Optimization Methods in ML Spring 2019 - HW 3

Lecturer: Dan Garber

Due date: 21.7.2020

Guidelines:

- Please make an effort to type your answers on a computer
- you may submit in pairs
- you may consult with your fellow classmates ("high-level" discussions) - but you may not copy answers!
- programming is allowed in whatever environment you prefer
- submit to me via email with title: "Optimization Methods in ML - HW 3" (expect my acknowledgment) (include names and IDs in email body)
- feel free to contact me with questions

Question 1 (Follow-the-Leader). *Recall that Follow the Leader (FTL) strategy for online learning plays on each round t an action $\mathbf{x}_t \in \mathcal{K}$ such that $\mathbf{x}_t \in \arg\min_{\mathbf{x}\in\mathcal{K}} \sum_{i=1}^{t-1} f_i(\mathbf{x})$.*

1. *Design an online convex optimization setting, i.e., choose a set of actions $\mathcal{K} \subset \mathbb{R}^d$ - convex and compact (you can choose d as you would like) and a sequence of convex loss functions $f_1, \ldots, f_T$ (for all T large enough) such that FPL provably attains $\Omega(T)$ regret for this setting.*
2. *Prove that when all loss functions $f_1, \ldots, f_T$ are α-strongly convex over a convex and compact set $\mathcal{K} \subset \mathbb{R}^d$, then FTL guarantees $O(\log T)$ regret. Hint: consult with Subsection 3.3. in Elad's book to see results on logarithmic regret.*

Question 2 (OGD and adaptive regret). *Recall that the Online (sub)Gradient Descent guarantees $O(\sqrt{T})$ regret against the best fixed point in $\mathcal{K}$ in hindsight, in case all losses are convex. However, in certain settings, competing with the best fixed point $\mathbf{x}^*$ is not a strong-enough guarantee, since $\mathbf{x}^*$ itself might attain high overall loss. In such cases, it can be preferable to compete with a shifting benchmark, i.e., on different intervals of the sequence, compete with the best point with respect to the interval. This is often called adaptive regret.*
Prove that for any integer $k \in \{1, \ldots, T\}$ there exists a choice of a fixed step-size η, such that for any partition of $\{1, 2, \ldots, T\}$ into k disjoint intervals $I_1 =$

$\{t_1, \ldots, t_1'\}, \ldots, I_k = \{t_k, \ldots, t_k'\}$, *where for all* $i > 1$*:* $t_i = t_{i-1}' + 1$ *(note the intervals can be of different lengths), Online (sub)Gradient Descent produces a sequence* $\{\mathbf{x}_t\}_{t=1}^T$ *such that*

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_k^* \in \mathcal{K}} \sum_{i=1}^{k} \sum_{t=t_i}^{t_i'} f_t(\mathbf{x}_i^*) = O(\sqrt{kT})$$

Question 3. *Recall the FKM algorithm for bandit convex optimization (Subsection 6.4 in Elad's book). Prove that when all loss functions are* α*-strongly convex, there exist a choice for the parameters of the algorithm such that it guarantees* $\tilde{O}(T^{2/3})$ *regret, where* $\tilde{O}(\cdot)$ *hides poly-logarithmic terms. Hint: consult with Subsection 3.3. in Elad's book to see results on logarithmic regret.*

Question 4. *In this question you will experiment with online convex optimization algorithms for the online rebalancing portfolio selection (ORPS) problem (see end of Lecture 10 which appears on the moodle webpage). Recall that in the ORPS problem, on each trading round we use our entire wealth to buy stocks according to a certain distribution (portfolio), and at the end of the round, we sell all our assets, and then repeat the process on the next trading round.*

The loss on each trading round t *is* $f_t(\mathbf{x}) := -\log\left(\mathbf{x}^\top \mathbf{r}_t\right)$*, where* $\mathbf{x}$ *is a point in the unit simplex and* $\mathbf{r}_t$ *is the vector such that*

$$\mathbf{r}_t(i) = \frac{\textit{price of asset } i \textit{ at end of round } t}{\textit{price of asset } i \textit{ at beginning of round } t}.$$

Recall that the feasible set is the unit simplex, i.e., $\Delta := \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} \geq \mathbf{0},\ \sum_{i=1}^n \mathbf{x}_i = 1\}$*, where* n *is the number of stocks.*

Load the dataset in `http://www.cs.princeton.edu/~ehazan/pfdata/data_490_1000.mat` *(could be done via matlab, though using matlab for the experiment is not mandatory).*

The file contains a matrix named A *which contains the prices of 490 stocks on 1000 consecutive trading dates. For each asset (stock) in the dataset, add also its corresponding* ***short asset*** *- i.e., when buying a short asset on time* t*, you pay for it the price at* ***end of time t*** *and you gain the price at the* ***beginning of time t*** *(that is, you are selling an asset that you commit to buy later on).*

Apply the following online learning algorithms for online portfolio selection with the above data: 1) Online (sub)Gradient Descent, 2) Online Exponentiated Gradient (see Lecture 11) and 3) the Online Newton Step method (which was not covered in class), see Algorithm 9 on page 61 in `http://ocobook.cs.princeton.edu/OCObook.pdf`*. See Theorem 4.3 in the book for the regret bound for this algorithm. Also note that for ONS, the notation* $\Pi_{\mathcal{K}}^{\mathbf{A}_t}(\mathbf{x})$*, where* $\mathbf{A}_t$ *is a positive definite matrix, means* $\Pi_{\mathcal{K}}^{\mathbf{A}_t}(\mathbf{x}) = \arg\min_{\mathbf{z} \in \mathcal{K}} (\mathbf{z} - \mathbf{x})^\top \mathbf{A}_t (\mathbf{z} - \mathbf{x})$ *(i.e., it is the projection with respect to the metric induced by the matrix* $\mathbf{A}_t$*).*

Plot on a single figure the overall wealth of each algorithm vs. the number of trading rounds. Plot also (on the same figure), the wealth of the best fixed rebalancing portfolio in hindsight and the wealth of the best fixed stock (a single asset) in hindsight. The graph should be clear so the performance of the different benchmarks can be clearly seen.

Submit:

1. *Graphs (with the performance of the different algorithms clearly seen).*
2. *Code used for the experiments.*