Online DR-Submodular Maximization: Minimizing Regret and Constraint Violation

Prasanna Raut*[1] , Omid Sadeghi*[2] , Maryam Fazel[2]
1. Mechanical Engineering Department, University of Washington
2. Electrical and Computer Engineering Department, University of Washington

Online Protocol

- A convex domain set $\mathcal{X} \subset \mathbb{R}_+^n$ and total available budget BT.
- At step $t \in [T]$, the player chooses $x_t \in \mathcal{X}$.
- Then, a monotone DR-submodular utility function $f_t: \mathcal{X} \to \mathbb{R}$ and a constraint vector p_t with $\mathbb{E}[p_t] = p$ are revealed, the player obtains the reward $f_t(x_t)$ and uses $< p_t, x_t >$ of her budget.

Goal: Maximize the overall obtained reward while ensuring the budget constraint is satisfied on average, i.e., the total budget violation $C_T = \frac{1}{T}\sum_{t=1}^{T} < p, x_t > - B$ should grow sub-linearly in T.

$$\underset{x_t \in \mathcal{X}}{\text{maximize}} \sum_{t=1}^{T} f_t(x_t), \qquad \text{subject to } \frac{1}{T}\sum_{t=1}^{T} < p, x_t > \leq B.$$

DR-Submodular Functions. A differentiable function $f: \mathcal{X} \to \mathbb{R}, \mathcal{X} \subset \mathbb{R}_+^n$, is called DR-submodular if:

$$x \succcurlyeq y \to \nabla f(x) \preccurlyeq \nabla f(y).$$

If f is twice differentiable, DR property is equivalent to the Hessian matrix being element-wise non-positive.

Regret metric. The $(1 - \frac{1}{e})$-regret is defined as:

$$R_T = \left(1 - \frac{1}{e}\right)\sum_{t=1}^{T} f_t(x^*) - \sum_{t=1}^{T} f_t(x_t),$$

where:

$$x^* = \text{argmax}_{x \in \{x \in \mathcal{X}: \mathbb{E}[< p_t, x >] \leq B]\}} \sum_{t=1}^{T} f_t(x).$$

Motivating Application: Online Ad Placement

- Your company produces some product and has a choice of n different websites where it can advertise under a fixed limited budget BT.
- At step $t \in [T]$, you choose a selection $x_t \in \mathcal{X} = \{x \in \mathbb{R}_+^n : 0 \preccurlyeq x \preccurlyeq 1\}$ and place your ads on the websites.
- Then, the cost for this assignment, $< p_t, x_t >$, is revealed where the components of p_t capture the traffic received by the ads on the n websites. Also, you get utility $f_t(x_t)$ which corresponds to the number of products sold on that website.

Our Algorithm

- The penalized formulation of the overall optimization problem can be written as:

$$\underset{x_t \in \mathcal{X}}{\text{maximize}} \sum_{t=1}^{T} f_t(x_t) - \frac{1}{2\delta\mu}\left(\sum_{t=1}^{T} < p, x_t > - BT\right)^2.$$

- Or equivalently as:

$$\max_{x_t \in \mathcal{X}} \min_{\lambda} \sum_{t=1}^{T} f_t(x_t) - \lambda\left(\sum_{t=1}^{T} < p, x_t > - BT\right) + \frac{\delta\mu}{2}\lambda^2.$$

- Thus, the per round Lagrangian can be written as:

$$\mathcal{L}_t(x, \lambda) = f_t(x) - \lambda(< p, x > - B) + \frac{\delta\mu}{2}\lambda^2.$$

Algorithm 1 Online Lagrangian Frank-Wolfe

Input: $\mathcal{X}$, T is the horizon, $\mu > 0$, $\delta > 0$, $\{\gamma_t\}_{t=1}^{T}$ and K.
Output: $\{x_t : 1 \leq t \leq T\}$.
Initialize K instances $\mathcal{E}_k$ $\forall k \in [K]$ of Online Gradient Ascent with step size μ for online maximization of linear functions over $\mathcal{X}$.
for $t = 1$ **to** T **do**
  Set $x_t = \sum_{k=1}^{K} v_t^{(k)}$.
  Let $\widehat{p}_t := \frac{1}{t-1}\sum_{s=1}^{t-1} p_s$ for $t > 1$.
  Let (I) $\widetilde{g}_t(\cdot) = \langle \widehat{p}_t, \cdot \rangle - B$ or (II) $\widetilde{g}_t(\cdot) = \langle \widehat{p}_t, \cdot \rangle - B - \gamma_t$.
  Set $\lambda_t = \frac{[\widetilde{g}_t(x_t)]_+}{\delta\mu}$ for $t > 1$ and 0 otherwise.
  Play x_t and observe $\mathcal{L}_t(x_t, \lambda_t) = f_t(x_t) - \lambda_t \widetilde{g}_t(x_t) + \frac{\delta\mu}{2}\lambda_t^2$.
  for $k = 1$ **to** K **do**
    Feedback $\langle v_t^{(k)}, \nabla_x \mathcal{L}_t(x_t^{(k)}, \lambda_t) \rangle$ as the payoff to $\mathcal{E}_k$.
  end for
end for

Theoretical Results

Theorem (Expectation Bounds): With parameters $\mu = \frac{R}{\beta\sqrt{T}}, K = \sqrt{T}, \delta = \beta^2$, Algorithm 1 with update rule (I) achieves $\mathcal{O}(T^{\frac{3}{4}})$ regret and $\mathcal{O}(T^{\frac{1}{2}})$ constraint violation bound in expectation.

Theorem (High Probability Bounds): For every $\epsilon > 0$, with parameters $\mu = \frac{R}{\beta\sqrt{T}}, K = \sqrt{T}, \delta = \beta^2$ & $\gamma_t = \sqrt{2G^2 \ln\left(\frac{2T}{\epsilon}\right)/t}$, Algorithm 1 with update rule (II) achieves $\mathcal{O}(T^{\frac{1}{2}})$ regret and constraint violation bound simultaneously with probability at least $1 - \epsilon$.

Numerical Experiments

Experiment 1: Online joke recommendation using the *Jester* dataset.

$$f_t(x) = r_t^T x + \sum_{i,j:i<j} \theta_{ij}^{(t)} x_i x_j \;\; \forall t \in [T],$$

where $0 \leq [r_t]_i \leq 10$ is the rating of user t for joke i in the dataset.

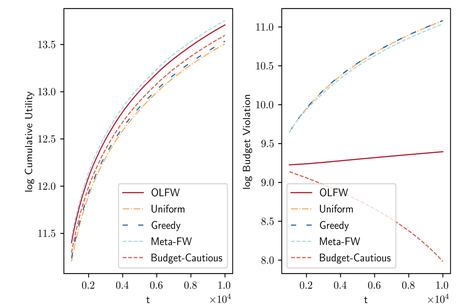

Figure 1. Cumulative utility $\sum_{t=1}^{T} f_t(x_t)$ and total budget violation C_T for experiment 1.

Experiment 2: Effect of parameter δ on the performance of the algorithm

$$f_t(x) = x^T Q_t x \;,$$

where Q_t is a random matrix whose elements are non-positive. δ is varied from 10^{-1} to 10^3.

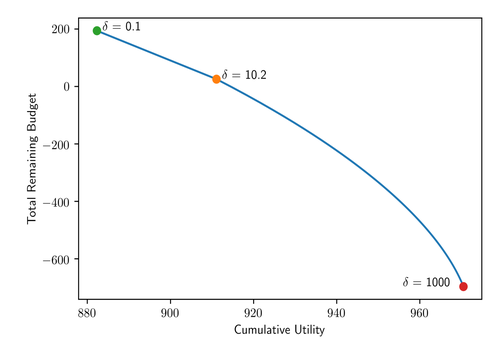

Figure 2. Cumulative utility $\sum_{t=1}^{T} f_t(x_t)$ and total remaining budget for experiment 2.

Acknowledgements

This work was supported in part by the following grants: NSF TRIPODS grant 1740551, DARPA Lagrange grant FA8650-18-2-7836, ONR MURI grant N0014-16-1-2710.

References

Jester dataset. http://eigentaste.berkeley.edu/dataset.