Corruption-Robust Contextual Search through Density Updates

Chara Podimata (Harvard)

Joint work with Renato Paes Leme (Google NYC) and Jon Schneider (Google NYC)
“Realizable” = Exists hidden $\theta^* \in \mathbb{R}^d$ same across rounds. For rounds $t = 1, \ldots, T$: 

1. Nature chooses $d$-dimensional context $u_t \in \mathbb{R}^d$, s.t., $||u_t|| = 1$. 
2. Learner queries a scalar $y_t \in \mathbb{R}$. 
3. Nature replies $\text{sign}(y_t - \langle u_t, \theta^* \rangle) \in \{-1, +1\}$ (binary feedback). 
4. Learner incurs (but does not observe) loss $\ell(y_t, \langle u_t, \theta^* \rangle) \in [0, 1]$. 

Regret = total loss incurred – total loss for benchmark querying policy 

- $\epsilon$-ball loss: $\ell(y_t, y_t^*) = 1 \cdot 1\{|y_t - y_t^*| \geq \epsilon\}$ 
- symmetric loss: $\ell(y_t, y_t^*) = |y_t - y_t^*|$ 
- pricing loss: $\ell(y_t, y_t^*) = y_t^* - y_t \cdot 1\{y_t \leq y_t^*\}$

## Known Results

<table>
<thead>
<tr>
<th>Loss</th>
<th>$\ell(y_t, y_t^*)$</th>
<th>Lower Bound [R]</th>
<th>Upper Bound [R]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[Cohen, Lobel, Paes Leme, EC16/MS19], [Lobel, Paes Leme, Vladu, EC17/OR18], [Paes Leme, Schneider, FOCS18], [Liu, Paes Leme, Schneider, SODA21]
## Known Results

<table>
<thead>
<tr>
<th>Loss</th>
<th>$\ell(y_t, y_t^*)$</th>
<th>Lower Bound [R]</th>
<th>Upper Bound [R]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon$-ball</td>
<td>$1{</td>
<td>y_t - y_t^*</td>
<td>\geq \epsilon}$</td>
</tr>
</tbody>
</table>

[Cohen, Lobel, Paes Leme, EC16/MS19], [Lobel, Paes Leme, Vladu, EC17/OR18], [Paes Leme, Schneider, FOCS18], [Liu, Paes Leme, Schneider, SODA21]
## Known Results

<table>
<thead>
<tr>
<th>Loss</th>
<th>$\ell(y_t, y^*_t)$</th>
<th>Lower Bound [R]</th>
<th>Upper Bound [R]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon$-ball</td>
<td>$1{</td>
<td>y_t - y^*_t</td>
<td>\geq \varepsilon}$</td>
</tr>
<tr>
<td>symmetric</td>
<td>$</td>
<td>y_t - y^*_t</td>
<td>$</td>
</tr>
</tbody>
</table>

[Cohen, Lobel, Paes Leme, EC16/MS19], [Lobel, Paes Leme, Vladu, EC17/OR18], [Paes Leme, Schneider, FOCS18], [Liu, Paes Leme, Schneider, SODA21]
## Known Results

<table>
<thead>
<tr>
<th>Loss</th>
<th>$\ell(y_t, y_t^*)$</th>
<th>Lower Bound [R]</th>
<th>Upper Bound [R]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon$ -ball</td>
<td>$1{</td>
<td>y_t - y_t^*</td>
<td>\geq \varepsilon}$</td>
</tr>
<tr>
<td>symmetric</td>
<td>$</td>
<td>y_t - y_t^*</td>
<td>$</td>
</tr>
<tr>
<td>pricing</td>
<td>$y_t^* - y_t \cdot 1{y_t \leq y_t^*}$</td>
<td>$\Omega(d \log \log T)$</td>
<td>$O(d \log \log T + d \log d)$ [LLS21]</td>
</tr>
</tbody>
</table>

[Cohen, Lobel, Paes Leme, EC16/MS19]. [Lobel, Paes Leme, Vladu, EC17/OR18], [Paes Leme, Schneider, FOCS18], [Liu, Paes Leme, Schneider, SODA21]
Contextual Search with Adversarial Noise

Exists hidden $\theta^* \in \mathbb{R}^d$ same across rounds.

For rounds $t = 1, \ldots, T$:

1. Nature chooses $d$-dimensional context $u_t \in \mathbb{R}^d$, s.t. $||u_t|| = 1$.

2. Learner queries a scalar $y_t \in \mathbb{R}$.

3. Nature replies $\sigma_t = \text{sign}(y_t - y_t^*) \in \{-1, +1\}$ (binary feedback).

4. Learner incurs (but does not observe) loss $\ell(y_t, y_t^*) \in [0,1]$. 

$y_t^* = \langle u_t, \theta^* \rangle + z_t$

$z_t \in [0,1]$ : adaptively and adversarially chosen
Contextual Search with Adversarial Noise

Exists hidden $\theta^* \in \mathbb{R}^d$ same across rounds.

For rounds $t = 1, \ldots, T$:

1. Nature chooses $d$-dimensional context $u_t \in \mathbb{R}^d$, s.t., $||u_t|| = 1$.

2. Learner queries a scalar $y_t \in \mathbb{R}$.

3. Nature replies $\sigma_t = \text{sign}(y_t - y_t^*) \in \{-1, +1\}$ (binary feedback).

4. Learner incurs (but does not observe) loss $\ell(y_t, y_t^*) \in [0,1]$.

for $z_t \in \{0, 1\}$:

- $C_0 = \#\{z_t = 1\}$
- $\text{Regret} = O(C_0 d^3 \log^3 T)$ for pricing, symmetric loss
- $\text{Regret} = O(C_0 d^3 \log^3 1/\varepsilon)$ for $\varepsilon$-ball loss

[Krishnamurthy, Lykouris, P., Schapire, STOC21]

Runtime $\text{poly}(d, \log T)^{\text{poly}(\log T)}$
Corruption-robust contextual search algorithms with nearly optimal regret rates for symmetric & $\varepsilon$–ball loss

<table>
<thead>
<tr>
<th>Loss</th>
<th>Lower Bound [R]</th>
<th>Upper Bound [R]</th>
<th>Upper Bound [C] [KLP, STOC21]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon$–ball</td>
<td>$\Omega(d \log 1/\varepsilon)$</td>
<td>$O(d \log 1/\varepsilon)$</td>
<td>$O(C_0 d^3 \log^3 1/\varepsilon)$</td>
</tr>
<tr>
<td>symmetric</td>
<td>$\Omega(d)$</td>
<td>$O(d \log d)$</td>
<td>$O(C_0 d^3 \log^3 T)$</td>
</tr>
</tbody>
</table>
Main Result

\[ \varepsilon \text{-ball loss: } \text{Regret} = \mathcal{O}(C_0 + d \log 1/\varepsilon) \]

symmetric loss: polytime, \( \text{Regret} = \mathcal{O}(C_1 + d \log T) \), where \( C_1 = \sum_t |z_t| \)

<table>
<thead>
<tr>
<th>Loss</th>
<th>Lower Bound [R]</th>
<th>Upper Bound [R]</th>
<th>Upper Bound [C] [KLPS, STOC21]</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \varepsilon ) -ball</td>
<td>( \Omega(d \log 1/\varepsilon) )</td>
<td>( O(d \log 1/\varepsilon) )</td>
<td>( O(C_0 d^3 \log^3 1/\varepsilon) )</td>
</tr>
<tr>
<td>symmetric</td>
<td>( \Omega(d) )</td>
<td>( O(d \log d) )</td>
<td>( O(C_0 d^3 \log^3 T) )</td>
</tr>
</tbody>
</table>

known
Traditional Approach for Contextual Search Algorithms

For rounds \( t = 1, \ldots, T \):

- Maintain active **knowledge set** with feasible values for \( \theta^* \).
- Learner chooses \( y_t \) to make enough "progress" (e.g., \( y_t = (u_t, \text{centroid of knowledge set}) \)).
- Eliminate **inconsistent side** of knowledge set.

 Aggressively introducing cuts \( \rightarrow \) **fast, logarithmic bounds**
Our Fundamentally Different Approach

Overview of the Approach

• Maintain probability density function $f(\cdot)$ over all possible values of $\theta^*$.

• Density at point $x = \text{extent}$ to which $x$ is consistent with $\theta^*$.

→ Never remove values from consideration, just shift its “weight”.

→ Higher weight to more probable values.

Seemingly more ”forgiving” approach → faster bounds for corruption-robust
Algorithm for $\varepsilon$ – Ball Loss

$\varepsilon$ – Window Median Algorithm

Initialize $f_1(x)$: uniform over $B(0,1)$.

For rounds $t = 1,\ldots,T$:

• Observe $u_t$ and query $y_t = \varepsilon - \text{window - median}(f_t)$
Algorithm for $\varepsilon$ – Ball Loss

**$\varepsilon$ – Window Median Algorithm**

Initialize $f_1(x)$: uniform over $B(0,1)$.

For rounds $t = 1, \ldots, T$:
- Observe $u_t$ and query $y_t = \varepsilon – \text{window – median}(f_t)$
- Update density:
  $f_{t+1}(x) = \begin{cases} 
  \frac{3}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (u_t, x) - y_t \geq \varepsilon \\
  f_t(x), & \text{if } -\varepsilon/2 \leq \sigma_t \cdot (u_t, x) - y_t \leq \varepsilon/2 \\
  \frac{1}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (u_t, x) - y_t \leq -\varepsilon/2 
\end{cases}$
Algorithm for $\varepsilon$ – Ball Loss

$\varepsilon$ – Window Median Algorithm

Initialize $f_1(x)$: uniform over $B(0,1)$.

For rounds $t = 1, ..., T$:

- Observe $u_t$ and query $y_t = \varepsilon – window – median(f_t)$
- Update density:

$$f_{t+1}(x) = \begin{cases} 
\frac{3}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \geq \frac{\varepsilon}{2} \\
1 \cdot f_t(x), & \text{if } -\frac{\varepsilon}{2} \leq \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq \frac{\varepsilon}{2} \\
\frac{1}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq -\frac{\varepsilon}{2} 
\end{cases}$$
Algorithm for $\varepsilon$ – Ball Loss

$\varepsilon$ – Window Median Algorithm

Initialize $f_1(x)$: uniform over $B(0,1)$.

For rounds $t = 1, \ldots, T$:

- Observe $u_t$ and query $y_t = \varepsilon – \text{window median}(f_t)$
- Update density:

$$f_{t+1}(x) = \begin{cases} 
\frac{3}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \geq \frac{\varepsilon}{2} \\
1 \cdot f_t(x), & \text{if } -\frac{\varepsilon}{2} \leq \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq \frac{\varepsilon}{2} \\
\frac{1}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq -\frac{\varepsilon}{2}
\end{cases}$$

Main Result

- $\varepsilon$ –ball loss: $\text{Regret} = O(C_0 + d \log 1/\varepsilon)$

This can happen only $C_0$ times!
Algorithm for $\varepsilon$ – Ball Loss

$\varepsilon$ – Window Median Algorithm

Initialize $f_1(x)$: uniform over $B(0,1)$.

For rounds $t = 1, ..., T$:

- Observe $u_t$ and query $y_t = \varepsilon – \text{window} – \text{median}(f_t)$
- Update density:

$$f_{t+1}(x) = \begin{cases} 
\frac{3}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \geq \frac{\varepsilon}{2} \\
1 \cdot f_t(x), & \text{if } -\frac{\varepsilon}{2} \leq \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq \frac{\varepsilon}{2} \\
\frac{1}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq -\frac{\varepsilon}{2}
\end{cases}$$

Proof Idea

1. Given updates above, $f_t(\cdot)$ is always a density.

2. Potential $\Phi_t = \int_{B(\theta^*, \varepsilon/2)} f_t(x) dx$:
   - (weakly) increases in uncorrupted rounds
Algorithm for \( \varepsilon - \) Ball Loss

\textbf{\( \varepsilon - \) Window Median Algorithm}

Initialize \( f_1(x) \): uniform over \( B(0,1) \).

For rounds \( t = 1, \ldots, T \):

• Observe \( u_t \) and query \( y_t = \varepsilon - \text{window - median}(f_t) \)

• Update density:

\[
 f_{t+1}(x) = \begin{cases} 
 \frac{3}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \geq \frac{\varepsilon}{2} \\
 1 \cdot f_t(x), & \text{if } -\frac{\varepsilon}{2} \leq \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq \frac{\varepsilon}{2} \\
 \frac{1}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq -\frac{\varepsilon}{2}
\end{cases}
\]

\textbf{Proof Idea}

1. Given updates above, \( f_t(\cdot) \) is always a density.

2. Potential \( \Phi_t = \int_{B(\theta^*, \varepsilon/2)} f_t(x) dx \):

   • (weakly) increases in uncorrupted rounds
   • decreases by \( 1/2 \) in corrupted ones (\( C_0 \) in total)
Algorithm for $\epsilon$ – Ball Loss

$\epsilon$ – Window Median Algorithm

Initialize $f_1(x)$: uniform over $B(0,1)$.

For rounds $t = 1, \ldots, T$:

- Observe $u_t$ and query $y_t = \epsilon - \text{window\_median}(f_t)$
- Update density:

$$f_{t+1}(x) = \begin{cases} 
\frac{3}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \geq \frac{\epsilon}{2} \\
1 \cdot f_t(x), & \text{if } -\frac{\epsilon}{2} \leq \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq \frac{\epsilon}{2} \\
\frac{1}{2} \cdot f_t(x), & \text{if } \sigma_t \cdot (\langle u_t, x \rangle - y_t) \leq -\frac{\epsilon}{2} 
\end{cases}$$

Main Result

- $\epsilon$ – ball loss: $\text{Regret} = O(C_0 + d \log 1/\epsilon)$
- symmetric loss: $\text{Regret} = O(C_0 + d \log T)$

This can happen only $C_0$ times!

Runtime $\approx O(T^d \poly(d,T))$
Efficient Algorithm for Symmetric Loss

Main Result

Polytime algorithm with $\text{Regret} = O(C_1 + d \log T)$ for symmetric loss, where $C_1 = \sum_t |z_t|$.

Idea

- Maintain “structured” $f_t(\cdot)$, such that it always is a log-concave density.
- Query centroid of distribution: $y_t = c g_t = \int x f_t(x) dx$.
- Update: $f_{t+1}(x) = f_t(x) \cdot \left(1 + \frac{1}{3} \sigma_t \langle u_t, x - c g_t \rangle\right)$
- Finer control over corruptions, as density changes proportionally to how close to $c g_t$ a point $x$ is (rather than constant update based on $\sigma_t$).

$C_1 < C_0 = \sum_t 1\{z_t \neq 0\}$

efficient sampling ([Applegate & Kannan, STOC91])
Main Result

Corruption-robust contextual search algorithms with rates:

- $\varepsilon$-ball loss: $\text{Regret} = O(C_0 + d \log 1/\varepsilon)$
- symmetric loss: $\text{Regret} = O(C_1 + d \log T)$, where $C_1 = \sum_t |z_t|$ & polytime

Open Questions

1. Variant of distribution-based algorithms for pricing loss.

2. Algorithms with $\text{Regret} = O(C_1 + d \log d)$ for symmetric loss.

3. Polytime algorithm for $\varepsilon$-ball loss.
Thank You!