Bandits for Online Optimization

Nicolò Cesa-Bianchi

Università degli Studi di Milano

The multiarmed bandit problem

. . .

K slot machines

- Each play of a slot machine (action) returns a payoff
- Design a strategy of repeated play to maximize cumulative payoff
- A classical problem in sequential design of experiments [Robbins, 1952]
- Motivating application: allocation of medical treatments
- Modern applications: web content adaptation, heuristic selection, routing, tree search

Key issues

- Partial feedback: payoff of each action changes over time, but only the payoff of the played action is observed at each time step
- Exploration/Exploitation dilemma: Focusing on most promising action (exploitation) may prevent the discovery of better actions (exploration)
- Payoff generation: What is a good generative model for payoffs?

Nonstochastic bandits

Sidestep the payoff modeling problem by avoiding any stochastic assumption on the mechanism generating payoffs

Online convex optimization with bandit feedback

Online version of gradient-free optimization

- Closed and convex action space $\mathcal{K} \subseteq \mathbb{R}^d$
- Hidden sequence $\ell_1, \ell_2 \dots$ of convex loss functions $\ell_t : \mathcal{K} \to \mathbb{R}_+$
- A paradigm for robust optimization in a changing environment

For each $t = 1, 2, \dots$

1. Pick action $X_t \in \mathcal{K}$
2. Observe value $\ell_t(X_t)$ of current loss function ℓ_t at X_t

Performance measures

Regret of sequence $X_1, X_2, \ldots$

$$R_T = \sum_{t=1}^{T} \ell_t(X_t) - \sum_{t=1}^{T} \ell_t(x_T^*) \qquad \text{where} \qquad x_T^* = \operatorname*{argmin}_{x \in \mathcal{K}} \sum_{t=1}^{T} \ell_t(x)$$

For all T, the total loss of action sequence $X_1, \ldots, X_T$ must be close to that of the best fixed action for any individual sequence $\ell_1, \ldots, \ell_T$ of convex loss functions

Goal

Assuming $\max_{t, x \in \mathcal{K}} \ell_t(x) \leqslant 1$, regret must grow sublinearly with time T

Online gradient descent

Pick $X_1 \in \mathcal{K}$ arbitrarily

For each $t = 1, 2, \dots$

1. Use $X_t \in \mathcal{K}$ and observe loss $\ell_t(X_t)$
2. Compute estimate $\widehat{g}_t$ of loss gradient $\nabla \ell_t(X_t)$
3. Gradient step $X'_{t+1} = X_t - \eta \widehat{g}_t$
4. Projection step $X_{t+1} = \operatorname*{argmin}_{x \in \mathcal{K}} \left\| x - X'_{t+1} \right\|$

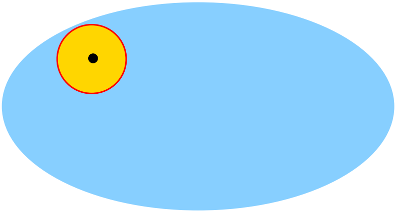
Point X_t must simultaneously:

- have small loss $\ell_t(X_t)$ (exploitation)
- lead to a good gradient estimate $\widehat{g}_t$ (exploration)

Gradient descent without a gradient

[Flaxman, Kalai and McMahan, 2004]

- Use a perturbed version of X_t: $\quad X_t + rU$
  (U is a random unit vector and $r > 0$)
- Gradient estimate $\quad \widehat{g}_t = \dfrac{d}{r}\ell_t(X_t + rU)U$
- Fact: If ℓ_t is differentiable, then
  $$\mathbb{E}\big[\widehat{g}_t\big] = \nabla\mathbb{E}\big[\ell_t(X_t + rB)\big]$$
  where B is a random vector in the unit sphere



$\widehat{g}_t$ estimates the gradient of a locally smoothed version of ℓ_t

Guarantees

Properties

- If ℓ_t is Lipschitz, then the smoothed version is a good approximation of ℓ_t
- Radius r of perturbation controls bias/variance trade-off

Regret of OGD for convex and Lipschitz loss sequences

$$\mathbb{E}\, R_T = \mathcal{O}\left(T^{3/4}\right)$$

Guarantees

Properties

- If ℓ_t is Lipschitz, then the smoothed version is a good approximation of ℓ_t
- Radius r of perturbation controls bias/variance trade-off

Regret of OGD for convex and Lipschitz loss sequences

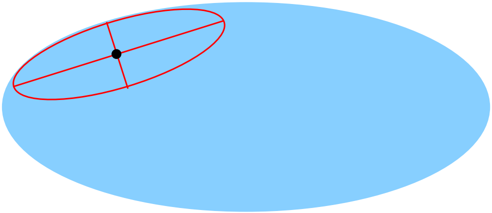
$$\mathbb{E}\, R_T = \mathcal{O}\left(T^{3/4}\right)$$

The linear case

- Losses are linear functions on $\mathcal{K}$, $\quad \ell_t(x) = \ell_t^\top x$
- Can we achieve a better rate?

Self-concordant functions [Abernethy, Hazan and Rakhlin, 2008]

- Fact: any convex closed set $\mathcal{K}$ admits a self-concordant function F (generally hard to find)
- Variance control through the Dikin ellipsoid $\nabla^2 F \subseteq \mathcal{K}$
- Loss estimate $\widehat{\ell}_t$ obtained via perturbed point $X_t \pm \mathbf{e}_i \sqrt{\lambda_i}$
  $\{\mathbf{e}_i, \lambda_i\}$ is a randomly drawn eigenvector-eigenvalue pair
- Run Online Mirror Descent regularized with a self-concordant function for $\mathcal{K}$



Regret for linear functions

$$R_T = \mathcal{O}\left(\sqrt{T \ln T}\right)$$

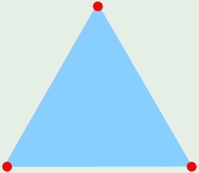
Online combinatorial optimization with bandit feedback

Setting

- Action space $\mathcal{S} \subseteq \{0,1\}^d$
- Linear loss functions $\ell_t(x) = \ell_t^\top x$
- Loss estimates $\widehat{\ell}_t$

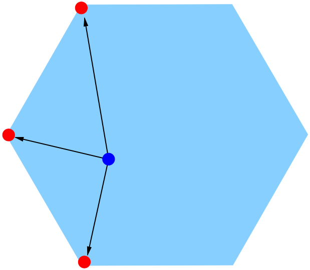
Example

K-armed bandits
$\mathcal{S} =$ corners of the simplex



Online Mirror Descent

1. Run Online Mirror Descent in $\mathcal{K}$ = convex hull of $\mathcal{S}$
2. Map current point $X_t \in \mathcal{K}$ to distribution over $\mathcal{S}$



This guarantees $\boxed{\mathbb{E}\, R_T = \mathcal{O}\big(\sqrt{T \ln T}\big)}$ but…

Issues

At each step t of OMD we need to:

- Solve a convex program to compute next point in $\mathcal{K}$
- Solve t linear programs to compute a sparse distribution over $\mathcal{S}$ (via Frank-Wolfe algorithm)

Can we get $\sqrt{T}$ regret in online linear optimization using only a linear optimization oracle?

Follow the perturbed leader

For each $t = 1, 2, \dots$

Add random perturbation Z_t to loss estimates and pick action with lowest perturbed loss

$$X_{t+1} = \operatorname*{argmin}_{x \in \mathcal{S}} \sum_{s=1}^{t} \left(\widehat{\ell}_s + Z_t\right)^\top x$$

- Requires a single call to a linear optimization oracle at each step
- However, best known bandit regret bound is suboptimal
$$R_T = \mathcal{O}\left(T^{2/3}\right)$$
- Variance control through Z_t is harder than in OMD

Solution for a special case

The semi-bandit model

- Action space $\mathcal{S} \subseteq \{0,1\}^d$
- Linear loss functions $\ell_t(x) = \ell_t^\top x$
- Bandit feedback is $\ell_t^\top X_t$
- Semi-bandit feedback is $\{\ell_{i,t} \,:\, X_{i,t} = 1\}$

The stronger feedback allows to construct estimates $\widehat{\ell}_t$ with smaller variance

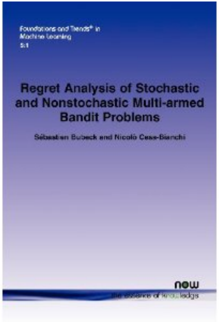
Regret of FPL with Laplace perturbations

$$\mathbb{E}\, R_T = \mathcal{O}\big(\sqrt{T}\big)$$

Conclusion

- In the convex case: optimal rate still unknown (between $T^{1/2}$ and $T^{3/4}$)
- In the linear case: optimal rate $T^{1/2}$ attained only via convex optimization
- In the combinatorial case: optimal rate $T^{1/2}$ attained via linear optimization, but using a stronger feedback model

If you want to know more. . .



S. Bubeck and N. Cesa-Bianchi (2012), “Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems”
Foundations and Trends in Machine Learning.