selected beers
**UCB values**

- Mean-reward estimate

\[ UCB(x) = \hat{\mu}(x) + \alpha \hat{\sigma}(x) \]

- Confidence width

At each round \( t \), select a tap. Optimize the quality of \( n \) selected beers.
UCB values

- Mean-reward estimate
  \[ UCB(x) = \hat{\mu}(x) + \alpha \hat{\sigma}(x) \]
- Confidence width

At each round \( t \), select a tap. Optimize the quality of \( n \) selected beers
UCB values

**Linearity** \(\Rightarrow\) No need to estimate mean-reward of all arms, estimating \(\theta^*\) is enough

- **Regression** \(\hat{\theta}_n = A_n^{-1}b_n\)

\[
\text{UCB}(x) = \hat{\mu}(x) + \alpha \hat{\sigma}(x)
\]

- Mahalanobis distance of \(x\) from \(A_n: \sqrt{x^TA_n^{-1}x}\)

Optimize the beer you drink, before you get drunk
**UCB values**

Linearity $\Rightarrow$ No need to estimate mean-reward of all arms, estimating $\theta^*$ is enough

- **Regression**
  \[
  \hat{\theta}_n = A_n^{-1} b_n
  \]

- **UCB**
  \[
  UCB(x) = \hat{\mu}(x) + \alpha \hat{\sigma}(x)
  \]

- **Mahalanobis distance of** $x$ **from** $A_n$
  \[
  A_n: \sqrt{x^T A_n^{-1} x}
  \]

---

Optimize the beer you drink, before you get drunk
UCB values

Linearity $\Rightarrow$ No need to estimate mean-reward of all arms, estimating $\theta^*$ is enough

- **Regression** $\hat{\theta}_n = A_n^{-1} b_n$

$$UCB(x) = \hat{\mu}(x) + \alpha \hat{\sigma}(x)$$

- **Mahalanobis distance of $x$ from** $A_n$: $\sqrt{x^T A_n^{-1} x}$

Optimize the beer you drink, before you get drunk
Performance measure

Best arm: $x^* = \arg \min_x \{x^\top \theta^*\}$.

Regret: $R_T = \sum_{i=1}^T (x^* - x_i)^\top \theta^*$

Goal: ensure $R_T$ grows sub-linearly with $T$
Performance measure

Best arm: $x^* = \arg\min_x \{x^T \theta^*\}$.

Regret: $R_T = \sum_{i=1}^{T} (x^* - x_i)^T \theta^*$

Goal: ensure $R_T$ grows sub-linearly with $T$

Linear bandit algorithms ensure sub-linear regret!
Complexity of Least Squares Regression

Regression Complexity

- $O(d^2)$ using the Sherman-Morrison lemma or
- $O(d^{2.807})$ using the Strassen algorithm or $O(d^{2.375})$ the Coppersmith-Winograd algorithm

Problem: Complacs News feed platform has high-dimensional features ($d \sim 10^5$) ⇒ solving OLS is computationally costly
Complexity of Least Squares Regression

Choose $x_n$ → Observe $y_n$ → Estimate $\hat{\theta}_n$

**Figure:** Typical ML algorithm using Regression

**Regression Complexity**
- $O(d^2)$ using the Sherman-Morrison lemma or
- $O(d^{2.807})$ using the Strassen algorithm or $O(d^{2.375})$ the Coppersmith-Winograd algorithm

**Problem:** Complacs News feed platform has **high-dimensional features** ($d \sim 10^5$) ⇒ solving OLS is computationally costly
Fast GD for Regression

\[ \theta_n \xrightarrow{\text{Pick } i_n \text{ uniformly}} \{1, \ldots, n\} \xrightarrow{\text{Random Sampling}} \theta_{n+1} \]

\[ \text{Update } \theta_n \text{ using } (x_{i_n}, y_{i_n}) \]

Solution: Use fast (online) gradient descent (GD)
- Efficient with complexity of only \( O(d) \) (Well-known)
- High probability bounds with explicit constants can be derived (not fully known)
Bandits+GD for News Recommendation

**LinUCB**: a well-known contextual bandit algorithm that employs regression in each iteration

**Fast GD**: provides good approximation to regression (with low computational cost)

**Strongly-Convex Bandits**: no loss in regret except log-factors Proved!

**Non Strongly-Convex Bandits**: Encouraging empirical results for linUCB+fast GD on two news feed platforms
Bandits+GD for News Recommendation

LinUCB: a well-known *contextual bandit* algorithm that employs *regression* in each iteration

Fast GD: provides good approximation to regression (*with low computational cost*)

Strongly-Convex Bandits: no loss in regret except log-factors *Proved!*

Non Strongly-Convex Bandits: Encouraging empirical results for *linUCB+fast GD* on two news feed platforms
fast GD

\[ \theta_n \rightarrow \text{Pick } i_n \text{ uniformly in } \{1, \ldots, n\} \rightarrow \text{Update } \theta_n \text{ using } (x_{i_n}, y_{i_n}) \rightarrow \theta_{n+1} \]

- Random Sampling
- GD Update

- Step-sizes

\[ \theta_n = \theta_{n-1} + \gamma_n (y_{i_n} - \theta_{n-1}^T x_{i_n}) x_{i_n} \]

- Sample gradient
Strongly convex bandits

fast GD

\[ \theta_n \xrightarrow{\text{Pick } i_n \text{ uniformly \ in } \{1, \ldots, n\}} \theta_n \xrightarrow{\text{Update } \theta_n \text{ using } (x_{i_n}, y_{i_n})} \theta_{n+1} \]

- **Step-sizes**
  \[ \theta_n = \theta_{n-1} + \gamma_n (y_{i_n} - \theta_{n-1}^T x_{i_n}) x_{i_n} \]

- **Sample gradient**
fast GD

\[ \theta_n \quad \text{Random Sampling} \quad \begin{array}{c} \text{Pick } i_n \text{ uniformly} \\ \text{in } \{1, \ldots, n\} \end{array} \quad \text{GD Update} \quad \begin{array}{c} \text{Update } \theta_n \\ \text{using } (x_{i_n}, y_{i_n}) \end{array} \quad \theta_{n+1} \]

- **Step-sizes**
  \[ \theta_n = \theta_{n-1} + \gamma_n \left( y_{i_n} - \theta_{n-1}^T x_{i_n} \right) x_{i_n} \]

- **Sample gradient**
Assumptions

Setting: \( y_n = x_n^T \theta^* + \xi_n \), where \( \xi_n \) is i.i.d. zero-mean

(A1) \( \sup_n \|x_n\|_2 \leq 1 \)

(A2) \( |\xi_n| \leq 1, \forall n \)

(A3) \( \lambda_{\min} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu \).
Assumptions

Setting: \( y_n = x_n^T \theta^* + \xi_n \), where \( \xi_n \) is i.i.d. zero-mean

(A1) \( \sup_n \| x_n \|_2 \leq 1 \).

(A2) \( |\xi_n| \leq 1, \forall n \).

(A3) \( \lambda_{\text{min}} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu \).

Bounded features

Bounded noise

Strongly convex co-variance matrix (for each \( n \)!)
Assumptions

Setting: \( y_n = x_n^T \theta^* + \xi_n \), where \( \xi_n \) is i.i.d. zero-mean

(A1) \( \sup_n \| x_n \|_2 \leq 1 \).

(A2) \( |\xi_n| \leq 1, \forall n \).

(A3) \( \lambda_{\min} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu \).

Bounded features

Bounded noise

Strongly convex co-variance matrix (for each \( n \)!)
Assumptions

Setting: \( y_n = x_n^T \theta^* + \xi_n \), where \( \xi_n \) is i.i.d. zero-mean

(A1) \( \sup_n \|x_n\|_2 \leq 1 \).

(A2) \( |\xi_n| \leq 1, \forall n \).

(A3) \( \lambda_{\min} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu \).

Bounded features

Bounded noise

Strongly convex co-variance matrix (for each \( n \))!
Why deriving error bounds is difficult?

\[
\theta_n - \hat{\theta}_n = \theta_n - \hat{\theta}_{n-1} + \hat{\theta}_{n-1} - \hat{\theta}_n \\
= \theta_{n-1} - \hat{\theta}_{n-1} + \hat{\theta}_{n-1} - \hat{\theta}_n + \gamma_n (y_i - \theta_{n-1}^T x_i) x_i \\
= \prod_n (\theta_0 - \theta^*) + \sum_{k=1}^{n} \gamma_k \prod_n \Pi_k^{-1} \Delta \tilde{M}_k - \sum_{k=1}^{n} \prod_n \Pi_k^{-1} (\hat{\theta}_k - \hat{\theta}_{k-1}),
\]

Initial Error

Sampling Error

Drift Error

Present in earlier SGD works and can be handled easily
Consequence of changing target
Hard to control!

Note: \( \bar{A}_n = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T \), \( \Pi_n := \prod_{k=1}^{n} (I - \gamma_k \bar{A}_k) \), and \( \Delta \tilde{M}_k \) is a martingale difference.
Why deriving error bounds is difficult?

\[ \theta_n - \hat{\theta}_n = \theta_n - \hat{\theta}_{n-1} + \hat{\theta}_{n-1} - \hat{\theta}_n \]
\[ = \theta_{n-1} - \hat{\theta}_{n-1} + \hat{\theta}_{n-1} - \hat{\theta}_n + \gamma_n (y_{i_n} - \theta_{n-1}^T x_{i_n}) x_{i_n} \]
\[ = \Pi_n (\theta_0 - \theta^*) + \sum_{k=1}^{n} \gamma_k \Pi_n \Pi_k^{-1} \Delta \tilde{M}_k - \sum_{k=1}^{n} \Pi_n \Pi_k^{-1} (\hat{\theta}_k - \hat{\theta}_{k-1}) , \]

Initial Error

\[ \text{Sampling Error} \]

\[ \text{Drift Error} \]

Present in earlier SGD works and can be handled easily

Consequence of changing target

Hard to control!

Note: \( \bar{A}_n = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T, \quad \Pi_n := \prod_{k=1}^{n} (I - \gamma_k \bar{A}_k) \), and \( \Delta \tilde{M}_k \) is a martingale difference.
Why deriving error bounds is difficult?

\[
\theta_n - \hat{\theta}_n = \theta_n - \hat{\theta}_{n-1} + \hat{\theta}_{n-1} - \hat{\theta}_n \\
\quad = \theta_{n-1} - \hat{\theta}_{n-1} + \hat{\theta}_{n-1} - \hat{\theta}_n + \gamma_n (y_{i_n} - \theta_{n-1}^T x_{i_n}) x_{i_n} \\
= \Pi_n (\theta_0 - \theta^*) + \sum_{k=1}^{n} \gamma_k \Pi_n \Pi_k^{-1} \Delta \tilde{M}_k - \sum_{k=1}^{n} \Pi_n \Pi_k^{-1} (\hat{\theta}_k - \hat{\theta}_{k-1}),
\]

Initial Error

Sampling Error

Drift Error

Present in earlier SGD works and can be handled easily

Consequence of changing target

Hard to control!

Note: \( \bar{A}_n = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T, \Pi_n := \prod_{k=1}^{n} (I - \gamma_k \bar{A}_k), \) and \( \Delta \tilde{M}_k \) is a martingale difference.
Handling Drift Error

Note $F_n(\theta) := \frac{1}{2} \sum_{i=1}^{n} (y_i - \theta^T x_i)^2$ and $\bar{A}_n = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T$. Also, $\mathbb{E}[y_n \mid x_n] = x_n^T \theta^*$.

To control the drift error, we observe that

\[
\left( \nabla F_n(\hat{\theta}_n) = 0 = \nabla F_{n-1}(\hat{\theta}_{n-1}) \right) \quad \Rightarrow \quad \left( \hat{\theta}_{n-1} - \hat{\theta}_n = \xi_n A_{n-1}^{-1} x_n - (x_n^T (\hat{\theta}_n - \theta^*)) A_{n-1}^{-1} x_n \right).
\]

Thus, drift is controlled by the convergence of $\hat{\theta}_n$ to $\theta^*$

Key: confidence ball result

---

Handling Drift Error

Note $F_n(\theta) := \frac{1}{2} \sum_{i=1}^{n} (y_i - \theta^T x_i)^2$ and $\bar{A}_n = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T$. Also, $\mathbb{E}[y_n | x_n] = x_n^T \theta^*$. 

To control the drift error, we observe that

$$\left( \nabla F_n(\hat{\theta}_n) = 0 = \nabla F_{n-1}(\hat{\theta}_{n-1}) \right)$$

$$\implies \left( \hat{\theta}_{n-1} - \hat{\theta}_n = \xi_n A_{n-1}^{-1} x_n - (x_n^T (\hat{\theta}_n - \theta^*)) A_{n-1}^{-1} x_n \right).$$

Thus, drift is controlled by the convergence of $\hat{\theta}_n$ to $\theta^*$.

Key: confidence ball result

---

Handling Drift Error

Note $F_n(\theta) := \frac{1}{2} \sum_{i=1}^{n} (y_i - \theta^T x_i)^2$ and $\bar{A}_n = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T$. Also, $\mathbb{E}[y_n \mid x_n] = x_n^T \theta^*$. 

To control the drift error, we observe that 

$$\left( \nabla F_n(\hat{\theta}_n) = 0 = \nabla F_{n-1}(\hat{\theta}_{n-1}) \right)$$

$$\implies \left( \hat{\theta}_{n-1} - \hat{\theta}_n = \xi_n A_{n-1}^{-1} x_n - (x_n^T (\hat{\theta}_n - \theta^*)) A_{n-1}^{-1} x_n \right).$$

Thus, drift is controlled by the convergence of $\hat{\theta}_n$ to $\theta^*$

**Key: confidence ball result**

---

Error bound

With \( \gamma_n = c/(4(c + n)) \) and \( \mu c/4 \in (2/3, 1) \) we have:

High prob. bound  For any \( \delta > 0 \),

\[
P \left( \| \theta_n - \hat{\theta}_n \|_2 \leq \sqrt{\frac{K_{\mu,c}}{n} \log \frac{1}{\delta} + \frac{h_1(n)}{\sqrt{n}}} \right) \geq 1 - \delta.
\]

Optimal rate \( O \left( n^{-1/2} \right) \)

Bound in expectation

\[
\mathbb{E} \| \theta_n - \hat{\theta}_n \|_2 \leq \frac{\| \theta_0 - \hat{\theta}_n \|_2}{n^{\mu c}} + \frac{h_2(n)}{\sqrt{n}}.
\]

- Initial error
- Sampling error

---

1. \( K_{\mu,c} \) is a constant depending on \( \mu \) and \( c \) and \( h_1(n) \), \( h_2(n) \) hide log factors.
2. By iterate-averaging, the dependency of \( c \) on \( \mu \) can be removed.
**Error bound**

With $\gamma_n = c/(4(c + n))$ and $\mu c/4 \in (2/3, 1)$ we have:

**High prob. bound**  For any $\delta > 0$,

$$P \left( \|\theta_n - \hat{\theta}_n\|_2 \leq \sqrt{\frac{K_{\mu,c}}{n} \log \frac{1}{\delta}} + \frac{h_1(n)}{\sqrt{n}} \right) \geq 1 - \delta.$$  

**Optimal rate** $O \left( n^{-1/2} \right)$

**Bound in expectation**

$$\mathbb{E} \left( \|\theta_n - \hat{\theta}_n\|_2 \right) \leq \frac{\|\theta_0 - \hat{\theta}_n\|_2}{n^{\mu c}} + \frac{h_2(n)}{\sqrt{n}}.$$  

- **Initial error**
- **Sampling error**

---

1. $K_{\mu,c}$ is a constant depending on $\mu$ and $c$ and $h_1(n)$, $h_2(n)$ hide log factors.

2. By iterate-averaging, the dependency of $c$ on $\mu$ can be removed.
Error bound

With $\gamma_n = c/(4(c + n))$ and $\mu c/4 \in (2/3, 1)$ we have:

High prob. bound  For any $\delta > 0$,

$$P \left( \left\| \theta_n - \hat{\theta}_n \right\|_2 \leq \sqrt{\frac{K_{\mu,c}}{n} \log \frac{1}{\delta} + \frac{h_1(n)}{\sqrt{n}}} \right) \geq 1 - \delta.$$ 

Optimal rate $O \left( n^{-1/2} \right)$

Bound in expectation

- Initial error
- Sampling error

---

1 $K_{\mu,c}$ is a constant depending on $\mu$ and $c$ and $h_1(n), h_2(n)$ hide log factors.
2 By iterate-averaging, the dependency of $c$ on $\mu$ can be removed.
**PEGE Algorithm**

**Input** A basis \( \{b_1, \ldots, b_d\} \in D \) for \( \mathbb{R}^d \).

**For each cycle** \( m = 1, 2, \ldots \) **do**

**Exploration Phase**

For \( i = 1 \) to \( d \)

- Choose arm \( b_i \)
- Observe \( y_i(m) \).

\[
\hat{\theta}_{md} = \frac{1}{m} \left( \sum_{i=1}^{d} b_i b_i^T \right)^{-1} \sum_{i=1}^{m} \sum_{j=1}^{d} b_i y_j(i).
\]

**Exploitation Phase**

Find \( x = \arg \min_{x \in D} \{ \hat{\theta}_{md}^T x \} \)

Choose arm \( x \) \( m \) times consecutively.

---

**PEGE Algorithm**

**Input** A basis $\{b_1, \ldots, b_d\} \in D$ for $\mathbb{R}^d$.

- Pull each of the $d$ basis arms once
- Using losses, compute OLS
- Use OLS estimate to compute a greedy decision
- Pull the greedy arm $m$ times

**For each cycle $m = 1, 2, \ldots$ do**

**Exploration Phase**

For $i = 1$ to $d$

- Choose arm $b_i$
- Observe $y_i(m)$.

**Exploitation Phase**

Find $x = \arg \min_{x \in D} \{\hat{\theta}_m^T x\}$

Choose arm $x$ $m$ times consecutively.

---

**Strongly convex bandits**

# PEGE Algorithm¹

**Input** A basis \( \{b_1, \ldots, b_d\} \in D \) for \( \mathbb{R}^d \).

**For each cycle** \( m = 1, 2, \ldots \) do

**Exploration Phase**

For \( i = 1 \) to \( d \)
- Choose arm \( b_i \)
- Observe \( y_i(m) \).

\[
\hat{\theta}_{md} = \frac{1}{m} \left( \sum_{i=1}^{d} b_i b_i^T \right)^{-1} \sum_{i=1}^{m} \sum_{j=1}^{d} b_i y_j(i).
\]

**Exploitation Phase**

Find \( x = \arg \min_{x \in D} \{\hat{\theta}_{md}^T x\} \)

Choose arm \( x \) \( m \) times consecutively.

---

**PEGE Algorithm**

**Input** A basis $\{b_1, \ldots, b_d\} \in D$ for $\mathbb{R}^d$.

- Pull each of the $d$ basis arms once
- Using losses, compute OLS
- Use OLS estimate to compute a greedy decision
- Pull the greedy arm $m$ times

**For each cycle** $m = 1, 2, \ldots$ do

**Exploration Phase**

For $i = 1$ to $d$
- Choose arm $b_i$
- Observe $y_i(m)$.

$$\hat{\theta}_{md} = \frac{1}{m} \left( \sum_{i=1}^{d} b_i b_i^\top \right)^{-1} \sum_{i=1}^{m} \sum_{j=1}^{d} b_i y_j(i).$$

**Exploitation Phase**

Find

$$x = \arg \min_{x \in D} \{ \hat{\theta}_{md}^\top x \}$$

Choose arm $x$ $m$ times consecutively.

---

**Strongly convex bandits**

**PEGE Algorithm with fast GD**

**Input** A basis \( \{b_1, \ldots, b_d\} \in D \) for \( \mathbb{R}^d \).

- Pull each of the \( d \) basis arms once
- Using losses, update fast GD iterate
- Use fast GD iterate to compute a greedy decision
- Pull the greedy arm \( m \) times

**For each cycle** \( m = 1, 2, \ldots \) do

**Exploration Phase**

<table>
<thead>
<tr>
<th>For ( i = 1 ) to ( d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Choose arm ( b_i )</td>
</tr>
<tr>
<td>- Observe ( y_i(m) ).</td>
</tr>
</tbody>
</table>

**Update fast GD iterate** \( \theta_{md} \)

**Exploitation Phase**

Find \( x = \arg\min_{x \in D} \{\theta_{md}^T x\} \)

Choose arm \( x \) \( m \) times consecutively.
**PEGE Algorithm with fast GD**

Input: A basis \( \{b_1, \ldots, b_d\} \in D \) for \( \mathbb{R}^d \).

- Pull each of the \( d \) basis arms once
- Using losses, update fast GD iterate
- Use fast GD iterate to compute a greedy decision
- Pull the greedy arm \( m \) times

For each cycle \( m = 1, 2, \ldots \) do

**Exploration Phase**

For \( i = 1 \) to \( d \)
- Choose arm \( b_i \)
- Observe \( y_i(m) \).

Update fast GD iterate \( \theta_{md} \)

**Exploitation Phase**

Find \( x = \arg \min_{x \in D} \{\theta_{md}^T x\} \)

Choose arm \( x \) \( m \) times consecutively.
PEGE Algorithm with fast GD

**Input** A basis \( \{b_1, \ldots, b_d\} \in D \) for \( \mathbb{R}^d \).

- Pull each of the \( d \) basis arms once
- Using losses, update fast GD iterate
- Use fast GD iterate to compute a greedy decision
- Pull the greedy arm \( m \) times

**For each cycle** \( m = 1, 2, \ldots \) do

**Exploration Phase**

- For \( i = 1 \) to \( d \)
- Choose arm \( b_i \)
- Observe \( y_i(m) \).

**Update fast GD iterate** \( \theta_m \)

**Exploitation Phase**

Find 

\[
    x = \arg \min_{\mathbb{R}^d} \{ \theta_m^T x \}, \quad x \in D
\]

Choose arm \( x \) \( m \) times consecutively.
**PEGE Algorithm with fast GD**

**Input** A basis \( \{b_1, \ldots, b_d\} \in D \) for \( \mathbb{R}^d \).

For each cycle \( m = 1, 2, \ldots \) do

- **Exploration Phase**
  - For \( i = 1 \) to \( d \)
    - Choose arm \( b_i \)
    - Observe \( y_i(m) \).

- **Exploitation Phase**
  - Find \( x = \arg \min_{x \in D} \{ \theta_{md}^T x \} \)
  - Choose arm \( x \) \( m \) times consecutively.
Regret bound for PEGE+fast GD

(Strongly Convex Arms):

(A3) The function $G : \theta \mapsto \arg \min_{x \in D} \{\theta^T x\}$ is $J$-Lipschitz.

Theorem

Under (A1)-(A3), regret $R_T := \sum_{i=1}^{T} x_i^T \theta^* - \min_{x \in D} x^T \theta^*$ satisfies

$$R_T \leq CK_1(n)^2 d^{-1}(\|\theta^*\|_2 + \|\theta^*\|_2^{-1}) \sqrt{T}$$

The bound is worse than that for PEGE by only a factor of $O(\log^4(n))$.
Non-strongly convex bandits

Fast linUCB

Choose $x_n := \arg \max_{x \in D} UCB(x)$

Observe $y_n$

Rewards $y_n$

$s.t. \mathbb{E}[y_n \mid x_n] = x_n^T \theta^*$

Use $\theta_n$ to estimate $\hat{\theta}_n$

Fast GD used to compute $UCB(x) := x^T \theta_n + \alpha \sqrt{x^T \phi_n(x)}$
Non-strongly convex bandits

Fast linUCB

Choose $x_n$ 

Observe $y_n$ 

Rewards $y_n$ 

$s.t. \ E[y_n | x_n] = x_n^T \theta^*$ 

$x_n := \ arg \ max \limits_{x \in D} \ UCB(x)$ 

Use $\theta_n$ to estimate $\hat{\theta}_n$ 

Fast GD used to compute $UCB(x) := x^T \theta_n + \alpha \sqrt{x^T \phi_n(x)}$
Non-strongly convex bandits

**Fast linUCB**

\[ x_n := \arg \max_{x \in D} UCB(x) \]

Rewards \( y_n \)

\[ \text{s.t. } \mathbb{E}[y_n | x_n] = x_n^T \theta^* \]

Choose \( x_n \)  

Observe \( y_n \)

Use \( \theta_n \) to estimate \( \hat{\theta}_n \)

Fast GD used to compute

\[ UCB(x) := x^T \theta_n + \alpha \sqrt{x^T \phi_n(x)} \]
Adaptive regularization

**Problem:** In many settings, \( \lambda_{\text{min}} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu \) may not hold.

**Solution:** Adaptively regularize with \( \lambda_n \)

\[
\tilde{\theta}_n := \arg \min_{\theta} \frac{1}{2n} \sum_{i=1}^{n} (y_i - \theta^T x_i)^2 + \lambda_n \|\theta\|^2
\]

- Pick \( i_n \) uniformly in \( \{1, \ldots, n\} \)
- Update \( \theta_n \) using \( (x_{i_n}, y_{i_n}) \)
- GD Update

GD update:

\[
\theta_n = \theta_{n-1} + \gamma_n \left( (y_{i_n} - \theta_{n-1}^T x_{i_n}) x_{i_n} - \lambda_n \theta_{n-1} \right)
\]
Adaptive regularization

**Problem:** In many settings, $\lambda_{\min} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu$ may not hold.

**Solution:** Adaptively regularize with $\lambda_n$

\[
\tilde{\theta}_n := \arg \min_{\theta}  \frac{1}{2n} \sum_{i=1}^{n} (y_i - \theta^T x_i)^2 + \lambda_n \|\theta\|^2
\]

- Pick $i_n$ uniformly in $\{1, \ldots, n\}$
- Random Sampling
- Update $\theta_n$ using $(x_{i_n}, y_{i_n})$
- GD Update

GD update:

\[
\theta_n = \theta_{n-1} + \gamma_n((y_{i_n} - \theta_{n-1}^T x_{i_n})x_{i_n} - \lambda_n \theta_{n-1})
\]
Adaptive regularization

**Problem:** In many settings, $\lambda_{\min} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu$ may not hold.

**Solution:** Adaptively regularize with $\lambda_n$

\[
\tilde{\theta}_n := \arg \min_{\theta} \frac{1}{2n} \sum_{i=1}^{n} (y_i - \theta^T x_i)^2 + \lambda_n \|\theta\|^2
\]

Pick $i_n$ uniformly in \{1, \ldots, n\}

Random Sampling

Update $\theta_n$ using $(x_{i_n}, y_{i_n})$

GD Update

GD update:

\[
\theta_n = \theta_{n-1} + \gamma_n ((y_{i_n} - \theta_{n-1}^T x_{i_n}) x_{i_n} - \lambda_n \theta_{n-1})
\]
Adaptive regularization

Problem: In many settings, \( \lambda_{\min} \left( \frac{1}{n} \sum_{i=1}^{n-1} x_i x_i^T \right) \geq \mu \) may not hold.

Solution: Adaptively regularize with \( \lambda_n \)

\[
\tilde{\theta}_n := \arg\min_{\theta} \frac{1}{2n} \sum_{i=1}^{n} (y_i - \theta^T x_i)^2 + \lambda_n \|\theta\|^2
\]

Pick \( i_n \) uniformly in \( \{1, \ldots, n\} \)

Random Sampling

Update \( \theta_n \) using \( (x_{i_n}, y_{i_n}) \)

GD Update

GD update:

\[
\theta_n = \theta_{n-1} + \gamma_n ( (y_{i_n} - \theta_{n-1}^T x_{i_n}) x_{i_n} - \lambda_n \theta_{n-1} )
\]
Why deriving error bounds is “really” difficult here?

\[
\theta_n - \tilde{\theta}_n = \Pi_n (\theta_0 - \theta^*) - \sum_{k=1}^{n} \Pi_n \Pi_k^{-1} (\tilde{\theta}_k - \tilde{\theta}_{k-1}) + \sum_{k=1}^{n} \gamma_k \Pi_n \Pi_k^{-1} \Delta \tilde{M}_k,
\]

(3)

Initial Error
Drift Error
Sampling Error

Need \( \sum_{k=1}^{n} \gamma_k \lambda_k \to \infty \) to bound the initial error

Set \( \gamma_n = O(n^{-\alpha}) \) (forcing \( \lambda_n = \Omega(n^{-(1-\alpha)}) \))

Bad news:
This choice when plugged into (3) results in only a constant error bound!

Note: \( \Pi_n := \prod_{k=1}^{n} (I - \gamma_k (A_k + \lambda_k I)) \) and \( \tilde{\theta}_{n-1} - \tilde{\theta}_n = \Omega(n^{-1}) \), whenever \( \alpha \in (0, 1) \)
Why deriving error bounds is “really” difficult here?

\[
\theta_n - \tilde{\theta}_n = \Pi_n (\theta_0 - \theta^*) - \sum_{k=1}^n \Pi_n \Pi_k^{-1} (\tilde{\theta}_k - \tilde{\theta}_{k-1}) + \sum_{k=1}^n \gamma_k \Pi_n \Pi_k^{-1} \Delta \tilde{M}_k, \quad (3)
\]

Initial Error

Drift Error

Sampling Error

Need \( \sum_{k=1}^n \gamma_k \lambda_k \to \infty \) to bound the initial error

Set \( \gamma_n = O(n^{-\alpha}) \) (forcing \( \lambda_n = \Omega(n^{-(1-\alpha)}) \))

Bad news:
This choice when plugged into (3) results in only a constant error bound!

Note: \( \Pi_n := \prod_{k=1}^n \left( I - \gamma_k (\bar{A}_k + \lambda_k I) \right) \) and \( \tilde{\theta}_{n-1} - \tilde{\theta}_n = \Omega(n^{-1}) \), whenever \( \alpha \in (0, 1) \).
Why deriving error bounds is “really” difficult here?

\[
\theta_n - \tilde{\theta}_n = \Pi_n (\theta_0 - \theta^*) - \sum_{k=1}^{n} \Pi_k \Pi_k^{-1} (\tilde{\theta}_k - \tilde{\theta}_{k-1}) + \sum_{k=1}^{n} \gamma_k \Pi_k \Pi_k^{-1} \Delta \tilde{M}_k, \tag{3}
\]

- Initial Error
- Drift Error
- Sampling Error

Need \( \sum_{k=1}^{n} \gamma_k \lambda_k \to \infty \) to bound the initial error

Set \( \gamma_n = O(n^{-\alpha}) \) (forcing \( \lambda_n = \Omega(n^{-(1-\alpha)}) \))

Bad news:
This choice when plugged into (3) results in only a constant error bound!

Note: \( \Pi_n := \prod_{k=1}^{n} (I - \gamma_k (\tilde{A}_k + \lambda_k I)) \) and \( \tilde{\theta}_{n-1} - \tilde{\theta}_n = \Omega(n^{-1}) \), whenever \( \alpha \in (0, 1) \)
Why deriving error bounds is “really” difficult here?

\[
\theta_n - \tilde{\theta}_n = \Pi_n (\theta_0 - \theta^*) - \sum_{k=1}^{n} \Pi_n \Pi_k^{-1} (\tilde{\theta}_k - \tilde{\theta}_{k-1}) + \sum_{k=1}^{n} \gamma_k \Pi_n \Pi_k^{-1} \Delta \tilde{M}_k, \quad (3)
\]

Initial Error

\[
\text{Drift Error}
\]

\[
\text{Sampling Error}
\]

Need \( \sum_{k=1}^{n} \gamma_k \lambda_k \to \infty \) to bound the initial error

Set \( \gamma_n = O(n^{-\alpha}) \) (forcing \( \lambda_n = \Omega(n^{-(1-\alpha)}) \))

Bad news:
This choice when plugged into (3) results in only a constant error bound!

Note: \( \Pi_n := \prod_{k=1}^{n} (I - \gamma_k (\bar{A}_k + \lambda_k I)) \) and \( \tilde{\theta}_{n-1} - \tilde{\theta}_n = \Omega(n^{-1}) \), whenever \( \alpha \in (0, 1) \)
Dilbert’s boss on news recommendation (and ML)

Based on your internet history, you might be dumb enough to enjoy extreme sports.

Click here to buy a ticket to base jump from the international space station.

I think the internet is trying to kill me.

We call it “machine learning.”
Preliminary Results on Complacs News Feed Platform

![Diagram showing cumulative reward over iterations for LinUCB and LinUCB-GD. The graph plots iteration on the x-axis and cumulative reward on the y-axis, with separate lines for each algorithm. The x-axis ranges from 0 to 600 iterations, and the y-axis ranges from 0 to -300 cumulative reward.]
Experiments on Yahoo! Dataset

Figure: The *Featured* tab in Yahoo! Today module

---

1 Yahoo User-Click Log Dataset given under the Webscope program (2011)
News recommendation application

Tracking Error

Tracking error: SGD

Tracking error: SVRG

Tracking error: SAG

---


Runtime Performance on two days of the Yahoo! dataset

<table>
<thead>
<tr>
<th></th>
<th>Day-2</th>
<th>Day-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>LinUCB</td>
<td>1.37 \times 10^6</td>
<td>1.72 \times 10^6</td>
</tr>
<tr>
<td>fLinUCB-GD</td>
<td>4,933</td>
<td>6,474</td>
</tr>
<tr>
<td>fLinUCB-SVRG</td>
<td>81,818</td>
<td>1.07 \times 10^5</td>
</tr>
<tr>
<td>fLinUCB-SAG</td>
<td>44,504</td>
<td>55,630</td>
</tr>
</tbody>
</table>

Convergence rate of TD(0)
For Further Reading

Nathaniel Korda and Prashanth L.A.,
On TD(0) with function approximation: Concentration bounds and a centered variant with exponential convergence.

Prashanth L.A., Nathaniel Korda and Rémi Munos,
Fast LSTD using stochastic approximation: Finite time analysis and application to traffic control.
ECML, 2014.

Nathaniel Korda, Prashanth L.A. and Rémi Munos,
Fast gradient descent for least squares regression: Non-asymptotic bounds and application to bandits.
AAAI, 2015.
Dilbert’s boss (again) on big data!

WE HAVE A GIGANTIC DATABASE FULL OF CUSTOMER BEHAVIOR INFORMATION.

EXCELLENT. WE CAN USE NON-LINEAR MATH AND DATA MINING TECHNOLOGY TO OPTIMIZE OUR RETAIL CHANNELS!

IF THAT’S THE SAME THING AS SPAM, WE’RE HAVING A GOOD MEETING HERE.