1 Introduction

The online bandit optimization setting is similar to the OCO setting introduced in the previous lectures. The difference is that at any given iteration \( t \), the player doesn’t have access to a gradient oracle on any arbitrary point \( x \in \mathcal{K} \). Instead, at any given time \( t \), the player first chooses an action \( x_t \), and then a convex cost function \( f_t \in \mathcal{F} \) is sampled. The player only gets to learn \( f_t(x_t) \). The general idea behind solving the problem is to construct an estimator \( g_t \) of the gradient \( \nabla f_t(x_t) \).

We will first introduce a general framework of BCO algorithm. Then we will see two different BCO algorithms that use two estimators and prove upper bounds on their regrets.

2 General BCO Framework

A general way of designing BCO algorithm consists of two stages. First is to construct a random variable \( g_t \) that satisfies \( E[g_t] = \nabla f_t(x_t) \). Second is to use this random variable \( g_t \) as an estimation of the real gradient and plug it in a OCO algorithm \( A \) that uses only the gradient of the cost function \( f_t \) at each iteration. This kind of reduction gives us a regret bound of the BCO algorithm:

**Theorem 1**

\[
E[\text{regret}_T(A)] = \sum_{t=1}^{T} E[f_t(x_t)] - f_t(x^*) \leq E[\text{regret}(g_1, g_2, \ldots, g_T)]
\]

where \( x^* \) is defined as,

\[
x^* = \arg\min_{x \in \mathcal{K}} \sum_{t=1}^{T} f_t(x_t)
\]

**Proof:**

First define function \( h_t \)

\[
h_t(x_t) = f_t(x_t) + \epsilon_t^T x \text{ where } \epsilon = g_t - \nabla f_t(x_t)
\]

Notice that \( E[\epsilon_t] = 0 \). And also we have

\[
\nabla h_t(x_t) = \nabla f_t(x_t) + g_t - \nabla f_t(x_t) = g_t
\]

By the regret bound of the OCO algorithm \( A \),

\[
\sum_{t=1}^{T} h_t(x_t) - h_t(x^*) \leq \text{regret}(g_1, g_2, \ldots, g_T)
\]
So the expected regret is,
\[
\sum_{t=1}^{T} \mathbb{E}[h_t(x_t)] - \mathbb{E}[h_t(x^*)] = \sum_{t=1}^{T} \mathbb{E}[f_t(x_t) + \epsilon_t^T x] - f_t(x^*)
\]
\[
= \sum_{t=1}^{T} \mathbb{E}[f_t(x_t)] + \mathbb{E}[\epsilon_t^T x] - f_t(x^*)
\]
\[
= \sum_{t=1}^{T} \mathbb{E}[f_t(x_t)] - f_t(x^*)
\]
\[
\leq \mathbb{E}[\text{regret}(g_1, g_2, \ldots, g_T)]
\]

\[\square\]

3 1D estimator

Let’s construct a naive gradient estimator in the one dimensional case. For \( x \in \mathbb{R} \) we have that \( f'(x) = \lim_{\delta \to 0} \frac{f(x+\delta)-f(x-\delta)}{2\delta} \). Let’s construct an unbiased estimator \( g(x) \) for \( f'(x) \).

\[
g(x) = \begin{cases} 
\frac{1}{2} f(x+\delta) & \text{w.p. } \frac{1}{2} \\
\frac{1}{2} f(x+\delta) & \text{w.p. } \frac{1}{2}
\end{cases}
\]

**Theorem 2** \( g(x) \) is an unbiased estimator of \( f'(x) \).

**Proof:**

\[
\mathbb{E}[g(x)] = \frac{1}{2\delta} (f(x+\delta) - f(x-\delta)) = f'(x)
\]

Therefore to \( g(x) \) is an unbiased estimator of \( f'(x) \).

4 Sphere sampling estimator

Now we generalize the 1D case to higher dimensions. Let’s first define unit ball and sphere in a \( n \) dimensional space.

\[
B_\delta = \{ x \mid \|x\| \leq \delta \}
\]
\[
S_\delta = \{ x \mid \|x\| = \delta \}
\]

For the cost function \( f(x) \), we define

\[
\hat{f}_\delta(x) = \mathbb{E}_{v \in B_1}[f(x + \delta v)]
\]

When \( f(x) \) is linear, the vectors \( v \) in the unit ball cancel out, and we are left with

\[
\hat{f}_\delta(x) = \mathbb{E}_{v \in B_1}[f(x + \delta v)] = f(x)
\]

Therefore if \( f(x) \) is linear, we would have,

\[
\nabla \hat{f}_\delta(x) = \nabla f(x)
\]

Now let’s construct an estimator of \( \nabla f_\delta(x) \).
Theorem 3

\[ \nabla \hat{f}_{\delta}(x) = \frac{n}{\delta} \mathbb{E}_{u \in S_1} [f(x + \delta u)u] \]

**Proof:** By definition,

\[ \hat{f}_{\delta}(x) = \frac{\int_{B_1} f(x + \delta v) dv}{\text{vol}(B_1)} \]

\[ \mathbb{E}_{u \in S_1} [f(x + \delta u)u] = \frac{\int_{S_1} f(x + \delta u)u du}{\text{vol}(S_1)} \]

By Stokes’ theorem,

\[ \nabla \int_{B_1} f(x + \delta v) dv = \frac{1}{\delta} \int_{S_1} f(x + \delta u)u du \]

\[ \nabla \hat{f}_{\delta}(x) = \frac{\nabla \int_{B_1} f(x + \delta v) dv}{\text{vol}(B_1)} \]

\[ = \frac{1}{\delta} \int_{S_1} f(x + \delta u)u du \]

\[ = \frac{1}{\delta} \frac{\text{vol}(S_1)}{\text{vol}(B_1)} \mathbb{E}[f(x + \delta u)u] \]

\[ = \frac{n}{\delta} \mathbb{E}[f(x + \delta u)u] \]

\[ g(x) = \frac{n}{\delta} f(x + \delta u)u \text{ is our sphere sampling estimator that satifies } \mathbb{E}[g(x)] = \nabla \hat{f}_{\delta}(x). \]

5 The FKM algorithm

We will first look at a BCO algorithm that uses the sphere sampling estimator and the online gradient descent algorithm. The first step in this algorithm is to shrink the convex set \( K \) to \( K_{\delta} = \{ x \mid \frac{1}{1-\delta} x \in K \} \).

Theorem 4 If \( K \) is convex, then all balls of radius \( \delta \) with their centers in \( K_{\delta} \) are in \( K \).

**Proof:**

\[ \forall y \in B_{\delta}, y = x + \delta v \text{ where } \|v\| \leq 1. \text{ Then for } x' = \frac{1}{1-\delta} x \text{ we have } y = (1 - \delta)x' + \delta v. \]

By the property of a convex set, \( y \in K \).

The FKM algorithm is the following
Algorithm 1 FKM algorithm

1. Initialize $x_1$ to be an arbitrary point in $K$.
2. For $t = 1, \ldots, T$ do
   1. Draw $u_t \in S_1$ uniformly at random.
   2. Play $y_t = x_t + \delta u_t$ and observe $f_t(y_t)$
   3. Generate $g_t = \frac{\delta}{\eta} f_t(y_t) u_t$
   4. Update $x_{t+1} = \prod_{K}(x_t - \eta g_t)$
3. End for

Theorem 5 If the convex set $K$ has a diameter $D$ and the cost functions $f \in F$ are $G$-Lipschitz, we have $E[\text{regret}] = O(n^4 D^2 G)$, when $\eta = \frac{D}{nT^4 G}$ and $\delta = \frac{1}{T^4}$.

Proof: Since $K$ has diameter $D$,
\[ \forall x, y \in K, \|x - y\| \leq D \]
$f$ is $G$-Lipschitz
\[ \forall x, y \in K, \|f(x) - f(y)\| \leq G\|x - y\| \]
From the two conditions we get that
\[ \|f(x + \delta u) - f(x)\| \leq G\|\delta u\| = \delta G \] (1)
Define $x_{\delta}^* = \prod_{K} x^*$, we have that
\[ \|f(x^*) - f(x_{\delta}^*)\| \leq G\|x^* - x_{\delta}^*\| \leq \delta GD \] (2)
\[ \mathbb{E}[\text{regret}] = \sum_{i=1}^{T} \mathbb{E}[f_t(y_t)] - f_t(x^*) \]

\[ = \sum_{i=1}^{T} \mathbb{E}[f_t(x_t)] - f_t(x^*) + \delta GT \text{ by (1)} \]

\[ = \sum_{i=1}^{T} \mathbb{E}[f_t(x_t)] - f_t(x^*_t) + \delta GT + \delta GT \text{ by (2)} \]

\[ = \sum_{i=1}^{T} \tilde{f}_t(x_t) - \tilde{f}_t(x^*_t) + \delta GT + 3\delta GT \text{ by (1)} \]

\[ \leq \mathbb{E}[\text{regret}_{OGD}(g_1, g_2, \ldots, g_T)] + \delta GT + 3\delta GT \text{ by Theorem 1} \]

\[ = \eta \sum_{i=1}^{T} \|g_t\|^2 + \frac{D^2}{\eta} + \delta GT + 3\delta GT \]

\[ = \eta \sum_{i=1}^{T} \left\| \frac{n}{\delta} f_t(y_t) u_t \right\|^2 + \frac{D^2}{\eta} + \delta GT + 3\delta GT \text{ assume the loss is bounded } \|f_t(y_t)\| \leq 1 \]

\[ \leq \eta T \frac{n^2}{\delta} + \frac{D^2}{\eta} + \delta GT + 3\delta GT \]

\[ = n DT^{\frac{3}{4}} + n DT^{\frac{3}{4}} + DGT^{\frac{3}{4}} + 3GT^{\frac{3}{4}} \]

\[ \leq O(nDGT^{\frac{3}{4}}) \]

\[ \square \]

6 Ellipsoidal sampling estimator

The sphere sampling estimator has some drawbacks. First we have to shrink the convex set to ensure that all the points on the balls are inside the set. Second the \( \delta \) term in the regret bound makes it difficult to improve the bound. So we will introduce another estimator for linear cost functions (\( f \in \mathcal{F} \) are linear) that avoids these problems. The construction is quite similar to the sphere estimator. Let’s define

\[ \tilde{f}(x) = \mathbb{E}_{B_1}[f(x + Av)] \]

**Theorem 6**

\[ \nabla \tilde{f}(x) = n \mathbb{E}_{u \in S_1}[f(x + Au)A^{-1}u] \]

**Proof:**

Define

\[ h(x) = f(Ax), \text{ and } \hat{h}(x) = \mathbb{E}_{v \in B_1}[h(x + v)] \]
\[ n \mathbb{E}_{u \in S_1} [f(x + Au)A^{-1}u] = nA^{-1} \mathbb{E}_{u \in S_1} [f(x + Au)u] = nA^{-1} \mathbb{E}_{u \in S_1} [h(A^{-1}x + u)u] = nA^{-1} \nabla \hat{h}(A^{-1}x) = A^{-1} A \nabla \hat{f}(x) = \nabla \hat{f}(x) \]

So we can construct an ellipsoid estimator \( g(x) = nf(x + Au)A^{-1}u \) satisfying \( \mathbb{E}_{u \in S_1} (g(x)) = \nabla \hat{f}(x) \).

### 7 Self-concordant barriers

Let \( K \) be a convex set in \( \mathbb{R}^n \) and a function \( R : \text{int}(K) \to \mathbb{R} \) is \( \nu \)-self-concordant if \( \forall x \in \text{int}(K) \) and \( h \in \mathbb{R}^n \),

1. \( R \) is three times differentiable, and
2. \( \| \nabla^3 R(x) \| \leq 2 \| \nabla^2 R(x) \| ^{3/2} \)
3. \( \| \nabla R(x)^T h \| \leq \nu \| h^T \nabla^2 R(x) h \| ^{1/2} \)

We will introduce Dikin’s ellipsoid and use it to give an alternative definition of \( \nu \)-self concordant function.

Define the Hessian of function \( R : \text{int}(K) \to \mathbb{R} \) to be \( H(x) = \nabla^2 R(x) \). Then the Dikin’s ellipsoid of \( R \) is,

\[ B_x(x, 1) = \{ y \mid (y - x)^T H(x) (y - x) \leq 1 \} \]

And we define the local norm of a vector \( u \) on \( x \) as

\[ \| u \|_x = \sqrt{u^T H(x) u} \]

Now we define a \( \nu \)-self concordant function \( R : \text{int}(K) \to \mathbb{R} \) as,

1. \( \forall x \in \text{int}(K), B_x(x, 1) \in K \)
2. \( \forall y \in B_x(x, 1), (1 - \| y - x \|_x)^2 H(x) \preccurlyeq H(y) \preccurlyeq \frac{H(y)}{(1 - \| y - x \|_x)^2} \)
3. \( \| \nabla R(x)^T h \| \leq \nu \| h^T \nabla^2 R(x) h \| ^{1/2} \)

**Theorem 7** \( F(x) = \sum_{i=1}^{m} - \log(b_i - A_i x) \) is a \( m \)-self concordant function on a \( n \)-dimensional polyhedron \( K = \{ x \mid Ax \leq b \} \), where \( A \) is a \( m \times n \) matrix and \( b \) is a \( m \times 1 \) vector.

**Proof:**
1. First prove that \( \forall y \in B_x(1), y \in K \)

\[
(y - x)^T H(x)(y - x) \leq 1
\]
\[
(y - x)^T \sum_{i=1}^{m} \frac{A_i^T A_i}{s_i(x)^2} (y - x) \leq 1
\]
\[
\sum_{i=1}^{m} \frac{\|A_i(y - x)\|^2}{s_i(x)^2} \leq 1
\]
\[
\sum_{i=1}^{m} \frac{(s_i(y) - s_i(x))^2}{s_i(x)^2} \leq 1
\]

This implies that \( \forall i, 0 \leq s_i(y) \leq 2s_i(x) \). Hence \( y \in K \).

2. Let \( \delta = \|y - x\|_x \), follow the same proof in part 1, we have

\[
\forall i, (1 - \delta)s_i(x) \leq s_i(y) \leq (1 + \delta)s_i(x)
\]
\[
\forall i, \frac{1}{1 + \delta}s_i(y) \leq s_i(x) \leq \frac{1}{1 - \delta}s_i(y)
\]

Therefore,

\[
\forall v, \frac{1}{(1 + \delta)^2} \times \sum_i \frac{\|A_i v\|^2}{s_i(x)^2} \leq \sum_i \frac{\|A_i v\|^2}{s_i(y)^2} \leq \frac{\sum_i \frac{\|A_i v\|^2}{s_i(x)^2}}{(1 - \delta)^2}
\]
\[
\rightarrow \frac{1}{(1 + \delta)^2} \times v^T H(x)v \leq v^T H(y)v \leq v^T \frac{H(y)}{(1 - \delta)^2} v
\]

since \( (1 - \delta)^2 \leq \frac{1}{(1 + \delta)^2} \)

\[
v^T (1 - \delta)^2 H(x)v \leq v^T H(y)v \leq v^T \frac{H(y)}{(1 - \delta)^2} v
\]
\[
\rightarrow (1 - \|y - x\|_x)^2 H(x) \preceq H(y) \preceq \frac{H(x)}{(1 - \|y - x\|_x)^2}
\]

3. Fix an \( h \), By Cauchy-Schwarz

\[
\sum_{i=1}^{m} \frac{A_i h}{s_i(x)} \leq \sqrt{m} \sqrt{\sum_{j=1}^{m} \frac{(A_j h)^2}{s_j(x)^2}}
\]

which implies

\[
\|\nabla R(x)h\| \leq m^{1/2} \|h^T \nabla^2 R(x)h\|^{1/2}
\]

\[\square\]
8 BLO and the AHR algorithm

In the special case of Bandit Linear Optimization, where the cost functions are linear, we can use the FTRL algorithm with an ellipsoid estimator and a self-concordant barrier function as the regularization function to construct a nearly optimal algorithm. The algorithm is the following:

Algorithm 2 AHR algorithm

\[ x_1 = \arg\min_{x \in \text{int}(\mathcal{K})} R(x) \]

\[ \text{for } t = 1, \ldots, T \text{ do} \]
\[ \quad \text{Draw } u_t \in S_1 \text{ uniformly at random.} \]
\[ \quad \text{Play } y_t = x_t + A_t u_t \text{ where } A_t = H(x)^{-\frac{1}{2}} \]
\[ \quad \text{Observe } f_t(y_t) \text{ and generate } g_t = nf_t(y_t)A_t^{-1}u_t \]
\[ \quad \text{Update } x_{t+1} = \arg\min_{x \in \mathcal{K}} (\sum_{s=1}^{t} \eta g_s^T x + R(x)) \]
\[ \text{end for} \]

We will prove an upper bound of the AHR algorithm’s regret next time.

References