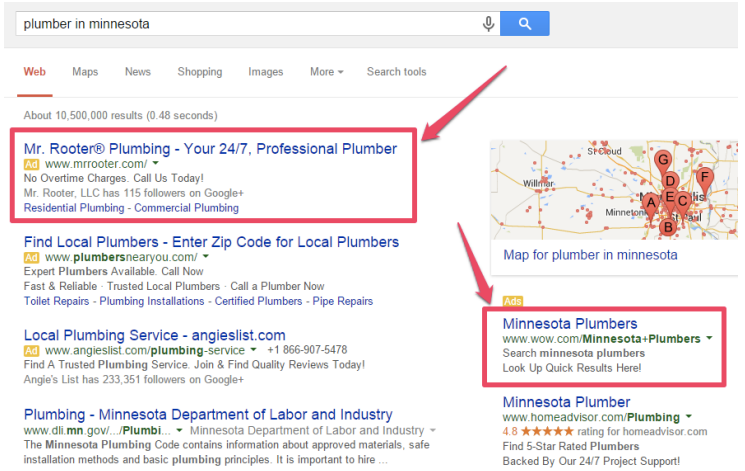
The Price of Differential Privacy for Online Learning

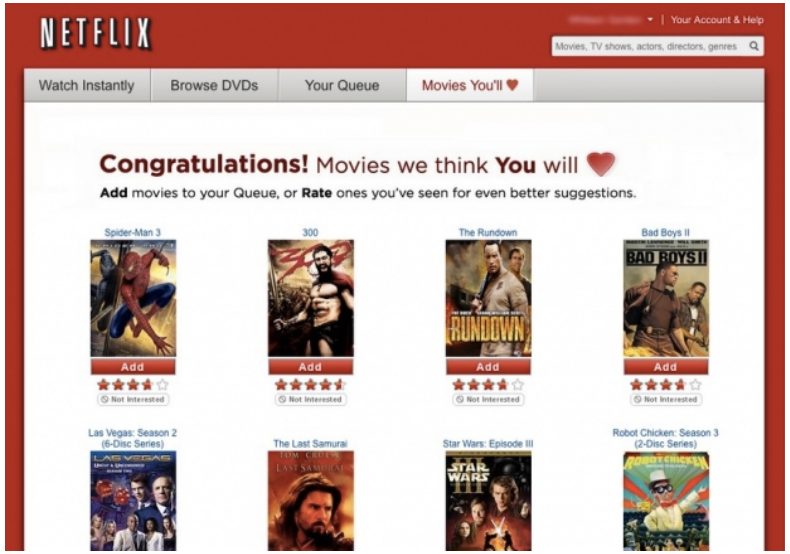
Naman Agarwal, Karan Singh

Sequential Decision Making

Adaptive Clinical Trials



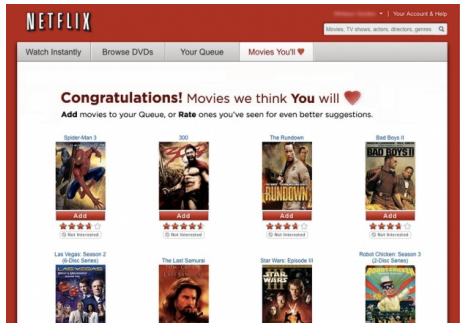
Ad Targeting



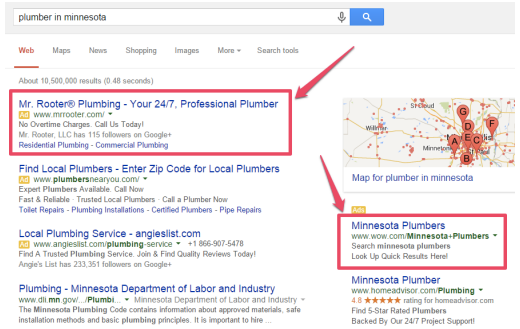
Product Recommendations

The Need for Privacy

Adaptive Clinical Trials



Product Recommendations



Ad Targeting

Learning Algorithms interact with sensitive **user data**.

Framework: Online Learning

No stochastic assumption on the environment.

- On each round $t = 1, 2, \ldots T$
  - The **learner** predicts $x_t \in \mathcal{X} \subseteq \mathbb{R}^N$ from a convex set.
  - Simultaneously, the **adversary** chooses a loss vector $l_t \in \mathcal{Y}$.
  - The learner suffers a loss of $\langle l_t, x_t \rangle$ and observes some **feedback**.
    - **Full Information Setting:** The learner observes the loss vector $l_t \in \mathcal{Y}$.
    - **Bandit Feedback:** The learner only observes the loss value $\langle l_t, x_t \rangle$.

$$\text{Regret} = \mathbb{E}\Bigg[\underbrace{\sum_{t=1}^{T} \langle l_t, x_t \rangle}_{\text{Loss of the learner}} - \underbrace{\min_{x \in \mathcal{X}} \sum_{t=1}^{T} \langle l_t, x \rangle}_{\text{Loss of the best fixed decision}} \Bigg]$$

$$O(\sqrt{T})\ \text{Regret} \implies O\left(\frac{1}{\varepsilon^2}\right)\ \text{Sample Complexity}$$

Framework: Private Online Learning

- A randomized learning algorithm is ϵ-**differentially private** if
  - For any pair of sequence of loss vectors differing in at most **one** vector

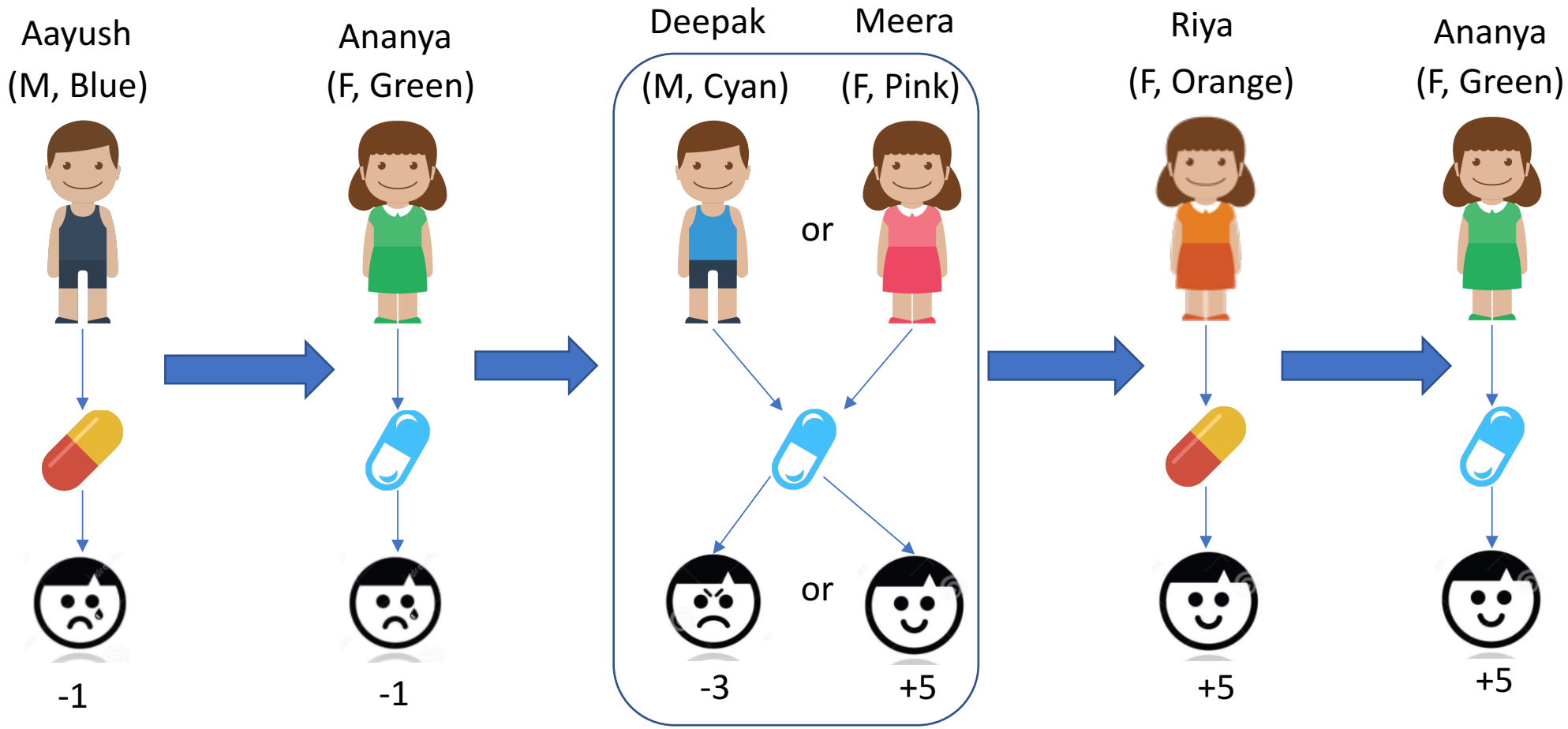
$$L = (l_1, \ldots, l_t, \ldots l_T) \xrightarrow{\mathcal{A}} (x_1, \ldots x_T)$$

$$L' = \underbrace{(l_1, \ldots, l'_t, \ldots l_T)}_{\text{Input}} \xrightarrow{\mathcal{A}} \underbrace{(x'_1, \ldots x'_T)}_{\text{Output}},$$

  - For any possible set $S \subseteq \mathcal{X}^T$ of output sequences, it holds that

$$\mathbb{P}(\underbrace{(x_1, \ldots x_T)}_{\text{Output of } \mathcal{A} \text{ on } L} \in S) \leq e^{\varepsilon} \mathbb{P}(\underbrace{(x'_1, \ldots x'_T)}_{\text{Output of } \mathcal{A} \text{ on } L'} \in S).$$

Illustration: The Promise of Privacy



Loss Vector (in OL) $\equiv$ **Feature Vector + Reward** (in Sup L)

Our Results

- **Full-Information Setting:** Any Follow-the-Regularized-Leader based non-private algorithm can be made ϵ-**differentially private** with

$$\text{Regret}_{\varepsilon\text{-DP}} = \text{Regret}_{\text{Non-private}} + O\left(\frac{\log^2 T}{\varepsilon}\right)$$

  - The regret scales as $\tilde{O}\left(\sqrt{T} + \frac{1}{\epsilon}\right)$.

    Privacy is Free! as long as $\varepsilon \geq \frac{1}{\sqrt{T}}$

  - The previous best bounds scale as $\tilde{O}\left(\frac{\sqrt{T}}{\epsilon}\right)$ [JKT12, ST13].

  - Adapts to the **Geometry** of the problem.
    - Optimal dependence on N.

Our Results

- **Bandit Feedback:** Any non-private low-regret bandit algorithm can be adapted to achieve ϵ-**differentially privacy** with

$$\text{Regret}_{\varepsilon\text{-DP}} = O\left(\frac{\text{Regret}_{\text{Non-private}}}{\varepsilon}\right)$$

  - The regret scales as $\tilde{O}\left(\frac{\sqrt{T}}{\epsilon}\right)$.
    - Optimal dependence $\tilde{O}(\sqrt{T})$ on the number of rounds (up to logarithmic factors).
  - The previous best bounds scale as $O(T^{\frac{2}{3}})$ [ST13].

	Previous Best	Our Regret Bound	Non-private
Mult-armed Bandits	$\tilde{O}\left(\frac{NT^{\frac{2}{3}}}{\varepsilon}\right)$ [ST13]	$\tilde{O}\left(\frac{\sqrt{TN\log N}}{\varepsilon}\right)$	$O(\sqrt{NT})$

Full-Information Setting: Algorithm

FTRL Update

$$x_t = argmin_{x \in \mathcal{X}} \left(\eta \langle \sum_{i=1}^{t-1} l_i, x \rangle + R(x) \right)$$

Prefix Sums

FTRL Template

1 Initialize an empty binary tree B to compute differentially private estimates of $\sum_{i=1}^{t} l_i$.
2 **for** $t = 1$ *to* T **do**
3 $x_t = argmin_{x \in \mathcal{X}}(\eta \langle x, \tilde{L}_{t-1} \rangle + R(x))$.
$\tilde{L}_{t-1} \approx \sum_{i=1}^{t} l_i$+noise
4 Observe l_t, and suffer a loss of $\langle l_t, x_t \rangle$.
5 $(\tilde{L}_t, B) \leftarrow \texttt{TreeBasedAgg}(l_t, B)$.

Full-Information Setting: Algorithm

FTRL Template

1 Initialize an empty binary tree B to compute differentially private estimates of $\sum_{i=1}^{t} l_i$.

2 **for** $t = 1$ *to* T **do**

3 $x_t = argmin_{x \in \mathcal{X}}(\eta \langle x, \tilde{L}_{t-1} \rangle + R(x))$.

$\tilde{L}_{t-1} \approx \sum_{i=1}^{t} l_i$+noise

4 Observe l_t, and suffer a loss of $\langle l_t, x_t \rangle$.

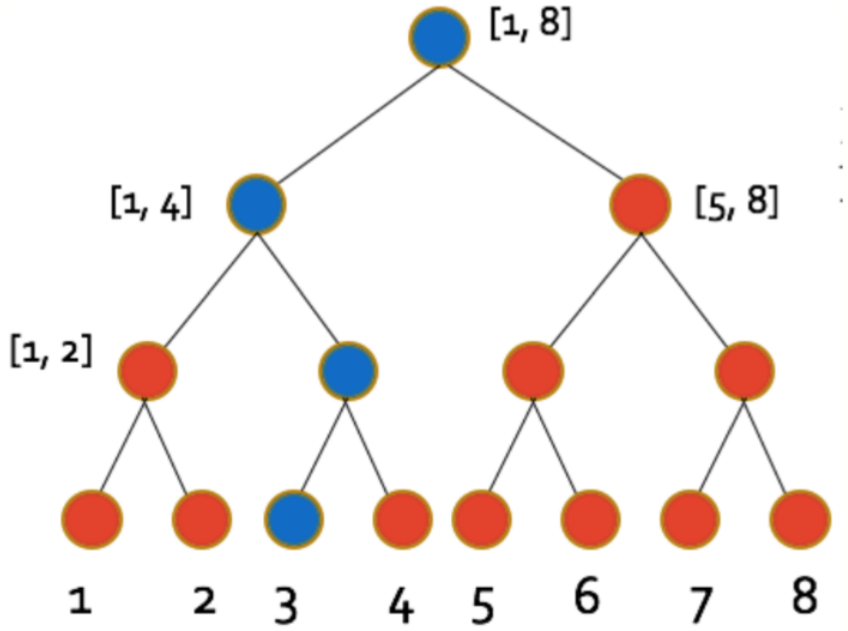
5 $(\tilde{L}_t, B) \leftarrow \texttt{TreeBasedAgg}(l_t, B)$.

Tree-based Aggregation

Input: A sequence of vectors $(l_1, \ldots l_T)$.

Output: ε-DP estimates $\tilde{L}_t$ of sums $\left(\sum_{i=1}^{t} l_i\right)$.

Utility: $|\tilde{L}_t - \sum_{i=1}^{T} l_i| \approx \frac{\log^2 T}{\varepsilon}$. [DNPR10, JKT12]



Full-Information Setting: Privacy Analysis

FTRL Template

1 Initialize an empty binary tree B to compute differentially private estimates of $\sum_{i=1}^{t} l_i$.
2 **for** $t = 1$ *to* T **do**
3 $\quad x_t = argmin_{x \in \mathcal{X}}(\eta \langle x, \tilde{L}_{t-1} \rangle + R(x))$.
$\quad \tilde{L}_{t-1} \approx \sum_{i=1}^{t} l_i$+noise
4 $\quad$ Observe l_t, and suffer a loss of $\langle l_t, x_t \rangle$.
5 $\quad (\tilde{L}_t, B) \leftarrow$ `TreeBasedAgg`(l_t, B).

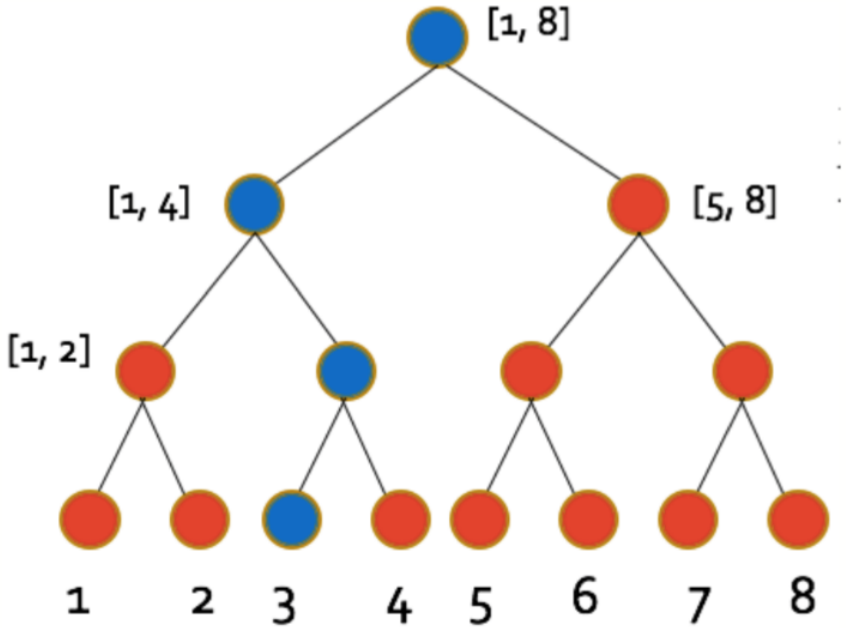
Privacy guaranteed by Tree-based Aggregation Scheme.
The output of the algorithm is completely determined given the estimates of the cumulative sums.

Tree-based Aggregation

Input: A sequence of vectors $(l_1, \ldots l_T)$.

Output: ε-DP estimates $\tilde{L}_t$ of sums $\left(\sum_{i=1}^{t} l_i\right)$.

Utility: $|\tilde{L}_t - \sum_{i=1}^{T} l_i| \approx \frac{\log^2 T}{\varepsilon}$. [DNPR10, JKT12]



Full-Information Setting: Regret Analysis

FTRL Template

1 Initialize an empty binary tree B to compute differentially private estimates of $\sum_{i=1}^{t} l_i$.
2 **for** $t = 1$ *to* T **do**
3 $\quad x_t = argmin_{x \in \mathcal{X}}(\eta \langle x, \tilde{L}_{t-1} \rangle + R(x))$.
$\quad \tilde{L}_{t-1} \approx \sum_{i=1}^{t} l_i$+noise
4 $\quad$ Observe l_t, and suffer a loss of $\langle l_t, x_t \rangle$.
5 $\quad (\tilde{L}_t, B) \leftarrow \texttt{TreeBasedAgg}(l_t, B)$.

Not independent across time.
Not identical in value.

$$x_t = argmin_{x \in \mathcal{X}} \left(\eta \langle z_t + \sum_{i=1}^{t-1} l_i, x \rangle + R(x) \right)$$

where $|z_t| \approx \frac{\log^2 T}{\epsilon}$

$$\text{Regret} \approx \frac{1}{\eta} + \eta \sum_{t=1}^{T} |\tilde{L}_t - \tilde{L}_{t-1}|^2$$

$$\leq \frac{1}{\eta} + 2\eta \sum_{t=1}^{T} \left(|l_t|^2 + |z_t - z_{t-1}|^2 \right) = \frac{1}{\eta} + 2\eta T \frac{\log^2 T}{\epsilon^2} = O\left(\frac{\sqrt{T} \log T}{\epsilon} \right)$$

Full-Information Setting: Regret Analysis II

FTRL Template

1 Initialize an empty binary tree B to compute differentially private estimates of $\sum_{i=1}^{t} l_i$.
2 **for** $t = 1$ *to* T **do**
3 $x_t = argmin_{x \in \mathcal{X}}(\eta\langle x, \tilde{L}_{t-1}\rangle + R(x))$.
$\tilde{L}_{t-1} \approx \sum_{i=1}^{t} l_i$+noise
4 Observe l_t, and suffer a loss of $\langle l_t, x_t\rangle$.
5 $(\tilde{L}_t, B) \leftarrow$ `TreeBasedAgg`(l_t, B).

Modify TBP so that identically distributed.

$$x_t = argmin_{x \in \mathcal{X}} \left(\eta\langle z_t + \sum_{i=1}^{t-1} l_i, x\rangle + R(x) \right)$$

Alternate Algorithm

$$\tilde{x}_t = argmin_{x \in \mathcal{X}} \left(\eta\langle z + \sum_{i=1}^{t-1} l_i, x\rangle + R(x) \right)$$

where $z \sim \mathcal{D}; z_1, \ldots z_T \sim \mathcal{D}$

$\tilde{x}_t, x_t$ are the same in distribution.
Hence, incur the same expected regret.

Full-Information Setting: Regret Analysis III

FTRL Template

1 Initialize an empty binary tree B to compute differentially private estimates of $\sum_{i=1}^{t} l_i$.
2 **for** $t = 1$ *to* T **do**
3 $\quad x_t = argmin_{x \in \mathcal{X}}(\eta\langle x, \tilde{L}_{t-1}\rangle + R(x))$.
$\quad\quad \tilde{L}_{t-1} \approx \sum_{i=1}^{t} l_i$+noise
4 $\quad$ Observe l_t, and suffer a loss of $\langle l_t, x_t\rangle$.
5 $\quad (\tilde{L}_t, B) \leftarrow$ `TreeBasedAgg`(l_t, B).

Alternate Algorithm

$$\tilde{x}_t = argmin_{x \in \mathcal{X}} \left(\eta\langle z + \sum_{i=1}^{t-1} l_i, x\rangle + R(x) \right)$$

where $z_1, \ldots z_T \sim \mathcal{D}$

Equivalent to the adversary playing the loss vector z in round 0 (before the game begins).

$$\text{Regret}_{1:T}(\tilde{x}_t, x_t) \leq \text{Regret}_{0:T} + |z| \approx |z| + \frac{1}{\eta} + \eta\left(|z|^2 + \sum_{t=1}^{T} |l_t|^2\right) = O\left(\sqrt{T} + \frac{\log^2 T}{\epsilon}\right)$$

Bandit Feedback: Algorithm

Reduction to Non-private Setting

Require: Bandit Algorithm $\mathcal{A}$.
for $t = 1$ *to* T **do**
  Receive x_t from $\mathcal{A}$ and output x_t.
  Receive a loss value $\langle l_t, x_t \rangle$ from the adversary.
  Sample $Z_t \sim Lap\left(\frac{1}{\varepsilon}\right)$.
  Forward $\langle l_t, x_t \rangle + \langle Z_t, x_t \rangle$ as input to $\mathcal{A}$.

Perturbation

Bandit Feedback: Analysis

Reduction to Non-private Setting

Require: Bandit Algorithm $\mathcal{A}$.
for $t = 1$ *to* T **do**
  Receive x_t from $\mathcal{A}$ and output x_t.
  Receive a loss value $\langle l_t, x_t \rangle$ from the adversary.
  Sample $Z_t \sim Lap\left(\frac{1}{\epsilon}\right)$.
  Forward $\langle l_t, x_t \rangle + \langle Z_t, x_t \rangle$ as input to $\mathcal{A}$.

Since bandit algorithms importance-sampling estimators to get unbiased estimates, adding a scalar directly is not a good idea.

$$\tilde{O}\left(\frac{\sqrt{T}}{\epsilon}\right) \text{ Regret}$$

Adding a vector perturbation permits one to *pretend* that $|l_t| \approx \frac{1}{\epsilon}$

Poster Today @ Gallery 68

Summary

$$\tilde{O}\left(\sqrt{T}+\frac{1}{\epsilon}\right)$$

Full-Information Setting

$$\tilde{O}\left(\frac{\sqrt{T}}{\epsilon}\right)$$

Bandit Feedback

Extension to Convex Loss functions?

$$\tilde{O}\left(\sqrt{T}+\frac{1}{\epsilon}\right)$$ Under bandit feedback?