CS292A Convex Optimization: Gradient Methods and Online Learning **Spring 2019**

Lecture 16: June 6

Lecturer: Yu-Xiang Wang *Scribes: Dheeraj Baby*

Note: *LaTeX template courtesy of UC Berkeley EECS dept.*

16.1 Bandit Convex Optimization

In the last lecture we saw the FKM algorithm for Bandit Convex Optimization(BCO). We continue with proof of its regret guarantees.

Theorem 16.1 ***(FKM Regret)*** *By choosing the stepsize $\eta = \frac{D}{nT^{3/4}}$ and $\delta = \frac{1}{T^{1/4}}$, the FKM algorithm with decision pool $K \subseteq \mathbb{R}^n$ with diameter D and loss functions f_t which are G Lipschitz achieves a regret*

$$\sum_{t=1}^{T} E[f_t(y_t)] - \sum_{t=1}^{T} f_t(u) \leq 9nDGT^{3/4},$$

where the expectation is taken over the randomness of FKM strategy.

Proof: To prove this result we need to take care of the following.

- Bound the artifacts from changing K to K_δ.
- Bound the error from approximating the original loss fucntion f_t by $\hat{f}_{t,\delta}$.
- Bound $\|g_t\|_2$ for controlling the regret of OGD.
- Optimally choose η and δ

If $u \in K$, define $u_\delta = \Pi_{K_\delta}(u)$. Recall the definition of $K_\delta = \{x | \frac{x}{1-\delta} \in K\}$. Thus we can bound the worst case distance between u and u_δ as $\|u_\delta - u\|_2 \leq \|(1-\delta)u - u\|_2 \leq \delta \|u\|_2 \leq \delta D$.

Since f_t is G lipschitz, we have $f_t(u_\delta) - f_t(u) \leq G\|u_\delta - u\|_2 \leq G\delta D$. Using this, let's bound the regret against a fixed expert in K by regret against fixed expert in K_δ.

$$E[\sum_{t=1}^{T} f_t(y_t) - f_t(u)] \leq \sum_{t=1}^{T} E[f_t(y_t)] - f_t(u_\delta) + \delta GTD,$$

where $y_t = x_t = \delta v$ with $v \sim S_1$, S_1 denotes a unit shell. Now consider

$$\sum_{t=1}^{T} E[f_t(y_t)] - f_t(u_\delta) \leq \sum_{t=1}^{T} f_t(x_t) - f_t(u_\delta) + \delta GTD, \text{ by lipschitzness and } D \geq 1 \text{ since } B_1 \subset K$$

Now let's bound the error between $\hat{f}_{t,\delta}$ and f_t.Recall that $\hat{f}_{t,\delta} = f(x_t + \delta v)$ where $v \sim B_1$, B_1 denotes a solid unit ball. By lipschitzness we have

$$\begin{aligned} f_t(x_t) - E[\hat{f}_{t,\delta}(x_t)] &= E[f_t(x_t) - f_t(x_t + \delta v)] \\ &\leq G\delta D. \end{aligned}$$

Thus,

$$\begin{aligned} \sum_{t=1}^{T} f_t(x_t) - f_t(u_\delta) &\leq \sum_{t=1}^{T} E[\hat{f}_{t,\delta}(x_t) - \hat{f}_{t,\delta}(u_\delta)] + 2\delta GTD \\ &\leq \text{Regret}_{\text{OGD}}[g_1, \ldots, g_T] + 2\delta GTD \\ &\leq \eta \sum_{t=1}^{T} \|g_t\|_2^2 + \frac{D^2}{\eta} + 2\delta GTD \end{aligned}$$

Minimizing the last expression wrt η first and then wrt δ yields the theorem.

■

This regret bound is weaker than the multi armed bandit setting due to its $T^{3/4}$ growth rate. Is it possible to do better?

- For linear loss functions f_t and $\hat{f}_{t,\delta}$ are same. So there is no approximation error incurred by smoothing
- Since FKM only makes its predictions from a smaller decision pool K_δ, maybe we can do better by exploring close to the boundary of K. But this would demand making δ very small that can hurt the regret bound. To get rid of this issue, instead of sampling from a unit ball of radius δ, can we consider sampling from an ellipse of proper axes dimensions?

In light of last point, let's generalize our unbiased gradient estimator rule.

Corollary 16.2 ***(ellipsoid sampling)*** *For any invertible matrix A, construct $\hat{f}(x) = E_{v \sim B_1}[f(x + Av)]$, then $\nabla \hat{f}(x) = nE[f(x + Au)A^{-1}u]$ where $x \in \mathbb{R}^n$.*

This give rise to the following FTRL based algorithm for BCO.

BCO will ellipsoid sampling

- for $t = 1, \ldots, T$
  - set $A_t = [\nabla^2 R(x_t)]^{-1/2}$
  - play $y_t = x_t + A_t u_t$ with $u_t \sim S_1$
  - set $g_t = n f_t(y_t) A_t^{-1} u_t$
  - update $x_{t+1} = \text{argmin } R(x) - \eta \sum_{i=1}^{t} g_i^T x$

Table 16.1: summary of various results for BCO under different assumptions on the loss functions.

	convex	linear	smoothness or strong convexity	smoothness and strong convexity
Upper bound	$T^{3/4}$	$T^{1/2}$	$T^{2/3}$	$T^{1/2}$
Lower bound	$T^{1/2}$	$T^{1/2}$	$T^{1/2}$	$T^{1/2}$

By looking at the above algorithm one sees that the domain constraint K has disappeared. Actually we will fold the domain constraint to $R(x)$ by choosing it to be a barrier function.

As a digression let's discuss an example of a barrier function. Suppose we want to smoothly approximate the convex set $I = x| -2 \leq x \leq 1$. i.e we wish to construct a function which assumes a small value in the interior of I and tends to $+\infty$ at the boundary of I. This can be accomplished by letting R to be the negative log barrier function as follows.

$$R(x) = -\log(x+2) - \log(1-x)$$

This helps us to convert a constrained optimization problem to unconstrained problem. An optimization algorithm then will be forced to explore within the strict interior of the domain.

Log barriers are not the only barrier functions that can be constructed. There exists many such functions. However for the purpose of analysis of algorithms such as one presented above that uses second order information involving Hessian, a property known as self concordance is highly useful.

Definition 16.3 ***(Self Concordance)*** *$R(x)$ is self concordant if $\forall x, R^{(3)}(x) \leq c[R^{(2)}(x)]^{3/2}$ for a constant c. Here $R^{(n)}$ denotes the n-th derivative.*

For multivariate functions, the definition should hold for any directional derivative. This property states that rate of change of Hessian can be bounded by the Hessian himself. The log barrier function satisfy this property.

Theorem 16.4 ***(BLO)*** *If loss functions are linear, BCO with ellipsoid sampling algorithm and log barrier regularizer yields a regret of $O(\sqrt{T \log T} n)$ for step size $\eta \asymp \frac{1}{n}\sqrt{\frac{\log T}{T}}$.*

- For 1D problem (Bubeck et al., 2015) shows that we can get $\sqrt{T}$ upper bound.
- (Hazan and Levy, 2014) For higher dimensions, we can get $\sqrt{T}\exp(n)$ using an $\exp(n)$ time algorithm.
- (Bubeck and Eldan, 2015) For higher there exists an algorithm with $n^{11}\sqrt{T}$ regret.

16.2 Dynamic Regret Minimization

So far we considered regret against a fixed expert in hindsight.

$$\text{Regret}(u) = \sum_{t=1}^{T} f_t(x_t) - f_t(u).$$

This type of regret is known as static regret. We can define a stronger notion of regret called *dynamic regret* as follows.

$$\text{Regret}(u_1, \dots, u_n) = \sum_{t=1}^{T} f_t(x_t) - f_t(u_t).$$

In this framework, we compare against a stronger non-stationary comparator sequence $u_1 \dots, u_n$. We aim to build policies that has sub-linear dynamic regret. However this won't always be feasible as demonstrated in the example below.

Example 1 *Consider the convex domain $K = [-1, 1]$. Let $f_t(x) = xw.p0.5$ and $-xw.p0.5$. Thus f_t is decided by flipping a fair coin. If the coin falls in heads, let the expert that minimizes the loss function be $u_t = -1$. Otherwise let $u_t = 1$.*

It follows that any algorithm that do not see f_t while making the prediction suffers a dynamic regret that is $\Omega(T)$.

Thus inorder to get sublinear dynamic regret, we must impose some regularity assumptions on the comparator sequence. We discuss some regularity conditions studied in the literature.

- (Zinkevich, 2003) if $\sum_{t=2}^{T} \|u_t - u_{t-1}\|_2 = o(T)$, then we can get a sub-linear dynamic regret.
- (Hall and Willett, 2013) $\sum_{t=2}^{T} \|u_t - \phi(t)\|_2 = o(T)$. Here $\phi(t)$ is a dynamical system that can closely mimic the strongest comparator sequence.
- (Chen et al., 2015) redefines the dynamic regret minimization objective as $\sum_{t=1}^{T} f_t(x_t) + \lambda \|x_t - x_{t-1}\| - \sum_{t=1}^{T} f_t(u_t) + \lambda \|u_t - u_{t-1}\|$. This imparts stability as we penalize both the comparator and online learner for switching their decisions.

Variational Measures on loss function sequence Consider the variational measure

$$V_T = \sum_{t=2}^{T} \|f_t - f_{t-1}\|_\infty. \tag{16.1}$$

If V_T is $o(T)$, and suppose we observe noisy gradients of a lipschitz loss function. i.e we observe the true gradient corrupted by mean zero iid sub-gaussian noise noise. Then (Besbes et al., 2015) proposes a restarting OGD algorithm that enjoys a dynamic regret of $O(T^{2/3} V_T^{1/3} + \sqrt{T})$. If we assume that f_t is strongly convex, restarting OGD achieves a dynamic regret of $O(T^{1/2}(1 + V_T^{1/2}))$. They provide matching lower bounds also under the setting of noisy gradient feedback.

Under the same setting as above if we assume that $f_1, \dots, f_T$ are quadratic and 1D, (Baby and Wang, 2019) proposes an optimal policy that achieves a dynamic regret of $O(T^{1/3} V_T^{2/3})$. This is accomplished by leveraging techniques from wavelet smoothing literature.

References

Dheeraj Baby and Yu-Xiang Wang. Online forecasting of total-variation-bounded sequences, 2019.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244, 2015.

Sebastien Bubeck, Ofer Dekel, Tomer Koren, and Yuval Peres. Bandit convex optimization: $\sqrt{T}$ regret in one dimension. In *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 266–278, Paris, France, 03–06 Jul 2015. PMLR. URL http://proceedings.mlr.press/v40/Bubeck15a.html.

Sbastien Bubeck and Ronen Eldan. Multi-scale exploration of convex functions and bandit convex optimization, 2015.

Niangjun Chen, Anish Agarwal, Adam Wierman, Siddharth Barman, and Lachlan L. H. Andrew. Online convex optimization using predictions, 2015.

Eric C. Hall and Rebecca M. Willett. Dynamical models and tracking regret in online convex programming, 2013.

Elad Hazan and Kfir Y. Levy. Bandit convex optimization: Towards tight bounds. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'14, pages 784–792, Cambridge, MA, USA, 2014. MIT Press. URL http://dl.acm.org/citation.cfm?id=2968826.2968914.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, pages 928–935. AAAI Press, 2003. ISBN 1-57735-189-4. URL http://dl.acm.org/citation.cfm?id=3041838.3041955.