Competing in the Dark: An Efficient Algorithm for Bandit Linear Optimization
Abernethy, Hazan, Rakhlin 2008

Akshay Balsubramani

Feb. 14, 2011
Outline

1. Preliminaries
   - Bandit Linear Optimization
   - Related Work
   - Goals

2. The Algorithm
   - Intuition
   - Notation and Concepts
   - Statement
   - Proof Ideas

3. Summary
Outline

1 Preliminaries
   • Bandit Linear Optimization
   • Related Work
   • Goals

2 The Algorithm
   • Intuition
   • Notation and Concepts
   • Statement
   • Proof Ideas

3 Summary
Bandit Linear Optimization
Combination of Two Ideas

- **Bandit Framework**
  - Partial-information setting, cannot deduce best action in hindsight for any timestep with certainty
  - Only observe one’s own loss each time

- **Online Linear Optimization**
  - Subset of Online Convex Optimization
  - Adversary ("nature") can only choose linear functions
Here we work in $n$ dimensions

- $K$ is a convex (also compact, closed) subset of $\mathbb{R}^n$
- For each time step $t = 1, \ldots, T$:
  - Algorithm chooses $\mathbf{y}_t \in K$
  - Adversary obliviously chooses $\mathbf{f}_t \in \mathbb{R}^n$
  - Algorithm observes loss $\mathbf{f}_t^\top \mathbf{y}_t$

- Regret:
  $$R_T = \sum_{t=1}^{T} \mathbf{f}_t^\top \mathbf{y}_t - \min_{\mathbf{u} \in K} \sum_{t=1}^{T} \mathbf{f}_t^\top \mathbf{u}$$
Outline

1. Preliminaries
   - Bandit Linear Optimization
   - Related Work
   - Goals

2. The Algorithm
   - Intuition
   - Notation and Concepts
   - Statement
   - Proof Ideas

3. Summary
Previous papers

- Non-stochastic multi-armed bandit (MAB)
- Online gradient descent with one-point estimate
- MAB generalization to arbitrary infinite sets of "arms"
- Progression of ideas here, which motivate new algorithm
Non-Stochastic Multi-Armed Bandit
Review of Exp3 Algorithm

Full treatment in Auer et. al. 2001: Exp3 algorithm and its variants. Exp3 is as follows: Given: \( \eta > 0; \gamma \in (0, 1] \)

Initialize \( \text{Hedge}(\eta) \)

\[
\text{for } t = 1, \ldots, T:
\text{Get distribution } \tilde{\rho}(t) \text{ from } \text{Hedge}(\eta)
\text{Choose } i_t \text{ from the distribution } \tilde{\rho}'_t = (1 - \gamma)\tilde{\rho}(t) + \frac{\gamma}{K}
\text{Observe } x_{i_t}(t) \in [0, 1]
\hat{x}_j(t) = \frac{x_{i_t}(t)}{\tilde{\rho}'_{i_t}(t)} \text{ if } j = i_t \text{ and } 0 \text{ otherwise}
\text{Input the vector } \hat{x}(t) \text{ to } \text{Hedge}(\eta) \text{ again}
Non-Stochastic Multi-Armed Bandit

Context

- Special case of BLO on convex set $K$
- Here $K = \text{probability simplex}$
- Exponential weights full-information algorithm (Hedge) used
- Notable points:
  - Estimate $\hat{x}_i(t)$ of the actual full-information function
  - Mix uniform distribution into probability updates to explore
Online Gradient Descent "Without a Gradient"

- Flaxman et. al. 2005 - presented last time
- Uses online gradient descent as full-information algorithm, not Hedge
- Adds random "perturbation" to OGD output, before calling full-information OGD again
- Main contribution - gradient estimation reduced to sampling
Dani et. al. 2006

- $O(\sqrt{T})$ regret bound, like full-information setting
- Generalizes Exp3
  - Uses Hedge as full-information algorithm
  - $K$ is arbitrary, not simplex
  - So clever estimation techniques needed
- But *computationally inefficient* distribution over arbitrary set of $\vec{x}$'s
Outline

1. Preliminaries
   - Bandit Linear Optimization
   - Related Work
   - Goals

2. The Algorithm
   - Intuition
   - Notation and Concepts
   - Statement
   - Proof Ideas

3. Summary
Goals Attained in this Paper

- Optimal $O(\sqrt{T})$ bound (approximately) on expected regret
- *Efficient* algorithm implementation, polynomial time/space
- $K$ is any convex set
- All these are attained by this algorithm
Online Primal-Dual (OPD) Algorithm

**Recap**

- Works for general online convex optimization
- Uses Bregman divergence with respect to function $\mathcal{R}(\vec{x})$
- Natural choice of full-information algorithm for us
  - $K$ just a convex set, not restricted
  - Avoids inefficiency of Hedge-type reductions
Online Primal-Dual Algorithm
Follow the Regularized Leader

- For us, equivalent to *Follow the Regularized Leader (FTRL)*

\[ \tilde{x}_{t+1} = \arg \min_{\tilde{x} \in K} \left( \eta \sum_{i=1}^{t} \tilde{f}_i^{\top} \tilde{x} + \mathcal{R}(\tilde{x}) \right) \]

- What happens when \( \mathcal{R} = 0 \)?
- What happens when \( \mathcal{R}(\tilde{x}) = \sum_i (x_i \log x_i - x_i) \)?
- Important: Degree of freedom in choosing \( \mathcal{R} \)
Using FTRL

- So we use FTRL (OPD) as full-information algorithm, but how to estimate $\tilde{f}_t$?
- Only have observed loss, many $\tilde{f}_t$’s could have caused this
- Must somehow sample in the neighborhood of $\tilde{x}$
- Similar to Flaxman, we can get gradient for OPD and proceed
Doing linear optimization → Optimal $\vec{x}^*$ is on boundary of $K$ (like linear programming)
Problems With Sampling

- Problem: Magnitude of the sampled function $\|\hat{f}_t\|$ is $\propto \frac{1}{r}$, where $r$ is distance from boundary
  - True for any case we will encounter; was true for Exp3 and Flaxman OGD
  - Operating close to boundary, $\hat{f}_t$ variance is very high
  - $\hat{f}_t$ can be quite far off what we intend, so high regret
- Can choose $\mathcal{R}$ to mitigate this
Barrier Functions

- Note FTRL: $\tilde{x}_{t+1} = \arg\min_{\tilde{x} \in K} \left( \eta \sum_{i=1}^{t} \tilde{f}_i^\top \tilde{x} + R(\tilde{x}) \right)$

- This is **constrained optimization**; harder than unconstrained
  - Can we remove constraint $\tilde{x} \in K$ by setting $R$?
    - Yes, theoretically; $R = 0$ if $\tilde{x} \in K$, $R = \infty$ if $\tilde{x} \not\in K$
  - Approximate this by **barrier function**
Choosing the Regularizer

- \( \mathcal{R} = \|\vec{x}\|^2 \) gives OGD
- But we can use any convex \( \mathcal{R} \)
- Need to be able to get really close to the boundary, but stop there
- For \( K = \) ball (and in 1-D), this is easy with \(-\log\) function, but for arbitrary convex set?
Scaling $K$

- Define the Minkowsky function w.r.t. a "pole" $\vec{x}_1$:

$$
\pi_{\vec{x}_1}(\vec{u}) = \inf\{ t \geq 0 : \vec{x}_1 + \frac{1}{t} (\vec{u} - \vec{x}_1) \in K \}
$$

- Scale $K$ by including all points in $K$ which are far enough away from the boundary:

$$
K_\delta = \{ \vec{u} : \pi_{\vec{x}_1}(\vec{u}) \leq \frac{1}{1 + \delta} \}$$
The $\theta$-Self-Concordant Barrier

A $\theta$-self-concordant barrier $\mathcal{R}$ in one dimension is a convex function such that $|f'''(x)| \leq 2(f''(x))^{3/2}$ and $|f'(x)| \leq \sqrt{\theta f''(x)}$.

Important in analysis when dealing with barrier functions

Approaches infinity sharply (parametrized by $\theta$) as boundary of $K$ is approached

Let the vector differential be defined as usual, e.g.

$$D\mathcal{R}(\vec{x})[\vec{h}] = \left. \frac{\partial}{\partial t_1} \right|_{t_1=0} \mathcal{R}(\vec{x} + t_1 \vec{h})$$

Generalize to vectors using this
Algorithm for Bandit OLO

Given $\eta > 0; \mathcal{R}(\vec{x}) : \mathbb{R}^n \mapsto \mathbb{R}$, a $\theta$-self-concordant barrier

$\vec{x}_1 = \arg \min_{\vec{x} \in K} \mathcal{R}(\vec{x})$

for $t = 1, \ldots, T$:

- Compute the eigenvalues $\{\lambda_1, \ldots, \lambda_n\}$ and corresponding unit eigenvectors $\{\vec{e}_1, \ldots, \vec{e}_n\}$ of $\nabla^2(\mathcal{R}(\vec{x}_t))$
- Choose $i_t$ uniformly at random from $\{1, \ldots, n\}$
- Choose $\epsilon_t = \pm 1$ with probability $\frac{1}{2}$ each

$\vec{y}_t = \vec{x}_t + \epsilon_t \lambda_{i_t}^{-1/2} \vec{e}_{i_t}$

Observe $\vec{f}_t^\top \vec{y}_t$

$\hat{f}_t = \vec{f}_t^\top \vec{y}_t \epsilon_t n \sqrt{\lambda_{i_t}} \vec{e}_{i_t}$

$\vec{x}_{t+1} = \arg \min_{\vec{x} \in K} \left( \eta \sum_{i=1}^{t} \hat{f}_i^\top \vec{x} + \mathcal{R}(\vec{x}) \right)$
Let $K$ be a convex set, $\mathcal{R}(\vec{x})$ and $K'$ be the Minkowsky-scaled set $K_{1/\sqrt{T}}$ as previously defined, $\theta$ be the self-concordance parameter, and all gains be $\in [-1, 1]$.

If $T > 8\theta \log T$ (and $\eta = \frac{\sqrt{\theta \log T}}{4n\sqrt{T}}$),

$$
\mathbb{E} \left( \sum_{t=1}^{T} \vec{f}_{t}^\top \vec{y}_{t} \right) - \min_{u \in K'} \mathbb{E} \left( \sum_{t=1}^{T} \vec{f}_{t}^\top \vec{u} \right) \leq 16n\sqrt{\theta T \log T}
$$
Main Result

- Good regret bound
- Requires bounded gains, intuitively like older algorithms
- Why can $K_{1/\sqrt{T}}$ be used instead of $K$?
  - Easy to show: $K_\delta$ is distance $\leq \delta$ from any point in $K$
  - Retreating very little from boundary of $K$, so resulting regret increase is small
  - Asymptotically optimal as $\frac{1}{\sqrt{T}} \to 0$
Outline

1. Preliminaries
   - Bandit Linear Optimization
   - Related Work
   - Goals

2. The Algorithm
   - Intuition
   - Notation and Concepts
   - Statement
   - Proof Ideas

3. Summary
Lemma 1: $\hat{f}_t$ is Good Enough

We have actual function $\vec{f}_t$ and our estimate of it, $\hat{f}_t$. Recall $\vec{x}_t$ is the output of FTRL and $\vec{y}_t$ is sampled around it. Suppose that $\forall t$, $\mathbb{E}(\hat{f}_t) = \vec{f}_t$ and $\mathbb{E}(\vec{y}_t) = \vec{x}_t$ (as in this algorithm). If we have

$$\sum_{t=1}^{T} \hat{f}_t^\top \vec{x}_t - \min_{u \in K'} \sum_{t=1}^{T} \hat{f}_t^\top u \leq C_T$$

then this implies

$$\mathbb{E} \left( \sum_{t=1}^{T} \hat{f}_t^\top \vec{y}_t \right) - \min_{u \in K'} \mathbb{E} \left( \sum_{t=1}^{T} \hat{f}_t^\top u \right) \leq C_T$$
Lemma 2: Regret as a Running Sum of Divergences

For all \( \vec{u} \in K \), FTRL regret is bounded by

\[
\eta \sum_{t=1}^{T} \hat{f}_t^\top (\vec{x}_t - \vec{u}) \leq D_{R}(\vec{u}, \vec{x}_1) + \sum_{t=1}^{T} D_{R}(\vec{x}_t, \vec{x}_{t+1})
\]

- \( D_{R}(\vec{x}_t, \vec{x}_{t+1}) \) must be controlled
- Previous algorithms: high \( \hat{f}_t \) variance made consecutive \( \vec{x}_t \)'s vary
- Now we can control this by choosing \( R \) to adapt to local space
Define the norm $\| \vec{h} \|_{\vec{x}} = (\vec{h}^{\top} \nabla^2 (R(\vec{x})) \vec{h})^{-1/2}$

The Dikin ellipsoid of radius $r$ centered at $\vec{x}$ is $W_r(\vec{x}) = \{ \vec{y} \in K : \| \vec{y} - \vec{x} \|_{\vec{x}} < r \}$

$W_1(\vec{x}_t)$ is approx. the amount of space available around $\vec{x}_t$, in all directions

Axes in directions of eigenvectors of $\nabla^2 (R(\vec{x}))$, inverse Hessian very low along these axes
Sample randomly along eigendirections of $W_1(\tilde{x}_t)$, basically out until ellipsoid boundary ($\lambda_j^{-1/2}$ far)

To second order, $D_R(\tilde{x}_t, \tilde{x}_{t+1}) = \eta^2 \hat{f}_t^\top (\nabla^2(R))^{-1} \hat{f}_t$

Low inverse Hessian kills high-variance $\hat{f}_t$

Sampling is valid and generally goes close to boundary; $W_1(\tilde{x}_t) \subseteq K$, but $W_2(\tilde{x}_t) \not\subseteq K$
All this means that $D_R(\bar{x}_t, \bar{x}_{t+1})$ is now nicely bounded ($\bar{x}_{t+1} \in W_{1/2}(\bar{x}_t)$)

By Lemmas 1 and 2, expected regret is bounded.
Summary

- Bandit online linear optimization can basically be solved as well as full-information - efficiently with good regret guarantees.
- Algorithm uses a reduction to full-information FTRL (equivalent to Online Primal-Dual).
- Self-concordant barrier regularizer and associated Dikin ellipsoid adapt to local geometry of convex set $K$, so sampling can be done with less variance, leading to lower regret.

Other work
- Only *expected* regret bounds; high-confidence bounds have not really improved on Exp3.P, so inefficient
- Also could do general convex functions, tighter bounds, adaptive adversary, etc.
Thank you!
Questions?
Small modification made to achieve faster runtime
Similar to old algorithm, except does not require us to solve convex opt. problem for
\[ \tilde{x}_{t+1} = \arg \min_{\tilde{x} \in K} \left( \eta \sum_{i=1}^{t} \hat{f}_i^T \tilde{x} + \mathcal{R}(\tilde{x}) \right) \]
Instead work with new variables \( \tilde{z}_t \) (estimates of \( \tilde{x}_t \)) instead of \( \tilde{x}_t \)
Use an iteration of Newton’s method to update modified FTRL
Can be proved: that \( \tilde{z}_t \) are "close" to corresponding \( \tilde{x}_t \)
As for \( K_{1/\sqrt{T}} \) intuition, if these points are close enough, then regret will be also