Stochastic approximation for speeding up LSTD/LSPI (and least squares regression/LinUCB)

Prashanth L A[†]

Joint work with Nathaniel Korda[♯] and Rémi Munos[†]

[†]INRIA Lille - Team SequeL [♯]MLRG - Oxford University

November 24, 2014

Outline

1. Fast LSTD using SA
2. Fast LSPI using SA
3. Experiments - Traffic Signal Control
4. Extension to Least Squares Regression
5. Experiments - News Recommendation
6. Proof outline

Background

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

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

Bellman Operator
$$\mathcal{T}^{\pi}(V)(s) := r(s, \pi(s)) + \beta \sum_{s'} p(s, \pi(s), s') V(s')$$

TD with Function Approximation

Linear Function Approximation.

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

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

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

TD Fixed Point

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

Feature Matrix

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

Orthogonal Projection

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

TD Fixed Point

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

Feature Matrix

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

Orthogonal Projection

to $\mathcal{B} = \{\Phi\theta \mid \theta \in \mathbb{R}^d\}$

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

$$\boxed{\hat{\theta}_T = \bar{A}_T^{-1}\bar{b}_T}, \xrightarrow{\qquad} \boxed{\boldsymbol{O(d^2T)}\ \textbf{Complexity}}$$

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

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

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

$$\boxed{\hat{\theta}_T = \bar{A}_T^{-1}\bar{b}_T}, \xrightarrow{\qquad} \boxed{\boldsymbol{O(d^2T)}\ \textbf{Complexity}}$$

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

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

Complexity of LSTD [1]

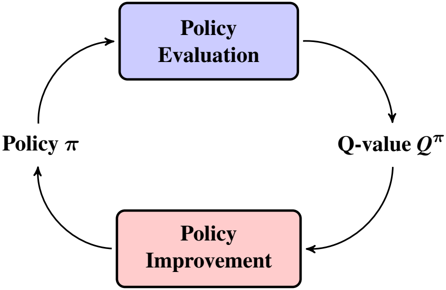


Figure : LSPI - a batch-mode RL algorithm for control

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

Complexity of LSTD [1]

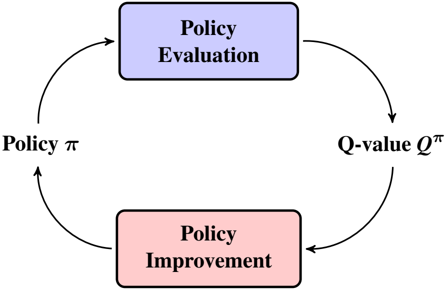


Figure : LSPI - a batch-mode RL algorithm for control

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

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!

[1] Sutton et al. (2009) A convergent O(n) algorithm for off-policy temporal difference learning. In: NIPS

[2] Sutton et al. (2009) Fast gradient-descent methods for temporal-difference learning with linear func- tion approximation. In: ICML

[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: $d \sim 10^6$) $\Rightarrow$ solving LSTD is computationally intensive

Related works: GTD [1], GTD2 [2], iLSTD [3]

Solution

Use stochastic approximation (SA)

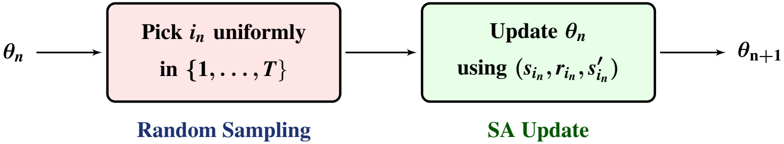
Complexity	$O(dT) \Rightarrow O(d)$ reduction in complexity
Theory	SA variant of LSTD does not impact overall rate of convergence
Experiments	On traffic control application, performance of SA-based LSTD is comparable to LSTD, while gaining in runtime!

[1] Sutton et al. (2009) A convergent O(n) algorithm for off-policy temporal difference learning. In: NIPS

[2] Sutton et al. (2009) Fast gradient-descent methods for temporal-difference learning with linear func- tion approximation. In: ICML

[3] Geramifard A et al. (2007) iLSTD: Eligibility traces and convergence analysis. In: NIPS

Fast LSTD using Stochastic Approximation



θ_n

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

Random Sampling

Update θ_n using $(s_{i_n}, r_{i_n}, s'_{i_n})$

SA Update

$\theta_{\mathbf{n+1}}$

Update rule:

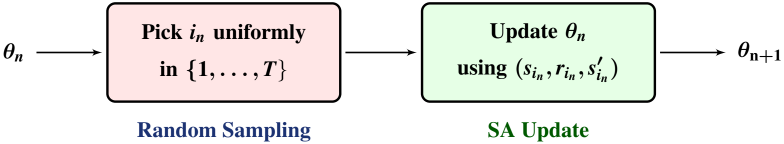
$$\theta_n = \theta_{n-1} + \gamma_n \left(r_{i_n} + \beta\theta_{n-1}^\top\phi(s'_{i_n}) - \theta_{n-1}^\top\phi(s_{i_n})\right)\phi(s_{i_n})$$

Step-sizes

Fixed-point iteration

Complexity: $O(d)$ per iteration

Fast LSTD using Stochastic Approximation



θ_n

Pick i_n uniformly in $\{1,\dots,T\}$

Random Sampling

Update θ_n using $(s_{i_n}, r_{i_n}, s'_{i_n})$

SA Update

θ_{n+1}

Update rule:

$$\theta_n = \theta_{n-1} + \underbrace{\gamma_n}_{\text{Step-sizes}} \underbrace{\left(r_{i_n} + \beta\theta_{n-1}^{\mathsf{T}}\phi(s'_{i_n}) - \theta_{n-1}^{\mathsf{T}}\phi(s_{i_n})\right)\phi(s_{i_n})}_{\text{Fixed-point iteration}}$$

Complexity: $O(d)$ per iteration

Assumptions

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

(A1) $\|\phi(s_i)\|_2 \leq 1$ — Bounded features

(A2) $|r_i| \leq R_{\max} < \infty$ — Bounded rewards

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

Assumptions

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

(A1) $\|\phi(s_i)\|_2 \leq 1$ → Bounded features

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

Bounded rewards

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

Co-variance matrix
has a min-eigenvalue

Assumptions

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

(A1) $\|\phi(s_i)\|_2 \leq 1$ → Bounded features

(A2) $|r_i| \leq R_{\max} < \infty$ → Bounded rewards

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

Assumptions

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

(A1) $\|\phi(s_i)\|_2 \leq 1$ → Bounded features

(A2) $|r_i| \leq R_{\max} < \infty$ → Bounded rewards

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

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 μ can be removed

Convergence Rate

Step-size choice

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

Bound in expectation

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

High-probability bound

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

By iterate-averaging, the dependency of c on μ 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 μ can be removed

Convergence Rate

Step-size choice

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

Bound in expectation

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

High-probability bound

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

By iterate-averaging, the dependency of c on μ can be removed

The constants

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

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

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

$$\gamma_n := \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{\alpha} \qquad \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n$$

Bound in expectation

$$\mathbb{E}\left\|\bar{\theta}_n - \hat{\theta}_T\right\|_2 \leq \frac{K_1^{IA}(n)}{(n+c)^{\alpha/2}}$$

High-probability bound

$$\mathbb{P}\left(\left\|\bar{\theta}_n - \hat{\theta}_T\right\|_2 \leq \frac{K_2^{IA}(n)}{(n+c)^{\alpha/2}}\right) \geq 1-\delta,$$

Dependency of c on μ 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

$$\mathbb{E}\left\|\bar\theta_n - \hat\theta_T\right\|_2 \leq \frac{K_1^{IA}(n)}{(n+c)^{\alpha/2}}$$

High-probability bound

$$\mathbb{P}\left(\left\|\bar\theta_n - \hat\theta_T\right\|_2 \leq \frac{K_2^{IA}(n)}{(n+c)^{\alpha/2}}\right) \geq 1-\delta,$$

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

Iterate Averaging

Bigger step-size + Averaging

$$\gamma_n := \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{\alpha} \qquad \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n$$

Bound in expectation

$$\mathbb{E}\left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_1^{IA}(n)}{(n+c)^{\alpha/2}}$$

High-probability bound

$$\mathbb{P}\left(\left\| \bar{\theta}_n - \hat{\theta}_T \right\|_2 \leq \frac{K_2^{IA}(n)}{(n+c)^{\alpha/2}} \right) \geq 1 - \delta,$$

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

Iterate Averaging

Bigger step-size + Averaging

$$\gamma_n := \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{\alpha} \qquad \bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n$$

Bound in expectation

$$\mathbb{E}\left\|\bar{\theta}_n - \hat{\theta}_T\right\|_2 \leq \frac{K_1^{IA}(n)}{(n+c)^{\alpha/2}}$$

High-probability bound

$$\mathbb{P}\left(\left\|\bar{\theta}_n - \hat{\theta}_T\right\|_2 \leq \frac{K_2^{IA}(n)}{(n+c)^{\alpha/2}}\right) \geq 1-\delta,$$

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

The constants

$$K_1^{IA}(n) := \frac{C\left\|\theta_0 - \hat{\theta}_T\right\|_2}{(n+c)^{(1-\alpha)/2}} + \frac{h(n)c^{\alpha}(1-\beta)}{(\mu c^{\alpha}(1-\beta)^2)^{\alpha\frac{1+2\alpha}{2(1-\alpha)}}}, \text{ and}$$

$$K_2^{IA}(n) := \frac{\sqrt{\log\delta^{-1}}}{\mu(1-\beta)}\left[3^{\alpha} + \left[\frac{2\alpha}{\mu c^{\alpha}(1-\beta)^2} + \frac{2^{\alpha}}{\alpha}\right]^2\right]\frac{1}{(n+c)^{(1-\alpha)/2}} + K_1^{IA}(n).$$

As before, both $K_1^{IA}(n)$ and $K_2^{IA}(n)$ are $O(1)$

Performance bounds

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

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

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

[2] Lazaric, A., Ghavamzadeh, M., Munos, R. (2012) Finite-sample analysis of least-squares policy iteration. In: JMLR

Performance bounds

$$\| v - \tilde{v}_n \|_T \leq \underbrace{\frac{\|v - \Pi v\|_T}{\sqrt{1-\beta^2}}}_{\textbf{approximation error}} + \underbrace{O\left(\sqrt{\frac{d}{(1-\beta)^2 \mu T}}\right)}_{\textbf{estimation error}} + \underbrace{O\left(\sqrt{\frac{1}{(1-\beta)^2 \mu^2 n} \ln \frac{1}{\delta}}\right)}_{\textbf{computational error}}$$ [1]

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

$$\| \, v - \tilde{v}_n \, \|_T \leq \underbrace{\frac{\|v - \Pi v\|_T}{\sqrt{1-\beta^2}}}_{\textbf{approximation error}} + \underbrace{O\left(\sqrt{\frac{d}{(1-\beta)^2 \mu T}}\right)}_{\textbf{estimation error}} + \underbrace{O\left(\sqrt{\frac{1}{(1-\beta)^2 \mu^2 n} \ln \frac{1}{\delta}}\right)}_{\textbf{computational error}}$$ [1]

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

$$\| \, v - \tilde{v}_n \, \|_T \leq \underbrace{\frac{\|v - \Pi v\|_T}{\sqrt{1-\beta^2}}}_{\textbf{approximation error}} + \underbrace{O\left(\sqrt{\frac{d}{(1-\beta)^2 \mu T}}\right)}_{\textbf{estimation error}} + \underbrace{O\left(\sqrt{\frac{1}{(1-\beta)^2 \mu^2 n} \ln \frac{1}{\delta}}\right)}_{\textbf{computational error}}$$ [1]

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!

Outline

1 Fast LSTD using SA

2 Fast LSPI using SA

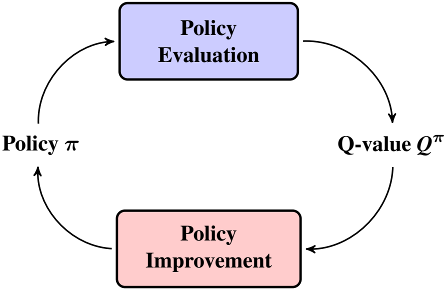
3 Experiments - Traffic Signal Control

4 Extension to Least Squares Regression

5 Experiments - News Recommendation

6 Proof outline

LSPI - A Quick Recap



Policy Evaluation

Q-value Q^π

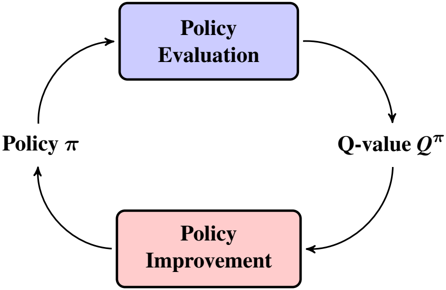
Policy Improvement

Policy π

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

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

LSPI - A Quick Recap



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

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

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^π by

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

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

Fast LSTDQ using SA:

$$\boxed{\theta_k = \theta_{k-1} + \gamma_k \left(r_{i_k} + \beta\theta_{k-1}^\mathsf{T} \phi(s'_{i_k}, \pi(s'_{i_k})) - \theta_{k-1}^\mathsf{T} \phi(s_{i_k}, a_{i_k}) \right) \phi(s_{i_k}, a_{i_k})}$$

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^π by

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

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

Fast LSTDQ using SA:

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

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 $\theta_k$
$\theta' \leftarrow \theta_\tau$, $\Delta = \|\theta - \theta'\|_2$
Policy Improvement
    Obtain a greedy policy $\pi'(s) = \arg\max_{a \in \mathcal{A}} \theta'^{\top} \phi(s, a)$
$\theta \leftarrow \theta'$, $\pi \leftarrow \pi'$
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 $\pi'(s) = \arg\max_{a \in \mathcal{A}} \theta'^{\mathsf{T}} \phi(s, a)$
$\theta \leftarrow \theta', \pi \leftarrow \pi'$
until $\Delta < \epsilon$
```

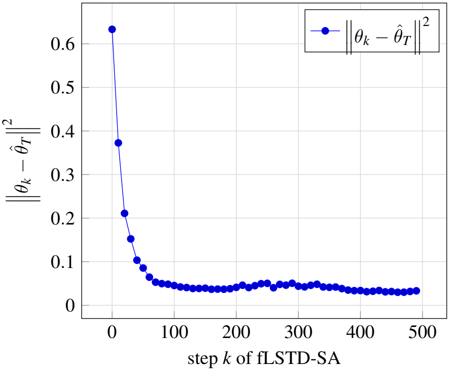
Outline

1. Fast LSTD using SA
2. Fast LSPI using SA
3. Experiments - Traffic Signal Control
4. Extension to Least Squares Regression
5. Experiments - News Recommendation
6. Proof outline

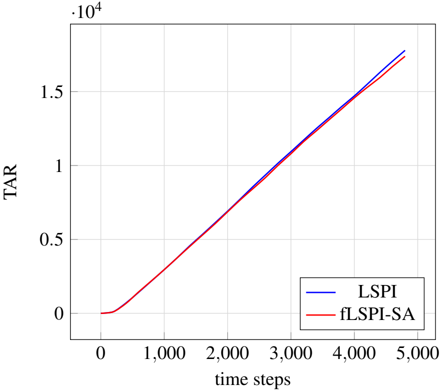
The traffic control problem

Simulation Results on 7x9-grid network

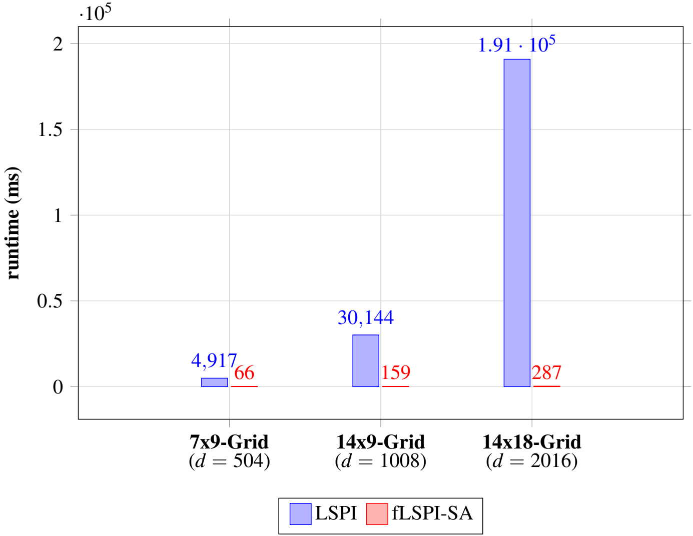
Tracking error



Throughput (TAR)



Runtime Performance on three road networks



Outline

1. Fast LSTD using SA
2. Fast LSPI using SA
3. Experiments - Traffic Signal Control
4. Extension to Least Squares Regression
5. Experiments - News Recommendation
6. Proof outline

Complexity of Ordinary Least Squares (OLS)

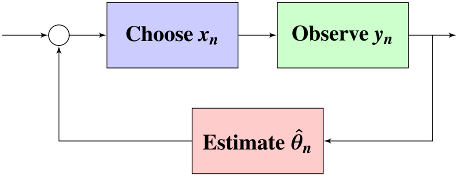


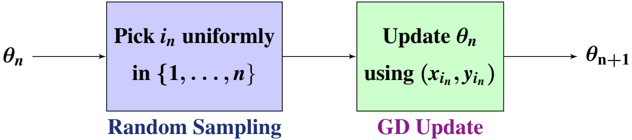
Figure : Typical ML algorithm using Regression

OLS 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: News feed platform has **high-dimensional features** ($d \sim 10^5$) $\Rightarrow$ solving OLS is computationally costly

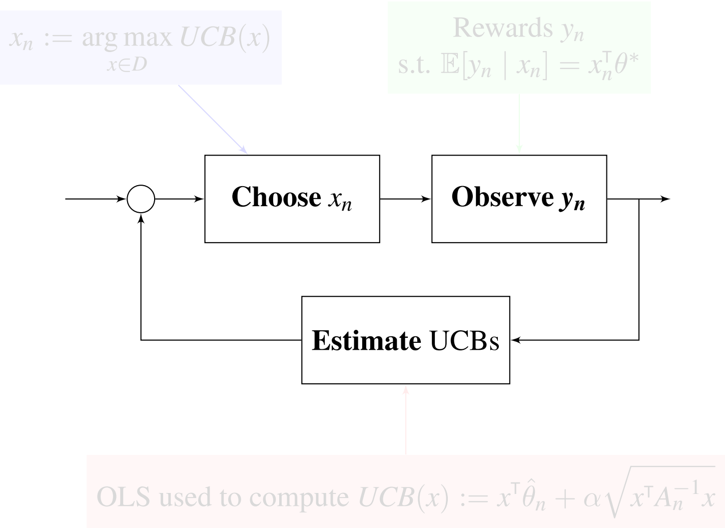
Fast GD for OLS


θ_n
Pick i_n uniformly in $\{1, \ldots, n\}$
Random Sampling
Update θ_n using (x_{i_n}, y_{i_n})
GD Update
$\theta_{\mathbf{n+1}}$

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**)

A linear bandit algorithm



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

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

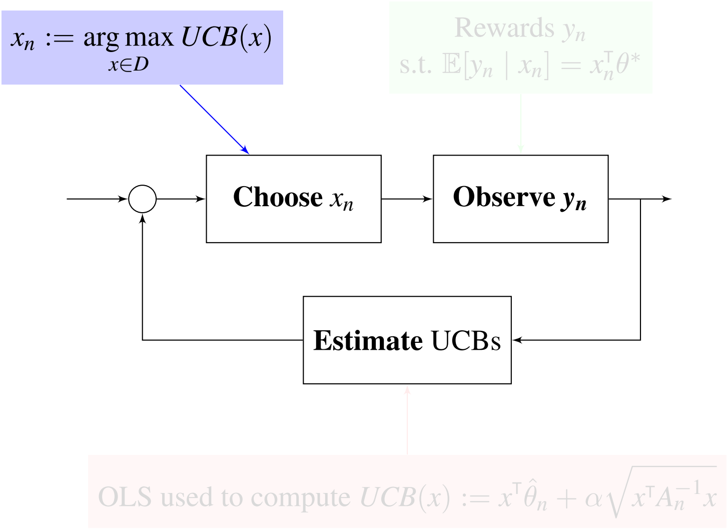
Choose x_n

Observe y_n

Estimate UCBs

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

A linear bandit algorithm



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

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

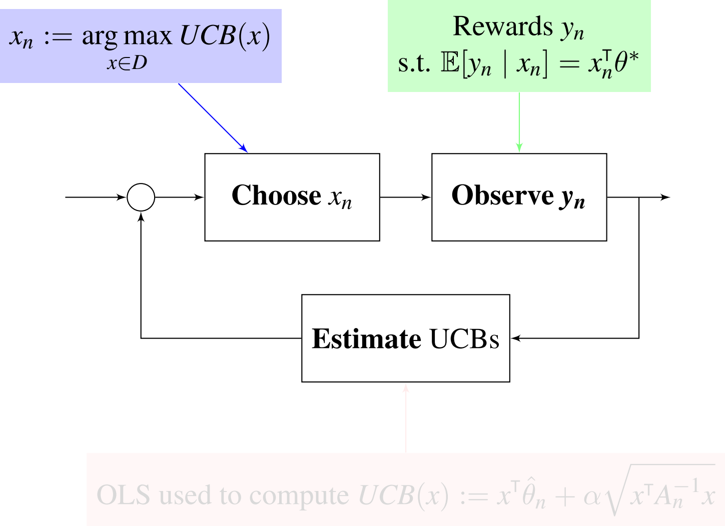
Choose x_n

Observe $\boldsymbol{y_n}$

Estimate UCBs

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

A linear bandit algorithm



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

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

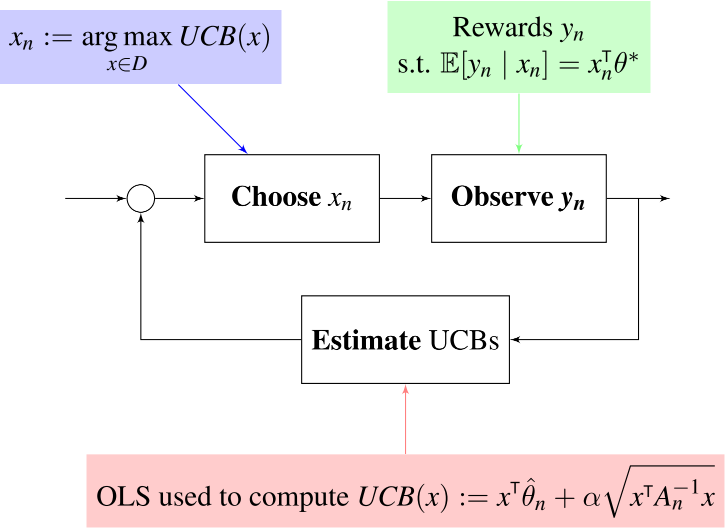
Choose x_n

Observe y_n

Estimate UCBs

OLS used to compute $UCB(x) := x^\intercal \hat{\theta}_n + \alpha \sqrt{x^\intercal A_n^{-1} x}$

A linear bandit algorithm



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

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

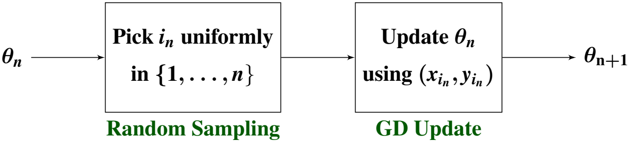
Choose x_n

Observe $\boldsymbol{y_n}$

Estimate UCBs

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

fast GD



θ_n

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

Random Sampling

Update θ_n using (x_{i_n}, y_{i_n})

GD Update

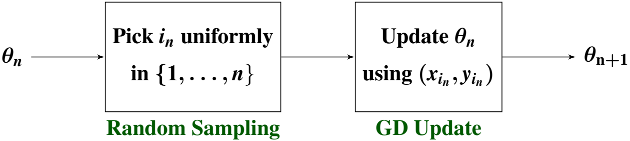
θ_{n+1}

- Step-sizes

$$\theta_n = \theta_{n-1} + \gamma_n \left(y_{i_n} - \theta_{n-1}^\mathsf{T} x_{i_n}\right) x_{i_n}$$

- Sample gradient

fast GD



θ_n

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

Random Sampling

Update θ_n using (x_{i_n}, y_{i_n})

GD Update

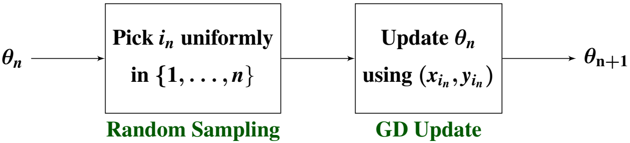
θ_{n+1}

- Step-sizes

$$\theta_n = \theta_{n-1} + \gamma_n \left(y_{i_n} - \theta_{n-1}^{\mathsf{T}} x_{i_n}\right) x_{i_n}$$

- Sample gradient

fast GD



θ_n

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

Random Sampling

Update θ_n using (x_{i_n}, y_{i_n})

GD Update

θ_{n+1}

- Step-sizes

$$\theta_n = \theta_{n-1} + \gamma_n \left(y_{i_n} - \theta_{n-1}^{\mathsf{T}} x_{i_n}\right) x_{i_n}$$

- Sample gradient

Assumptions

Setting: $y_n = x_n^\mathsf{T}\theta^* + \xi_n$, where ξ_n is i.i.d. zero-mean

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

Bounded features

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

Bounded noise

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

Strongly convex co-variance matrix

Assumptions

Setting: $y_n = x_n^\mathsf{T}\theta^* + \xi_n$, where ξ_n is i.i.d. zero-mean

(A1) $\sup\limits_n \|x_n\|_2 \leq 1.$ → Bounded features

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

(A3) $\lambda_{\min}\left(\dfrac{1}{T}\sum\limits_{i=1}^{T} x_i x_i^\mathsf{T}\right) \geq \mu.$ Strongly convex co-variance matrix

Assumptions

Setting: $y_n = x_n^\mathsf{T}\theta^* + \xi_n$, where ξ_n is i.i.d. zero-mean

(A1) $\sup_n \|x_n\|_2 \leq 1.$ → Bounded features

(A2) $|\xi_n| \leq 1, \forall n.$ → Bounded noise

(A3) $\lambda_{\min}\left(\frac{1}{T}\sum_{i=1}^{T} x_i x_i^\mathsf{T}\right) \geq \mu.$ Strongly convex co-variance matrix

Assumptions

Setting: $y_n = x_n^\intercal \theta^* + \xi_n$, where ξ_n is i.i.d. zero-mean

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

Bounded features

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

Bounded noise

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

Strongly convex co-variance matrix

Error bound

With $\gamma_n = \dfrac{c}{2(c+n)}$ with $\mu c \in (1.33, 2)$ we have:

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

$$P\left(\left\|\theta_n - \hat{\theta}_n\right\|_2 \leq \frac{K_2^{LS}}{\sqrt{n+c}}\right) \geq 1 - \delta.$$

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

Bound in expectation

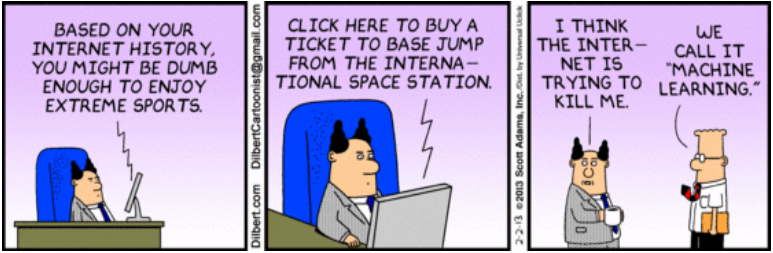
$$\mathbb{E}\left\|\theta_n - \hat{\theta}_n\right\|_2 \leq \frac{K_1^{LS}}{\sqrt{n+c}}$$

[1] By iterate-averaging, the dependency of c on μ can be removed.

Outline

1. Fast LSTD using SA
2. Fast LSPI using SA
3. Experiments - Traffic Signal Control
4. Extension to Least Squares Regression
5. Experiments - News Recommendation
6. Proof outline

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."

Dilbert.com DilbertCartoonist@gmail.com

2-2-13 ©2013 Scott Adams, Inc./Dist. by Universal Uclick

Application to Bandits[1]

Fast linUCB

- **linUCB:** a well-known **contextual bandit** algorithm that employs OLS in each iteration.
- **Fast GD:** provides good approximation to OLS (**with low computational cost**) in each iteration of linUCB
- **Experiments:**

  linUCB+fast GD on Yahoo news recommendation dataset [2]

[1] Thanks to Jérémie Mary and Olivier Nicol for help with the framework (ICML 2012 challenge)

[2] Yahoo! Webscope dataset (2011)

Application to Bandits[1]

Fast linUCB

- **linUCB:** a well-known **contextual bandit** algorithm that employs OLS in each iteration.
- **Fast GD:** provides good approximation to OLS (**with low computational cost**) in each iteration of linUCB
- **Experiments:**

  linUCB+fast GD on Yahoo news recommendation dataset [2]

[1] Thanks to Jérémie Mary and Olivier Nicol for help with the framework (ICML 2012 challenge)

[2] Yahoo! Webscope dataset (2011)

Application to Bandits[1]

Fast linUCB

- **linUCB:** a well-known **contextual bandit** algorithm that employs OLS in each iteration.
- **Fast GD:** provides good approximation to OLS (**with low computational cost**) in each iteration of linUCB
- **Experiments:**

  linUCB+fast GD on Yahoo news recommendation dataset [2]

[1] Thanks to Jérémie Mary and Olivier Nicol for help with the framework (ICML 2012 challenge)

[2] Yahoo! Webscope dataset (2011)

Application to Bandits[1]

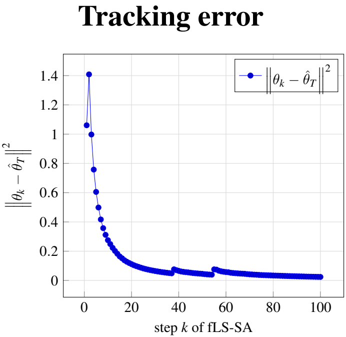
Fast linUCB

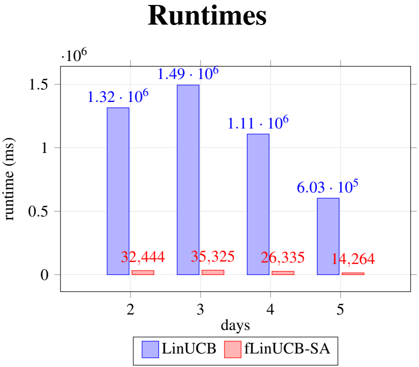
- **linUCB:** a well-known **contextual bandit** algorithm that employs OLS in each iteration.
- **Fast GD:** provides good approximation to OLS (**with low computational cost**) in each iteration of linUCB
- **Experiments:**
  linUCB+fast GD on Yahoo news recommendation dataset [2]

[1] Thanks to Jérémie Mary and Olivier Nicol for help with the framework (ICML 2012 challenge)

[2] Yahoo! Webscope dataset (2011)

Simulation Results





Outline

1 Fast LSTD using SA

2 Fast LSPI using SA

3 Experiments - Traffic Signal Control

4 Extension to Least Squares Regression

5 Experiments - News Recommendation

6 Proof outline

Proof Outline

Let $z_n = \theta_n - \hat{\theta}_T$. Then, first bound the deviation of this error from its mean:

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

and bound the size of the mean itself:

$$\mathbb{E}\left\|z_n\right\|_2 \leq \underbrace{\exp(-(1-\beta)\mu\Gamma_n)\left\|z_0\right\|_2}_{\textbf{initial error}}$$

$$+ \underbrace{\left(\sum_{k=1}^{n-1} h(k)\gamma_{k+1}^2 \exp\left(-2(1-\beta)\mu(\Gamma_n - \Gamma_{k+1})\right)\right)^{\frac{1}{2}}}_{\textbf{sampling error}},$$

Proof Outline

Let $z_n = \theta_n - \hat{\theta}_T$. Then, first bound the deviation of this error from its mean:

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

and bound the size of the mean itself:

$$\mathbb{E}\left\|z_n\right\|_2 \leq \underbrace{\exp(-(1-\beta)\mu\Gamma_n)\left\|z_0\right\|_2}_{\textbf{initial error}}$$

$$+ \underbrace{\left(\sum_{k=1}^{n-1} h(k)\gamma_{k+1}^2 \exp\left(-2(1-\beta)\mu(\Gamma_n - \Gamma_{k+1})\right)\right)^{\frac{1}{2}}}_{\textbf{sampling error}},$$

Proof Outline: High Probability Bound

Step 1: (Error decomposition)

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

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

Step 2: (Lipschitz continuity)

Functions g_i are Lipschitz continuous with Lipschitz constants L_i.

Step 3: (Concentration inequality)

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

Proof Outline: High Probability Bound

Step 1: (Error decomposition)

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

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

Step 2: (Lipschitz continuity)

Functions g_i are Lipschitz continuous with Lipschitz constants L_i.

Step 3: (Concentration inequality)

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

Proof Outline: High Probability Bound

Step 1: (Error decomposition)

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

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

Step 2: (Lipschitz continuity)

Functions g_i are Lipschitz continuous with Lipschitz constants L_i.

Step 3: (Concentration inequality)

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

Proof Outline: Bound in Expectation

Let $f_n(\theta) := (\theta^\intercal \phi(s_{i_n}) - (r_{i_n} + \beta\theta^\intercal\phi(s'_{i_n})))\phi(s_{i_n})$ and $F(\theta) := \mathbb{E}_{i_n}(f_n(\theta))$. Then

$$z_n = \theta_n - \hat{\theta}_T = \theta_{n-1} - \hat{\theta}_T - \gamma_n \left(F(\theta_{n-1}) - \Delta M_n\right),$$

Unrolling the above, noting $F(\hat{\theta}_T)) = 0$ and taking expectations, we obtain:

$$\mathbb{E}(\|z_n\|_2) \leq (\mathbb{E}(\langle z_n, z_n\rangle))^{\frac{1}{2}} = \left(\mathbb{E}\left\|\Pi_n z_0\right\|_2^2 + \sum_{k=1}^{n} \gamma_k^2 \mathbb{E}\left\|\Pi_n \Pi_k^{-1} \Delta M_k\right\|_2^2\right)^{\frac{1}{2}}$$

where $\bar{A}_n = \dfrac{1}{n}\sum_{i=1}^{n} \phi(s_i)(\phi(s_i) - \beta\phi(s'_i))^\intercal$ and $\Pi_n := \prod_{k=1}^{n}\left(I - \gamma_k \bar{A}_k\right)$.

Rest of the proof amounts to bounding each of the terms on RHS above.

Proof Outline: Bound in Expectation

Let $f_n(\theta) := (\theta^\mathsf{T}\phi(s_{i_n}) - (r_{i_n} + \beta\theta^\mathsf{T}\phi(s'_{i_n})))\phi(s_{i_n})$ and $F(\theta) := \mathbb{E}_{i_n}(f_n(\theta))$. Then

$$z_n = \theta_n - \hat{\theta}_T = \theta_{n-1} - \hat{\theta}_T - \gamma_n\left(F(\theta_{n-1}) - \Delta M_n\right),$$

Unrolling the above, noting $F(\hat{\theta}_T)) = 0$ and taking expectations, we obtain:

$$\mathbb{E}(\|z_n\|_2) \leq (\mathbb{E}(\langle z_n, z_n\rangle))^{\frac{1}{2}} = \left(\mathbb{E}\left\|\Pi_n z_0\right\|_2^2 + \sum_{k=1}^{n}\gamma_k^2\mathbb{E}\left\|\Pi_n\Pi_k^{-1}\Delta M_k\right\|_2^2\right)^{\frac{1}{2}}$$

where $\bar{A}_n = \frac{1}{n}\sum_{i=1}^{n}\phi(s_i)(\phi(s_i) - \beta\phi(s'_i))^\mathsf{T}$ and $\Pi_n := \prod_{k=1}^{n}\left(I - \gamma_k\bar{A}_k\right)$.

Rest of the proof amounts to bounding each of the terms on RHS above.

References I

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 drifting least squares regression, with application to bandits.

AAAI, 2015.