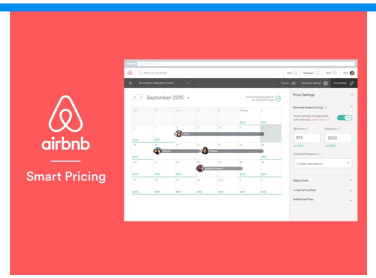
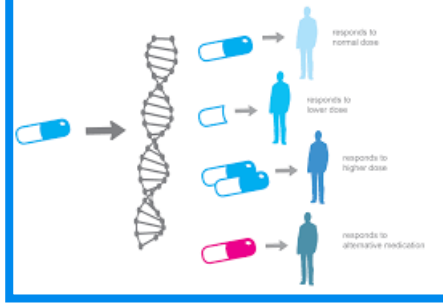
Corruption-Robust Contextual Search through Density Updates

Chara Podimata (Harvard)

Joint work with Renato Paes Leme (Google NYC) and Jon Schneider (Google NYC)

Contextual Search Realizable Version





"Realizable" = Exists hidden $\theta^\star \in \mathbb{R}^d$ 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 $\mathrm{sign}(y_t - \langle u_t, \theta^\star \rangle) \in \{-1, +1\}$ (binary feedback).
4. Learner incurs **(but does not observe)** loss $\boxed{\ell(y_t, \langle u_t, \theta^\star \rangle) \in [0,1].}$

- ε –ball loss: $\ell(y_t, y_t^\star) = 1 \cdot 1\{|y_t - y_t^\star| \geq \varepsilon\}$
- symmetric loss: $\ell(y_t, y_t^\star) = |y_t - y_t^\star|$
- pricing loss: $\ell(y_t, y_t^\star) = y_t^\star - y_t \cdot 1\{y_t \leq y_t^\star\}$

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

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

Contextual Search Realizable Version

Known Results

Loss	$\ell(y_t, y_t^\star)$	Lower Bound [R]	Upper Bound [R]

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

Contextual Search Realizable Version

Known Results

Loss	$\ell(y_t, y_t^\star)$	Lower Bound [R]	Upper Bound [R]
ε –ball	$1\{\lvert y_t - y_t^\star \rvert \geq \varepsilon\}$	$\Omega(d \log 1/\varepsilon)$	$O(d \log 1/\varepsilon)$ [LLV17]

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

Contextual Search Realizable Version

Known Results

Loss	$\ell(y_t, y_t^\star)$	Lower Bound [R]	Upper Bound [R]
ε –ball	$1\{\lvert y_t - y_t^\star\rvert \geq \varepsilon\}$	$\Omega(d \log 1/\varepsilon)$	$O(d \log 1/\varepsilon)$ [LLV17]
symmetric	$\lvert y_t - y_t^\star\rvert$	$\Omega(d)$	$O(d \log d)$ [LLS21]

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

Contextual Search Realizable Version

Known Results

Loss	$\ell(y_t, y_t^\star)$	Lower Bound [R]	Upper Bound [R]
ε –ball	$1\{\lvert y_t - y_t^\star\rvert \geq \varepsilon\}$	$\Omega(d \log 1/\varepsilon)$	$O(d \log 1/\varepsilon)$ [LLV17]
symmetric	$\lvert y_t - y_t^\star\rvert$	$\Omega(d)$	$O(d \log d)$ [LLS21]
pricing	$y_t^\star - y_t \cdot 1\{y_t \leq y_t^\star\}$	$\Omega(d \log\log T)$	$O(d \log\log T + d \log d)$ [LLS21]

[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 $\theta^\star \in \mathbb{R}^d$ 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 $\sigma_t = \mathrm{sign}(y_t - y_t^\star) \in \{-1, +1\}$ (binary feedback).
4. Learner incurs **(but does not observe)** loss $\ell(y_t, y_t^\star) \in [0,1]$.

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

$z_t \in [0,1]$: adaptively and adversarially chosen

Contextual Search with Adversarial Noise

Exists hidden $\theta^\star \in \mathbb{R}^d$ 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 $\sigma_t = \text{sign}(y_t - y_t^\star) \in \{-1, +1\}$ (binary feedback).
4. Learner incurs **(but does not observe)** loss $\ell(y_t, y_t^\star) \in [0,1]$.

$y_t^\star = \langle u_t, \theta^\star \rangle + z_t$
$z_t \in [-1,1]$: adaptively and adversarially chosen

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

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

$C_0 = \#\{z_t = 1\}$

- $Regret = O(C_0\, d^3 \log^3 T)$ for pricing, symmetric loss
- $Regret = O(C_0\, d^3 \log^3 1/\varepsilon)$ for ε −ball loss

Runtime
$poly\,(d, \log T)^{poly(\log T)}$

Main Question

Corruption-robust contextual search algorithms with

nearly optimal regret rates for symmetric & ε −ball loss

Loss	Lower Bound [R]	Upper Bound [R]	Upper Bound [C] [KLPS, STOC21]
ε −ball	$\Omega(d \log 1/\varepsilon)$	$O(d \log 1/\varepsilon)$	$O(C_0\, d^3 \log^3 1/\varepsilon)$
symmetric	$\Omega(d)$	$O(d \log d)$	$O(C_0\, d^3 \log^3 T)$

known

Main Result

ε –ball loss: $Regret = O(C_0 + d \log 1/\varepsilon)$

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

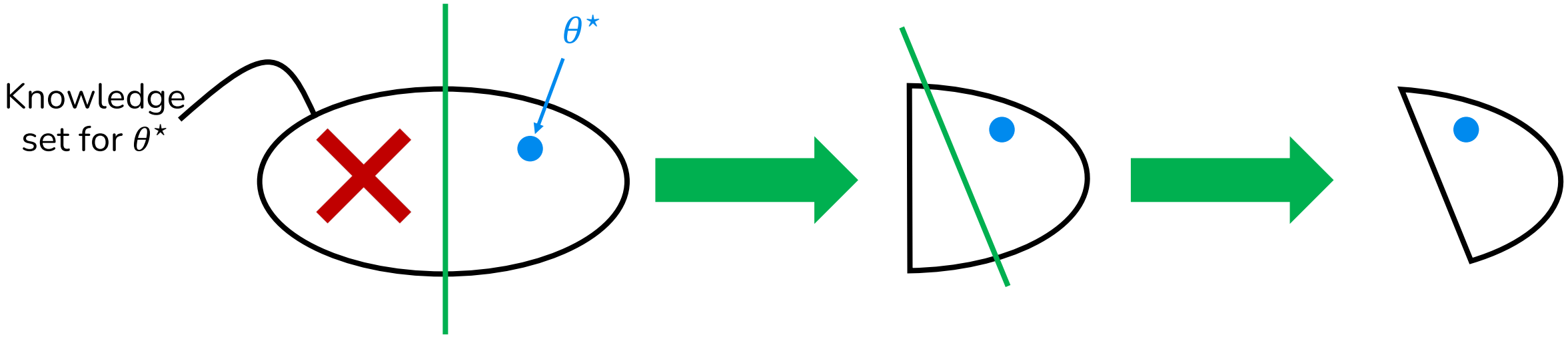
Loss	Lower Bound [R]	Upper Bound [R]	Upper Bound [C] [KLPS, STOC21]
ε –ball	$\Omega(d \log 1/\varepsilon)$	$O(d \log 1/\varepsilon)$	$O(C_0\, d^3 \log^3 1/\varepsilon)$
symmetric	$\Omega(d)$	$O(d \log d)$	$O(C_0\, d^3 \log^3 T)$

known

Traditional Approach for Contextual Search Algorithms

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

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



Aggressively introducing cuts → **fast**, **logarithmic** bounds

Our Fundamentally Different Approach

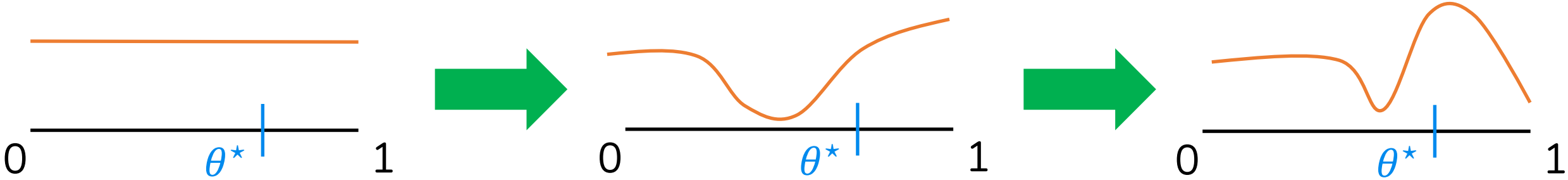
Overview of the Approach

- Maintain **probability density function** $f(\cdot)$ over all possible values of $\theta^\star$.
- **Density** at point x = **extent** to which x is **consistent** with $\theta^\star$.

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

→ Higher weight to more probable values.

Seemingly more "forgiving" approach → faster bounds for corruption-robust



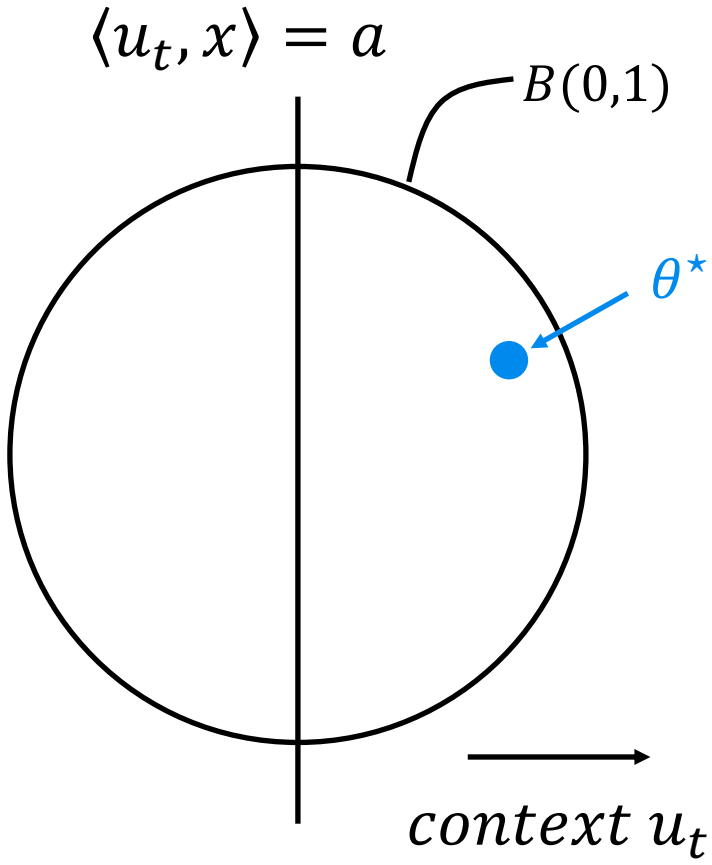
Algorithm for ε – Ball Loss

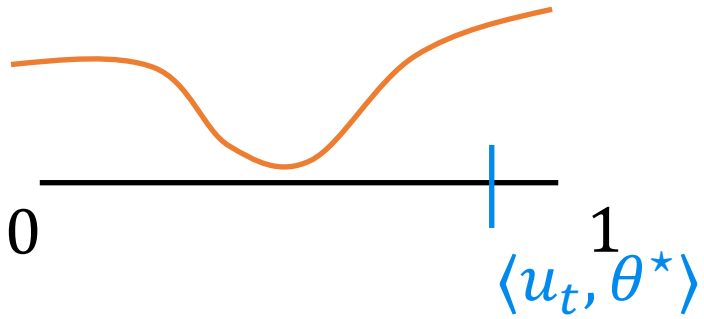
$\boldsymbol{\varepsilon}$ – Window Median Algorithm

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

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

- Observe u_t and query $y_t = \varepsilon - window - median(f_t)$





Algorithm for ε – Ball Loss

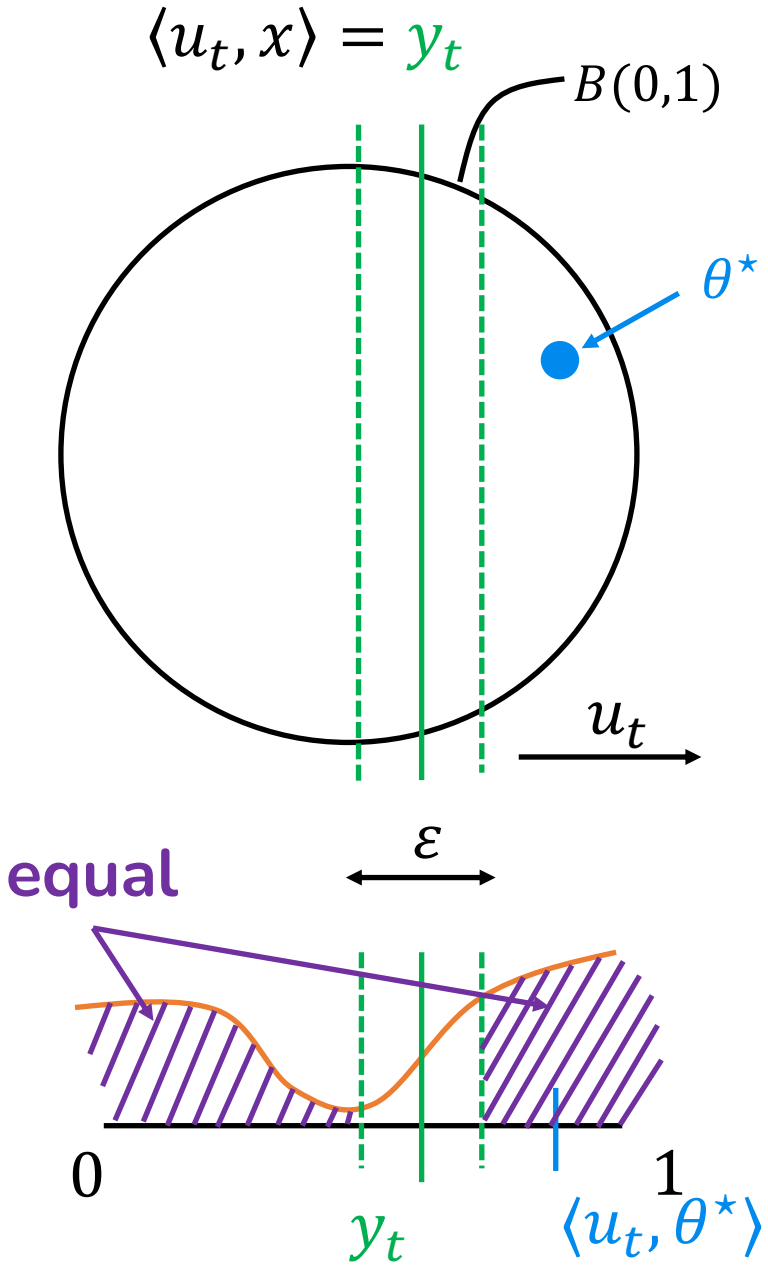
ε – Window Median Algorithm

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

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

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

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



Algorithm for ε – Ball Loss

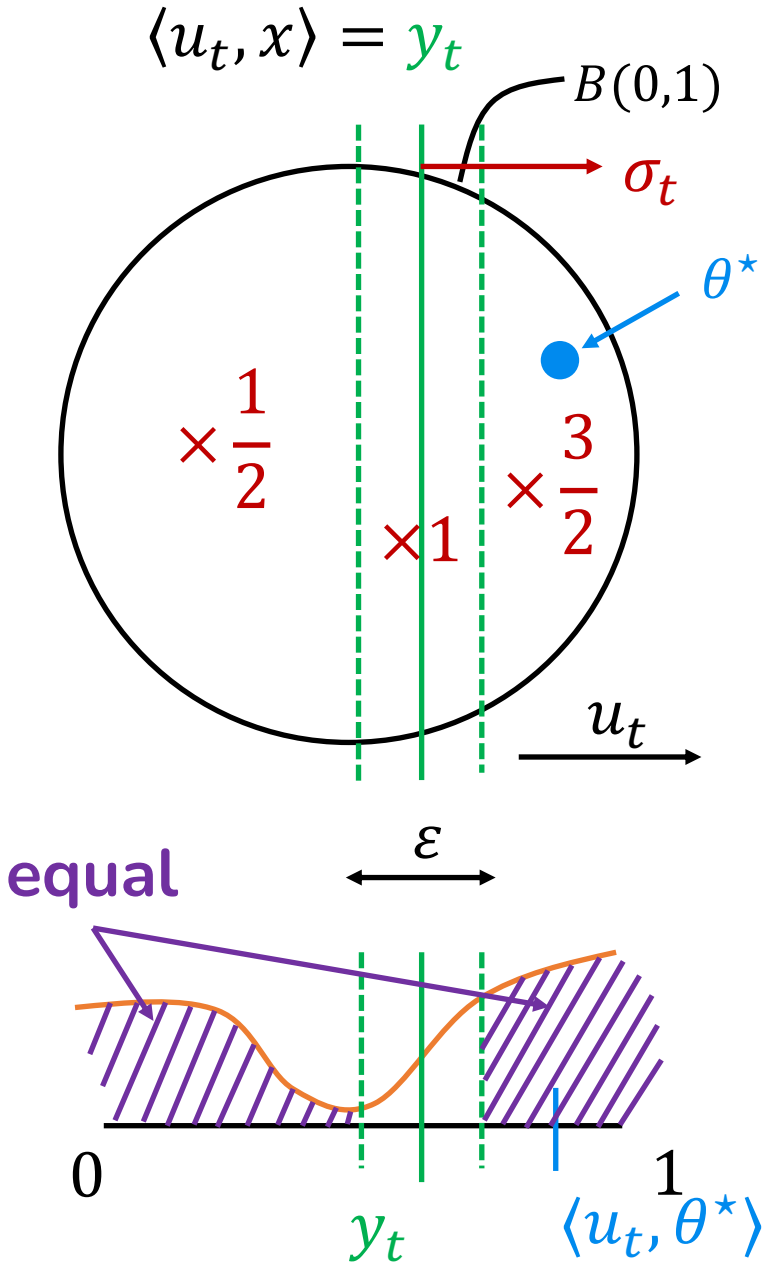
ε – Window Median Algorithm

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

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

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

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



Algorithm for ε – Ball Loss

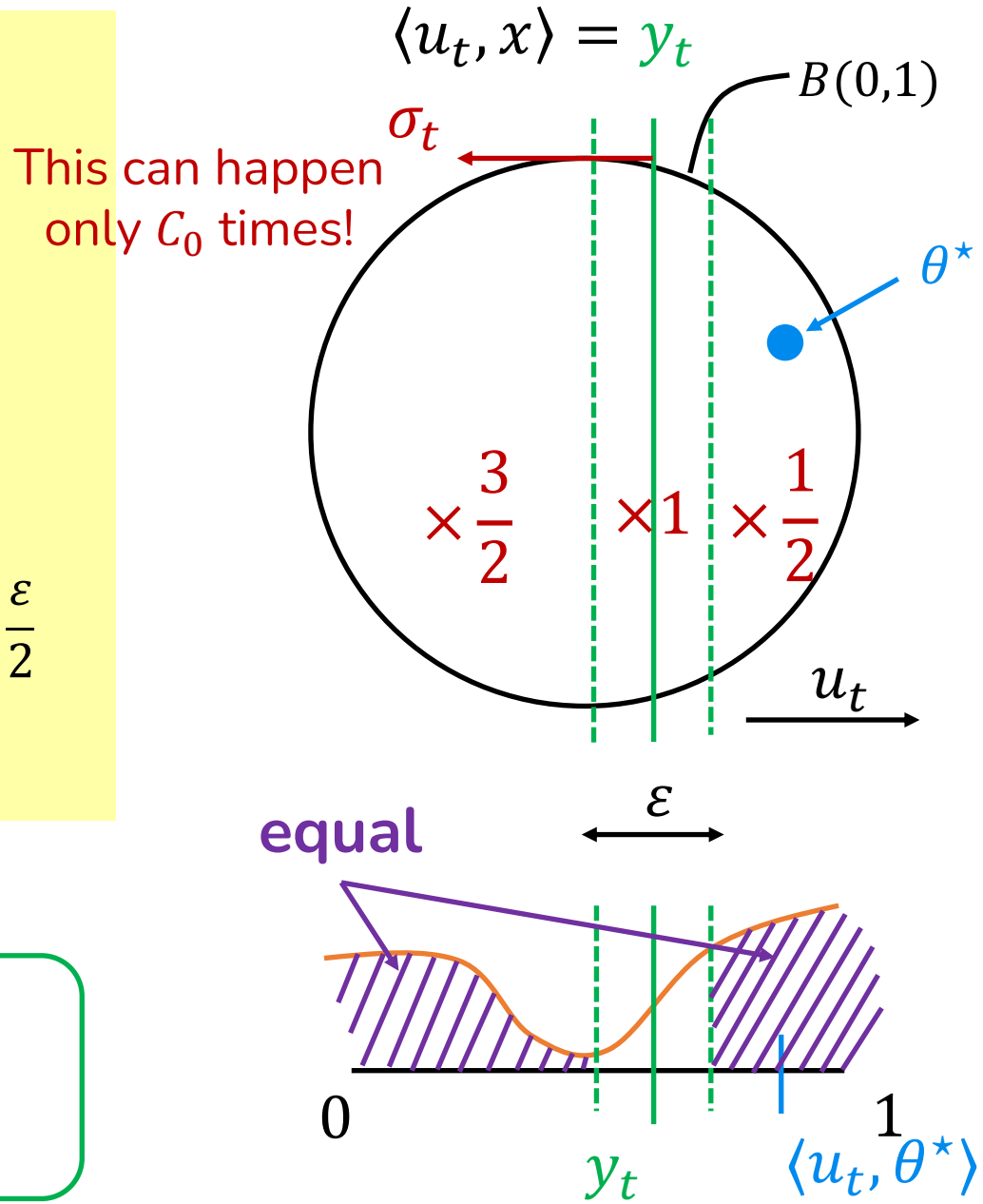
ε – Window Median Algorithm

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

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

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

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



Main Result

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

Algorithm for ε – Ball Loss

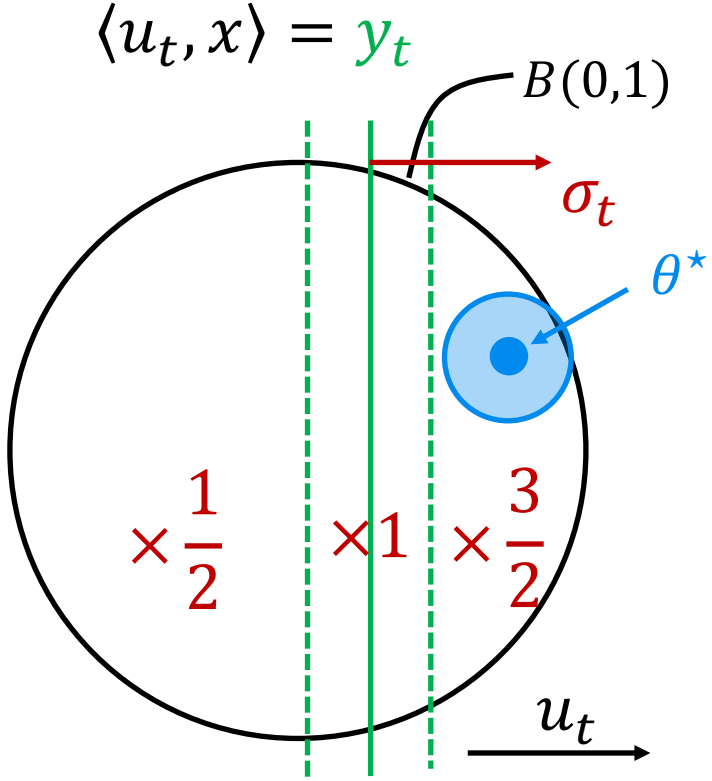
ε – Window Median Algorithm

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

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

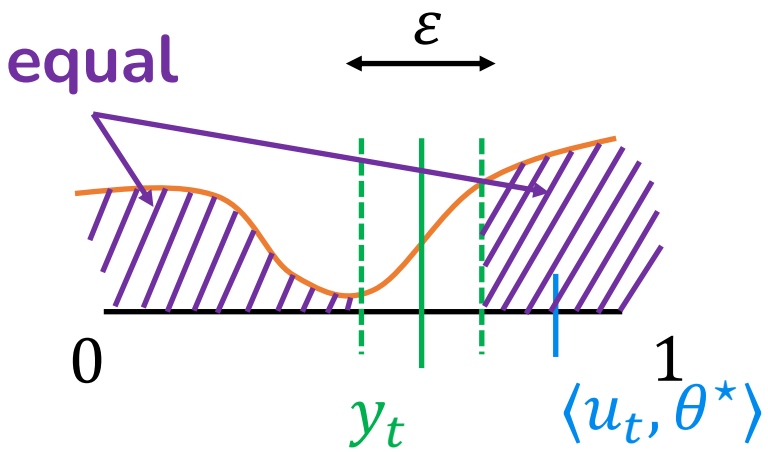
- Observe u_t and query $y_t = \varepsilon - window - median(f_t)$
- Update density:

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



Proof Idea

1. Given updates above, $f_t(\cdot)$ is **always a density**.
2. Potential $\Phi_t = \int_{B(\theta^\star, \varepsilon/2)} f_t(x)dx$:
   - **(weakly) increases** in uncorrupted rounds



Algorithm for ε – Ball Loss

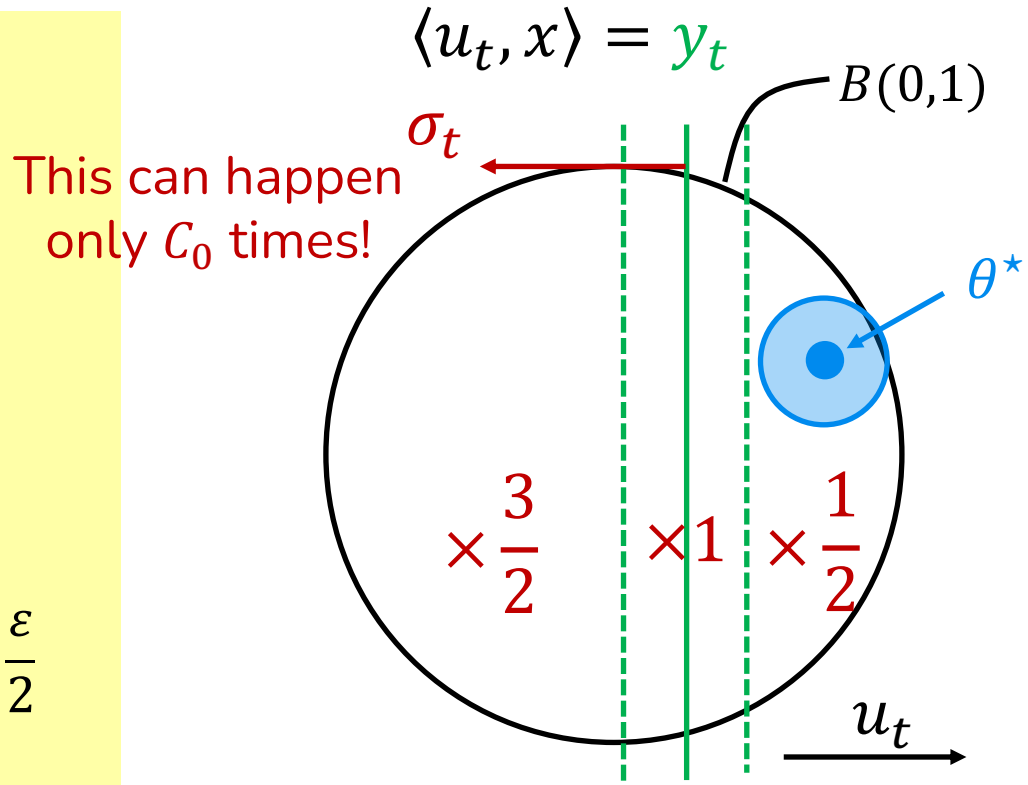
ε – Window Median Algorithm

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

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

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

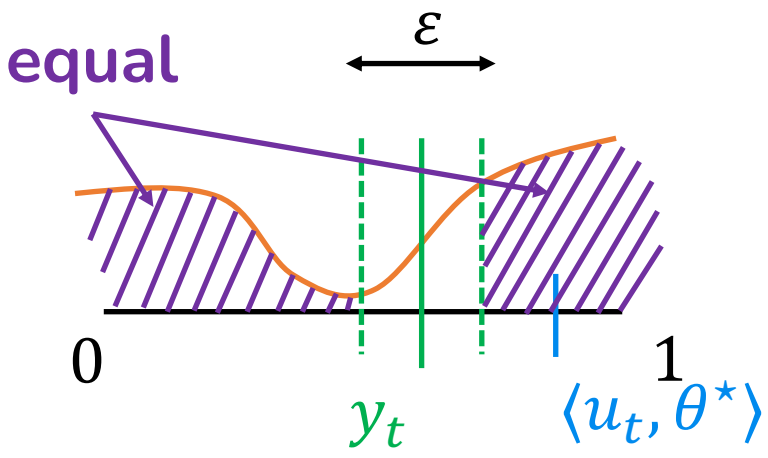
$$f_{t+1}(x) = \begin{cases} \frac{3}{2} \cdot f_t(x), & if\ \sigma_t \cdot (\langle u_t, x\rangle - y_t) \geq \frac{\varepsilon}{2} \\ 1 \cdot f_t(x), & if\ -\frac{\varepsilon}{2} \leq \sigma_t \cdot (\langle u_t, x\rangle - y_t) \leq \frac{\varepsilon}{2} \\ \frac{1}{2} \cdot f_t(x), & if\ \sigma_t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



Proof Idea

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

2. Potential $\Phi_t = \int_{B(\theta^\star, \varepsilon/2)} f_t(x)dx$:
   - **(weakly) increases** in uncorrupted rounds
   - **decreases by $1/2$** in corrupted ones (**C_0 in total**)



Algorithm for ε – Ball Loss

ε – Window Median Algorithm

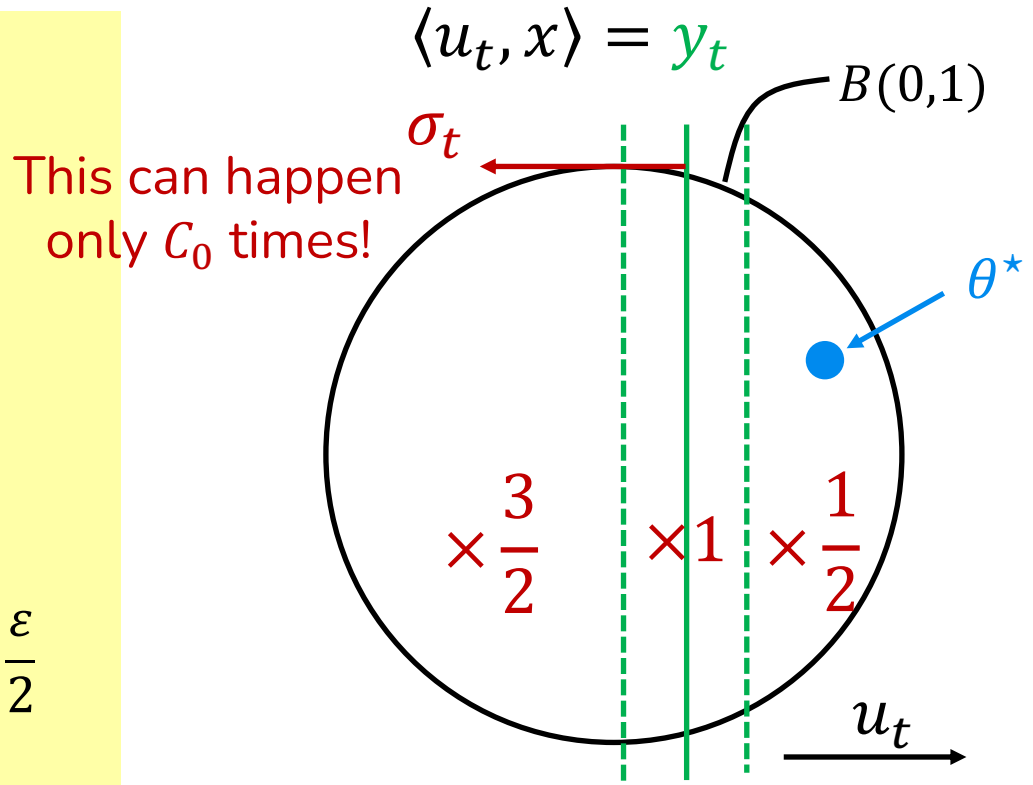
Initialize $f_1(x)$: uniform over $B(0,1)$.

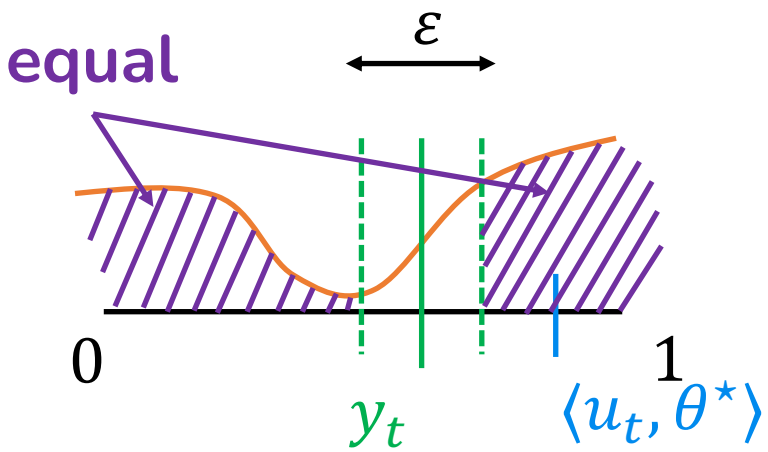
For rounds $t = 1, \dots, T$:

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

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

Runtime $\approx$ $O(T^d\, poly(d, T))$





Main Result

- ε –ball loss: $Regret = O(C_0 + d \log 1/\varepsilon)$
- symmetric loss: $Regret = O(C_0 + d \log T)$

Efficient Algorithm for Symmetric Loss

Main Result

Polytime algorithm with $Regret = O(C_1 + d \log T)$ for symmetric loss,

where $C_1 = \sum_t |z_t|$.

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

Idea

efficient sampling
([Applegate & Kannan, STOC91])

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

Main Result

Corruption-robust contextual search algorithms with rates:

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

Open Questions

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

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

3. **Polytime** algorithm for **ε –ball loss**.

Thank You!