Following the Perturbed Leader for Online Structured Learning

Alon Cohen ALONCOH@CSWEB.HAIFA.AC.IL
University of Haifa, University of Haifa, Dept. of Computer Science, 31905 Haifa, Israel

Tamir Hazan TAMIR@CS.HAIFA.AC.IL
University of Haifa, University of Haifa, Dept. of Computer Science, 31905 Haifa, Israel

Abstract

We investigate a new Follow the Perturbed Leader (FTPL) algorithm for online structured prediction problems. We show a regret bound which is comparable to the state of the art of FTPL algorithms and is comparable with the best possible regret in some cases. To better understand FTPL algorithms for online structured learning, we present a lower bound on the regret for a large and natural class of FTPL algorithms that use logconcave perturbations. We complete our investigation with an online shortest path experiment and empirically show that our algorithm is both statistically and computationally efficient.

1. Introduction

Learning from a constant flow of data is one of the central challenges of machine learning. Online learning require to sequentially decide on actions in ever-changing environments. Such environments include the stock market and ranking newly published movies. In past years, a variety of online learning algorithms have been devised, each such algorithm for a different setting. Among well-known settings are prediction with expert advice (Littlestone & Warmuth, 1994) and online convex optimization (Zinkevich, 2003). Despite their applicability, many sequential decision-making problems still remain open — one such problem is that of online structured learning.

Structured problems drive many of the recent applications in machine learning, such as road navigation, network routing, adaptive resource allocation, personalized content recommendation, online ad display and many more. Unfortunately, known algorithms for this setting are suboptimal in the following sense — they are either statistically efficient or computationally efficient, but not both.

In our work we investigate Follow the Perturbed Leader algorithms, whose advantage is their simplicity and computational efficiency. We present a new FTPL algorithm whose regret bound is comparable to the best known bound of FTPL algorithms for structured problems. We further show a lower bound that implies that most FTPL algorithms found in the literature are, in fact, suboptimal in terms of regret. Finally, we run an experiment evaluating our algorithm against the state of the art. We empirically show that our algorithm has similar regret and yet has a much faster runtime.

1.1. Related work

The problem of online learning has been studied in the last few decades across many different fields. Since the 1990s, however, it has become a prominent field of research alongside the Machine Learning community. This is attributed to, for example, works of Cover (1991); Blum (1998); Foster & Vohra (1999).

Perhaps the most well known setting of online learning is that of prediction with expert advice. Originally, this was the problem of predicting a binary sequence, being given the predictions of a number of experts at each round. This problem has been studied extensively in, for example, Littlestone & Warmuth (1994); Cesa-Bianchi et al. (1997) that have given an algorithm for the problem based on the multiplicative updates rule. This algorithm is known as Hedge, Exponential Weights (EXP) or Randomized Weighted Majority.

Hannan (1957); Kalai and Vempala (2002; 2005) have introduced another algorithmic scheme for online learning, now known as Follow the Perturbed Leader (FTPL). Since then, FTPL algorithms were the subject of many papers, including Rakhlin et al. (2012); Neu & Bartók (2013); Devroye et al. (2013); Abernethy et al. (2014); Van Erven et al. (2014). It is a known folklore result, that the Hedge al-

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37. Copyright 2015 by the author(s).

gorithm is equivalent to FTPL with Gumbel perturbations (Kuzmin & Warmuth, 2005).

In our paper, we consider an extension of the prediction with expert advice setting, known as online structured learning or online combinatorial optimization. In structured learning, the predictions of the learner are taken from a discrete set $\mathcal{X} \subseteq \{0,1\}^d$. We assume that $\mathcal{X}$ is endowed with an offline optimization oracle, that given a vector $z \in \mathbb{R}^d$, produces $x \in \arg\min_{x' \in \mathcal{X}} \langle z, x' \rangle$. We assume that querying the oracle is done in a computationally efficient manner. As is common, we denote k such that for all $x \in \mathcal{X}$ we have $\sum_{i=1}^d x_i \leq k$.

Originally, Takimoto & Warmuth (2003) introduced a special case of this setting — the online shortest path problem. Given some graph, this is the problem of sequentially predicting a shortest path between two fixed vertices, s and t. There, the authors have shown a reduction from the problem of shortest path to prediction with expert advice. Namely, that each path is an expert, and that an efficient implementation of the Hedge algorithm exists under this reduction. This algorithm has a regret bound of $O(\sqrt{kT \log |\mathcal{X}|})$, and here $\mathcal{X}$ is the set of all $s - t$ paths in the graph including ones that contain cycles and k is the maximum number of the edges in any such path.

Kalai and Vempala (2005) have shown an FTPL algorithm for this setting. Cesa-Bianchi & Lugosi (2006) (cf. Ex 5.12) have improved the analysis of that algorithm and have given a regret bound of $O(\sqrt{dkT \log |\mathcal{X}|})$[1]. However, here $\mathcal{X}$ is the set of all $s - t$ paths without cycles. Recently, Neu & Bartók (2013) have further improved the analysis of the algorithm and have shown a regret bound of $O(k^{3/2}\sqrt{T \log(d)})$. The latter analysis is done for the general structured setting and is applicable for any structured problem.

In Helmbold and Warmuth (2007), another special case of structured learning was studied — the problem of predicting permutations. The authors considered applying mirror descent with entropy regularization over the convex hull of the set of all permutation matrices. As a site note, other algorithms besides mirror descent and FTPL exist for the special case of permutations. See, for example, Ailon (2014).

Koolen et al. (2010) have extended the algorithm of Helmbold and Warmuth (2007) to the general problem of structured learning and have given a regret bound of $O(k\sqrt{T \log(d/k)})$. They have further shown that this bound is minimax optimal. In contrast, note that the regret of our algorithm is upper bounded by $O(k^{3/2}\sqrt{T \log(d/k)})$.

The downside of Helmbold & Warmuth and Koolen et al.'s algorithm is its computational complexity. At each round, the algorithm performs a projection onto the convex hull of $\mathcal{X}$, which is expensive for certain sets. Furthermore, in order to return an element from $\mathcal{X}$ rather than one from the convex hull, the algorithm has to call the offline optimization oracle d times. Thus, it can represent the current iterate as a convex combination of $d + 1$ points from $\mathcal{X}$ (Carathéodory's theorem). Thinking of this convex combination as a distribution over $\mathcal{X}$, the algorithm proceeds to sample an element from $\mathcal{X}$ accordingly. In contrast, at each iteration our algorithm does not involve a projection step and only requires one call to the offline optimization oracle.

Digressing slightly, one may ask how the Hedge algorithm fares in this setting. That is when the problem is reduced to prediction with expert advice such that each $x \in \mathcal{X}$ is an expert. Audibert et al. (2013) have shown that the Hedge algorithm is suboptimal by showing a setting in which its regret is $\Omega(d^{3/2}\sqrt{T})$ (for $k = \Theta(d)$).

Bubeck (2011) has conjectured that a similar lower bound is attainable for FTPL algorithms. We partially prove this conjecture by showing that a lower bound of $\Omega(d^{5/4}\sqrt{T})$ is attained for a large class of distributions, thus showing that FTPL is provably suboptimal. This class of distributions includes the Laplace and negative exponential distributions, the uniform distribution over the cube (Kalai & Vempala, 2005) and the Gaussian distribution (Abernethy et al., 2014).

1.2. Contributions

Our contributions are as following:

- We extend the proof technique of Abernethy et al. (2014) to the structured setting. We show an FTPL algorithm with a regret bound of $O(k\sqrt{T \log |\mathcal{X}|})$, which is comparable to the regret bound of Neu & Bartók (2013). We further show that for certain sets, our algorithm has minimax optimal regret of $O(\sqrt{kT \log |\mathcal{X}|})$.
- We demonstrate a lower bound $\Omega(d^{5/4}\sqrt{T})$ for a large and natural class of FTPL algorithms for structured learning. This lower bound is based on a lower bound for EXP2 found in Audibert et al. (2013).

2. Preliminaries

2.1. Online learning

Online learning, or sequential prediction, is a game played between a *learner* and an omniscient *adversary*, also known as nature. The game is played for T rounds. In each round t the learner predicts an element x_t from a pre-

[1] Here d denotes the number of edges in the graph.

determined compact set $\mathcal{X}$. Simultaneously, nature decides on a loss θ_t which the learner is to suffer.

In our case, we assume the loss is linear. That is $\mathcal{X} \subseteq \mathbb{R}^d$ and $\theta_t \in \mathbb{R}^d$ for all $t \in [T]$. At each round t, the learner suffers a loss of $\langle x_t, \theta_t \rangle$. The goal of the learner is to minimize the regret, defined as the difference between the cumulative loss of the learner over all T rounds and the cumulative loss of the best $x \in \mathcal{X}$ in hindsight. More concretely,

$$\mathrm{Regret} := \sum_{t=1}^{T} \langle x_t, \theta_t \rangle - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} \langle x, \theta_t \rangle$$

We say that a learning algorithm is *Hannan-consistent* if it achieves sublinear regret, namely $\mathrm{Regret} \leq o(T)$.

A first attempt at a learning scheme would be the Follow the Leader algorithm. Denote the cumulative loss at time t by $\Theta_t = \sum_{\tau=1}^{t-1} \theta_\tau$. Then Follow the Leader involves picking x_t to be a minimizer of the cumulative loss at time t, i.e. $x_t \in \arg\min_{x \in \mathcal{X}} \langle x, \Theta_t \rangle$. It is easy to see that any deterministic learning algorithm, and in particular the Follow the Leader scheme, is not Hannan-consistent (Cesa-Bianchi & Lugosi, 2006).

In view of the last paragraph, to achieve consistency we shall allow the learner to randomize. Namely, from now on x_t is a random variable supported on $\mathcal{X}$. In this case the learner would like to minimize the expected regret, and accordingly, we say that a strategy is Hannan-consistent if its expected regret is sublinear.

One algorithm for achieving Hannan-consistency in the *Regularized Follow the Leader* (RFTL) algorithm (Shalev-Shwartz, 2011; Hazan, 2012). Here, we assume that $\mathrm{conv}(\mathcal{X})$ is endowed with a convex regularizer $R : \mathrm{conv}(\mathcal{X}) \to \mathbb{R}$. Let $\eta > 0$. At every round t, we choose

$$y_t = \operatorname*{arg\,min}_{x \in \mathrm{conv}(\mathcal{X})} \langle \Theta_t, x \rangle + \eta R(x)$$

Then, in order to choose $x_t \in \mathcal{X}$ the algorithms creates a distribution over $\mathcal{X}$ whose mean is y_t. The algorithm proceeds by sampling x_t from this distribution.

Following Hannan (1957) and Kalai and Vempala (2005), *Follow the Perturbed Leader* (abbreviated FTPL) is another algorithm for achieving sublinear regret. Let γ be some d-dimensional random variable. FTPL is the algorithmic scheme of choosing an $x \in \mathcal{X}$ that minimizes the perturbed cumulative loss:

$$x_t \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \langle x, \Theta_t + \eta\gamma \rangle \tag{1}$$

Intuitively, for both algorithms, if η is small then we expect x_t to be "close" to a minimizer of the (non-perturbed) cumulative loss. If η is large then we expect the distribution over $\mathcal{X}$ to be "close" to the uniform distribution. Namely, η controls how similar the algorithm is to Follow the Leader.

Lastly, throughout our work we assume that the adversary is *oblivious*. This means that the adversary chooses $\theta_1, ..., \theta_T$ in advance. In contrast, at each round t a *non-oblivious* adversary can choose θ_t based on the random choices of the learner up to that round (non-inclusive). An extension of our algorithm to non-oblivious adversaries follows immediately from Cesa-Bianchi and Lugosi (2006, Lemma 4.1).

2.2. Online structured learning

In online structured learning the set $\mathcal{X}$ is a subset of the hypercube, $\mathcal{X} \subseteq \{0,1\}^d$, and we tend to think of the dimension d as being very large. As such, we would like to avoid polynomial dependence on d in our regret bound. This is done in two manners. The first is by dependence on the size of $\mathcal{X}$, which means that we expect low regret for small sets. The second is by assuming that there is a $k \in [d]$ such that for all $x \in \mathcal{X}$, $\sum_{i=1}^{d} x_i \leq k$. We think of k as being much smaller than d, which means we expect low regret for sparse sets.

We now give a few examples of possible applications, also found in Koolen et al. (2010).

2.2.1. Examples

k-sets In this problem, at each round the learner is to predict a set of exactly k coordinates out of d, so that $|\mathcal{X}| = \binom{d}{k}$. Note that for $k = 1$ this is the problem of prediction with expert advice.

Applications include online ad display and personalized news recommendation. In personalized news recommendation, for example, a website can display k news topics out of d for every user. These topics can include current events, economy, foreign affairs and so on. The user can report which news items are for her liking and the website needs to adjust the topics accordingly.

k-truncated permutations This is an extension of the k-sets problem which requires the learner to choose k elements out of n in an ordered manner. In other words, this is a matching between k elements and n elements.

As an application, consider the problem of handling a search query in a database of size n. For each query, we would like to present only k results ordered decreasingly by relevance.

Here each coordinate is 1 iff a specific element is put in a specific position within the ordering, so that $d = k \cdot n$ and $|\mathcal{X}| = n!/(n-k)!$.

Shortest paths Consider a graph $G = (V, E)$ with two

Algorithm 1 Follow the perturbed leader

```
Input: $\eta > 0$, set $\mathcal{X} \subseteq \{0,1\}^d$
Set $\Theta_1 \leftarrow 0$
for $t = 1$ to $T$ do
    Sample $\gamma_t \sim \mathcal{N}(0, I)$
    Predict $x_t \in \arg\min_{x \in \mathcal{X}} \langle x, \Theta_t + \eta\gamma_t \rangle$
    Suffer loss $\langle x_t, \theta_t \rangle$ and accumulate $\Theta_{t+1} \leftarrow \Theta_t + \theta_t$
end for
```

vertices $s, t \in V$, respectively referred to as the *source* and *sink*. The learner is to predict a shortest path between s and t. Among applications of this problem are network routing in asymmetric communication and road navigation.

The set $\mathcal{X}$ represents the set of all such possible paths. With $d = |E|$, each element $x \in \mathcal{X}$ is a vector representing a path, with $x_i = 1$ iff the path crosses the i'th edge. Here k represent the maximum number of edges in any path.

Spanning trees In this problem, we are once again fixing a graph $G = (V, E)$. At every round the learner has to predict a spanning tree of G.

Spanning trees are often used in network-level communication protocols. Probably their most famous use is for Ethernet bridges to distributively decide on a cycle-free topology of the network.

Here, $d = |E|$ and $k = |V| - 1$. Every element $x \in \mathcal{X}$ represents a spanning tree of G with $x_i = 1$ iff the i'th edge participates in the spanning tree.

3. Our algorithm

In the following we bound the regret of our FTPL algorithm in the structured learning setting. The proof is an extension of the proof technique of Abernethy et al. (2014).

Theorem 1. *Consider algorithm 1. Suppose that for all $x \in \mathcal{X}$, $\sum_{i=1}^{d} x_i \le k$ for some $k \in [d]$. Further suppose the adversary is oblivious and its losses are bounded as $\theta_t \in [0,1]^d$. Then the expected regret satisfies*

$$\mathbb{E}_{\gamma_1,\dots,\gamma_T}[\text{Regret}] \le \sqrt{2k \log|\mathcal{X}|}\left(\frac{kT}{\eta} + \eta\right)$$

Additionally, by setting $\eta = \sqrt{kT}$ the expected regret is bounded by $2k\sqrt{2T \log|\mathcal{X}|}$.

For each round t, consider the function $\min_{x \in \mathcal{X}} \langle x, \Theta_t + \eta\gamma_t \rangle$. It is differentiable almost everywhere (w.r.t Θ_t) and we can write its gradient as $\sum_{x \in \mathcal{X}} \mathbf{1}_{[x_t = x]} \cdot x$. The probability of selecting a certain $x \in \mathcal{X}$ is the expected value of the indicator function $\mathbf{1}_{[x = x_t]}$. With these in mind and following Abernethy et al. (2014), we define a potential function

$$\Phi(\theta) = \mathbb{E}_{\gamma \sim \mathcal{N}(0,I)}\left[\min_{x \in \mathcal{X}} \langle x, \theta + \eta\gamma \rangle\right]$$

and we get that its gradient at Θ_t is $\mathbb{E}_{\gamma_t}[x_t]$. Namely, $\nabla\Phi(\Theta_t) = \sum_{x \in \mathcal{X}} \Pr_\gamma[x_t = x] \cdot x = \mathbb{E}_{\gamma_t}[x_t]$. Note that as a consequence, $\langle \nabla\Phi(\Theta_t), \theta_t \rangle = \mathbb{E}[\langle x_t, \theta_t \rangle]$.

In the following, we will bound the regret of the algorithm in terms of Φ. We have that Φ is twice continuously differentiable everywhere (Abernethy et al., 2014, Lemma 7). A Taylor's expansion of Φ with a second order remainder, is

$$\Phi(\Theta_{t+1}) = \Phi(\Theta_t) + \langle \nabla\Phi(\Theta_t), \theta_t \rangle + \frac{1}{2}\langle \theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t \rangle$$

for some $\tilde{\theta}_t$ on the line segment connecting Θ_t and $\Theta_{t+1} = \Theta_t + \theta_t$. Thus,

$$\mathbb{E}[\langle x_t, \theta_t \rangle] = \Phi(\Theta_{t+1}) - \Phi(\Theta_t) - \frac{1}{2}\langle \theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t \rangle \quad (2)$$

Note that Φ is concave since the minimum of linear functions is a concave function, and thus its Hessian is negative semidefinite.

Since the right hand side of equation 2 consists of a telescopic term, summing it over all $t \in [T]$ results in

$$\mathbb{E}\left[\sum_{t=1}^{T}\langle x_t, \theta_t \rangle\right] = \Phi(\Theta_{T+1}) - \Phi(\Theta_1) - \frac{1}{2}\sum_{t=1}^{T}\langle \theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t \rangle$$

By Jensen's inequality, $\Phi(\Theta_{T+1}) \le \min_{x \in \mathcal{X}} \mathbb{E}_\gamma[\langle x, \Theta_{T+1} + \eta\gamma \rangle]$. Then the bound $\Phi(\Theta_{T+1}) \le \min_{x \in \mathcal{X}} \langle x, \Theta_{T+1} \rangle$ is obtained since $\mathbb{E}[\gamma] = 0$. Recall that $\min_{x \in \mathcal{X}} \langle x, \Theta_{T+1} \rangle$ is the loss of the best decision in hindsight, then by rearranging we have the following bound on the expected regret.

$$\mathbb{E}[\text{Regret}] \le -\Phi_\eta(\Theta_1) - \frac{1}{2}\sum_{t=1}^{T}\langle \theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t \rangle \quad (3)$$

We now prove theorem 1 by bounding the terms on the right hand side of inequality 3.

Proof of theorem 1. For the moment, assume that $-\langle \theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t \rangle \le (2k/\eta)\sqrt{2k \log|\mathcal{X}|}$ holds. We will prove the correctness of this statement in lemma 2.

Thus, to complete the regret bound we consider $-\Phi(\Theta_1) = -\eta\mathbb{E}[\min_{x \in \mathcal{X}} \langle x, \gamma \rangle]$, which equals $\eta\mathbb{E}[\max_{x \in \mathcal{X}} \langle x, \gamma \rangle]$ since normal random variables are symmetric. Bounds on the expected maxima of normal random variables imply that $\mathbb{E}[\max_{x \in \mathcal{X}} \langle x, \gamma \rangle] \le \sqrt{2k \log|\mathcal{X}|}$, which we derive in lemma 9 in the appendix.

Putting the bounds into inequality 3, we have that

$$\mathbb{E}[\text{Regret}] \le \eta\sqrt{2k\log|\mathcal{X}|} + \frac{kT}{\eta}\sqrt{2k\log|\mathcal{X}|}$$

which concludes the proof of the theorem. □

We finish this section with the proof of the following lemma.

Lemma 2. *Suppose the conditions of theorem 1 hold. We have,*

$$-\langle\theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t\rangle \le \frac{2k}{\eta}\sqrt{2k\log|\mathcal{X}|}$$

Proof. By our assumption $\|\theta_t\|_\infty \le 1$. Thus, $-\langle\theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t\rangle \le \sum_{i,j}|H_{i,j}|$, where we denoted $H = \nabla^2\Phi(\tilde{\theta}_t)$.

By Abernethy et al. (2014, Lemma 7), we have that

$$H_{i,j} = \frac{1}{\eta}\mathbb{E}\left[\hat{x}(\tilde{\theta}_t + \eta\gamma)_i\gamma_j\right] \tag{4}$$

with $\hat{x}(z) \in \arg\min_{x\in\mathcal{X}}\langle x, z\rangle$. Note that the right hand side is well defined since $\min_{x\in\mathcal{X}}\langle x, \theta + \eta\gamma\rangle$ is differentiable almost everywhere (w.r.t θ) and its gradient is $\hat{x}(\theta + \eta\gamma)$.

Let us abbreviate $\hat{x}(\tilde{\theta} + \eta\gamma)$ as $\hat{x}$. Fix some j, then $\sum_{i=1}^d |H_{i,j}| \le (k/\eta)\sum_{x\in\mathcal{X}}|\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}]|$. This result is derived in the appendix by separating the positive and negative entries of the Hessian.

Next, we have two observations. The first is that $\sum_{x\in\mathcal{X}}\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}] = \mathbb{E}[\gamma_j] = 0$. The second is about the sign of $\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}]$. We claim that it is negative if $x_j = 1$ and positive otherwise. To see that, notice that γ_j is a symmetric random variable, so that for each $\alpha > 0$ the density of γ_j at α and at $-\alpha$ is the same. If $x_j = 1$, the event $\hat{x} = x$ is more probable if $\gamma_j = -\alpha$ than when $\gamma_j = \alpha$. If $x_j = 0$ then the opposite is true.

Following these two observations, $|\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}]|$ equals to $\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}]$ whenever $x_j = 0$, and equals to $-\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}]$ whenever $x_j = 1$. Since the sum of these two expectations is 0, it follows that $\sum_{x\in\mathcal{X}}|\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}]| = -2\sum_{x\in\mathcal{X}}\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x\wedge x_j=1]}]$. Thus, $\sum_{x\in\mathcal{X}}|\mathbb{E}[\gamma_j\mathbf{1}_{[\hat{x}=x]}]| = -2\mathbb{E}[\gamma_j\hat{x}_j] = -2\eta H_{j,j}$.

Putting it all together, we get that $\sum_{i,j}|H_{i,j}| \le -2k\,\mathrm{Tr}(H)$. Equation 4 implies that $-\mathrm{Tr}(H)$ equals to $-\frac{1}{\eta}\sum_{j=1}^d\mathbb{E}[\hat{x}(\tilde{\theta} + \eta\gamma)_j\gamma_j]$. Since a Gaussian is symmetric this also equals to $\frac{1}{\eta}\sum_{j=1}^d\mathbb{E}[\hat{x}(\tilde{\theta} - \eta\gamma)_j\gamma_j]$ and it is upper bounded by $\frac{1}{\eta}\mathbb{E}\left[\max_{x\in\mathcal{X}}\langle x, \gamma\rangle\right]$. As shown in lemma 9 (in the appendix), this expected maxima of normal random variables is bounded from above by $(1/\eta)\sqrt{2k\log|\mathcal{X}|}$. □

3.1. A better bound for the k-sets problem

For the k-sets problem, we can show an upper bound of $2\sqrt{2Tk\log\binom{d}{k}}$ on the regret of our algorithm. The proof of the bound follows along the lines of theorem 1. The only difference is that lemma 2 is replaced by the following lemma.

Lemma 3. *Let $k \in [d]$. Let $\mathcal{X} = \{x \in \{0,1\}^d : \sum_{i=1}^d x_i = k\}$. We have,*

$$-\langle\theta_t, \nabla^2\Phi(\tilde{\theta}_t)\theta_t\rangle \le \frac{2}{\eta}\sqrt{2k\log|\mathcal{X}|}$$

The difference between this lemma and lemma 2 is that this lemma lacks a factor of k. Similarly, the regret bound attained for this problem is better than that of theorem 1 by a factor of $\sqrt{k}$. Note that this upper bound is comparable to the one achieved by the algorithm of Koolen et al. (2010).

What makes this improvement possible is the fact that for the k-sets problem, for any $i \neq j$, the perturbation of the i'th coordinate and the event that the learner would pick the j'th coordinate are positively correlated. Namely, we have $\mathbb{E}[\hat{x}(\theta_t + \eta\gamma)_j\gamma_i] \ge 0$. This property does not hold for general structured problems, and we will use this fact in the lower bound of the next section.

4. Lower bound

In this section we will present our lower bound for FTPL in structured learning. We will focus on sets $\mathcal{X}$ such that for all $x \in \mathcal{X}$, $\sum_{i=1}^d x_i = k$ and $k = \Theta(d)$. The lower bound is an adaption of Audibert et al. (2013, Theroem 1).

Theorem 4. *Consider an FTPL algorithm whose perturbations satisfy the following: $\gamma_1, ..., \gamma_d$ are IID, logconcave and have variance 1. Furthermore, there is a constant $L > 0$ such that for all $\mu_P, \mu_Q \in \mathbb{R}^d$, the joint distributions P, Q of $\gamma + \mu_P, \gamma + \mu_Q$ respectively, satisfy the total variation distance[2] constraint of $\mathrm{TV}(P, Q) \le L\|\mu_P - \mu_Q\|_1$. Let $T \ge \sqrt{d}/L$. There is a set $\mathcal{X} \subseteq \{0,1\}^d$ and a constant $c > 0$ for which the regret of any such FTPL algorithm satisfies*

$$\sup_{\text{adversary}} \mathbb{E}[\text{Regret}] \ge c \cdot d^{5/4}\sqrt{T/L}$$

We defer the proof of theorem 4 to section 4.3 and present its main ideas in the following simpler theorem for Gaussian perturbations.

Theorem 5. *Consider an FTPL algorithm with $\gamma \sim \mathcal{N}(0, I)$. Let $T \ge 10\sqrt{d}$. There is a set $\mathcal{X} \subseteq \{0,1\}^d$ and a constant $c > 0$ for which this FTPL algorithm satisfies*

$$\sup_{\text{adversary}} \mathbb{E}[\text{Regret}] \ge c \cdot d^{5/4}\sqrt{T}$$

[2] $\mathrm{TV}(P, Q) := \sup_E|\Pr_P[E] - \Pr_Q[E]|$.

Proof. In the sequel we will describe a set $\mathcal{X}$ and two adversaries. The first adversary satisfies $\mathbb{E}[\mathrm{Regret}_1] \geq \min\left\{Td/16, d\eta\sqrt{2}/32\right\}$ and the second adversary satisfies $\mathbb{E}[\mathrm{Regret}_2] = (Td/16) \cdot \mathrm{erf}(\sqrt{d}/4\eta)$. Therefore, the expected regret of the algorithm satisfies

$$\sup_{\text{adversary}} \mathbb{E}[\mathrm{Regret}] \geq \max\{\mathbb{E}[\mathrm{Regret}_1], \mathbb{E}[\mathrm{Regret}_2]\}$$
$$\geq \min\left\{0.02Td, 0.05d^{5/4}\sqrt{T}\right\}$$

where the last inequality is technical, and is provided as lemma 11 in the appendix. Applying the assumption that $T \geq 10\sqrt{d}$ completes the proof. □

Intuitively, $\mathcal{X}$ consists of two separate problems. The first is the problem of selecting any $d/4$ coordinates out of the first $d/2$ coordinates. The second is, out of the remaining $d/2$ coordinates, to select either the first $d/4$ coordinates or the last $d/4$ coordinates.

Let us define the set concretely. We will split the d coordinates into three intervals. I_1 includes coordinates $1, ..., d/2$, I_2 includes coordinates $d/2+1, ..., 3d/4$ and I_3 includes coordinates $3d/4+1, ..., d$. Then the set $\mathcal{X}$ is defined as,

$$\mathcal{X} = \{x \in \{0,1\}^d : \sum_{i \in I_1} x_i = \frac{d}{4} \text{ and}$$
$$((x_i = 1,\ \forall i \in I_2) \text{ or } (x_i = 1,\ \forall i \in I_3))\}$$

Notice that $k = d/2$ and $|\mathcal{X}| = 2\binom{d/2}{d/4} = O(2^{d/2})$, so that for this specific set $\mathcal{X}$, the regret of algorithm 1 is $O(d^{3/2}\sqrt{T})$ while the regret of Koolen et al.'s algorithm is $O(d\sqrt{T})$. Theorem 5 shows a lower bound of $\Omega(d^{5/4}\sqrt{T})$ on the regret of algorithm 1.

In the following we will define two adversaries, each causing the learner to suffer loss on a disjoint half of the coordinates. The first adversary assigns positive loss only to the first $d/2$ coordinates. For it, the learner will want η to be as small as possible. The second adversary assigns positive loss only for coordinates $d/2+1, ..., d$. For it, the learner would like η to be as large as possible, as its regret depends on $1/\eta$.

4.1. The first adversary

In every round the adversary assigns a loss of $1-\epsilon$ to coordinates $\{1, ..., d/4\}$ for $\epsilon \in (0,1)$, a loss of 1 to coordinates $\{d/4+1, ..., d/2\}$ and 0 for the rest of the coordinates.

Lemma 6. *Under the conditions of theoren 5 the expected regret satisfies*

$$\mathbb{E}[\mathrm{Regret}_1] \geq \min\left\{\frac{Td}{16}, \frac{d\eta\sqrt{2}}{32}\right\}$$

Proof. Denote by $p_{t,i}$ the probability that at time t the learner would predict an x such that $x_i = 1$. Recall that at each round, our algorithm chooses $x_t \in \mathcal{X}$ as in equation 1. We note the following:

- Since the learner has to pick exactly $d/4$ coordinates at every round, this implies that $\sum_{i=1}^{d/2} p_{t,i} = d/4$.
- The coordinates $1, ..., d/4$ suffer the same loss, and the coordinates $d/4+1, ..., d/2$ suffer the same loss. We assume that the perturbations γ are IID. Hence it must be that by symmetry, $p_{t,1} = ... = p_{t,d/4}$ and that $p_{t,d/4+1} = ... = p_{t,d/2}$.
- From the last two bullets we conclude that for every t, for every $i \in \{1, ..., d/4\}$ and for every $j \in \{d/4+1, ..., d/2\}$ we have that $p_{t,i} + p_{t,j} = 1$.
- At every round, the difference in the cumulative loss between a coordinate in $\{d/4+1, ..., d/2\}$ and any coordinate in $\{1, ..., d/4\}$ is strictly increasing. Hence, as the rounds progress, the learner is less likely to pick a coordinate from $\{d/4+1, ..., d/2\}$. Concretely, for every coordinate $i \in \{d/4+1, ..., d/2\}$, $p_{t,i}$ is nonincreasing in t.

Against this adversary, the best choice in hindsight would be to predict coordinates $1, ..., d/4$. The cumulative loss for this choice is $Td(1-\epsilon)/4$. Then, the regret of the learner is

$$\sum_{t=1}^{T}\left(\frac{d(1-\epsilon)}{4}p_{t,1} + \frac{d}{4}p_{t,d/2}\right) - \frac{Td(1-\epsilon)}{4}$$
$$= \sum_{t=1}^{T}\frac{d\epsilon}{4}p_{t,d/2} \geq \frac{Td\epsilon}{4}p_{T,d/2}$$

We will later prove in lemma 7 that $p_{T,1} - p_{T,d/2} \leq (\epsilon T)/(\eta\sqrt{2})$. Since $p_{T,1} + p_{T,d/2} = 1$, it holds that $p_{T,d/2} \geq 1/2 - \epsilon T/(2\sqrt{2}\eta)$. Finally, setting $\epsilon = \min\{\eta/(T\sqrt{2}), 1\}$ gives $p_{T,d/2} \geq 1/4$. Thus, the lower bound on the regret follows. □

To complete the lower bound of the first adversary, we state the following lemma:

Lemma 7. *We have that $p_{T,1} - p_{T,d/2} \leq (\epsilon T)/(\eta\sqrt{2})$.*

Proof. After T rounds, the difference in the cumulative loss between a coordinate in $\{d/4+1, ..., d/2\}$ and any coordinate in $\{1, ..., d/4\}$ is exactly ϵT. Let P be the distribution of $\mathcal{N}(\mu_P, I)$, where μ_P is $-\epsilon T/\eta$ for the first $d/4$ coordinates and 0 for the rest. Note that for the learner to pick the $d/4$ coordinates with the least perturbed loss is the same as sampling from P and picking the smallest $d/4$ coordinates (out of the first $d/2$).

Let Q be the distribution of $\mathcal{N}(\mu_Q, I)$, where μ_Q is the same as μ_P except that the first coordinate is interchanged with the $d/2$'th coordinate. In other words μ_Q is 0 on the first coordinate and $-\epsilon T/\eta$ on the $d/2$'th coordinate.

Consider the same sampling process as described before, except over Q rather than P. Denote by q_i the probability of selecting the i'th coordinate out of the smallest $d/4$ coordinates. Therefore, $p_{T,1} = q_{d/2}$. In particular, $p_{T,1} - p_{T,d/2} = q_{d/2} - p_{T,d/2} \leq \text{TV}(Q,P)$ and by Pinsker's inequality, $\text{TV}(Q,P) \leq \sqrt{(1/2)\text{KL}(P||Q)}$. Note that P and Q are both multivariate normal distributions and hence the KL divergence between them equals $(1/2)\|\mu_P - \mu_Q\|^2 = (\epsilon T/\eta)^2$. □

4.2. The second adversary

On even rounds the adversary assigns a loss of 1 for each coordinate in $\{3d/4+1, ..., d\}$ and 0 for the rest. On odd rounds it assigns a loss of 1 for every coordinate in $\{d/2+1, ..., 3d/4\}$ and 0 for the rest.

Lemma 8. *Under the conditions of theorem 5, for the second adversary we have*

$$\mathbb{E}[\text{Regret}_2] = \frac{Td}{16}\,\text{erf}\left(\frac{\sqrt{d}}{4\eta}\right)$$

Proof. At the beginning of odd rounds, the cumulative loss of coordinates in $\{d/2+1, ..., d\}$ is identical. By symmetry, our algorithm would pick an $x \in \mathcal{X}$ uniformly at random. Therefore the expected loss of the learner in any odd round is $d/8$ and $Td/16$ in total.

On even rounds the distribution over the choices of the learner is the same between such rounds. Recall that the cumulative loss of coordinates in $\{d/2+1, ..., 3d/4\}$ is larger by 1 than coordinates in $\{3d/4+1, ..., d\}$. Then the probability of suffering a loss on an even round is:

$$\Pr\left[\eta \sum_{i=3d/4+1}^{d} \gamma_i < \sum_{i=d/2+1}^{3d/4} (\eta\gamma_i + 1)\right]$$

Let $Z = \sqrt{2/d}\sum_{i=d/2+1}^{3d/4}(\gamma_{i+d/4} - \gamma_i)$, then the above probability is $\Pr[Z \leq \sqrt{2d}/4\eta]$ and the expected loss of the learner in even rounds is $(Td/8)\Pr[Z \leq \sqrt{2d}/4\eta]$.

After T iterations, the loss of any coordinate in $\{d/2+1, ..., d\}$ is $T/2$, so that the loss of any $x \in \mathcal{X}$ is $Td/8$. To conclude we note that $\Pr[Z \leq \sqrt{2d}/4\eta] = (1 + \text{erf}(\sqrt{d}/4\eta))/2$. □

4.3. The logconcave case

To prove our regret lower bound for any logconcave distribution, two adjustments to the proof for the Gaussian case are required. The first of which, is in the proof of lemma 7. There, instead of bounding the total variation distance between P and Q using Pinsker's inequality, we use our assumption that $\text{TV}(P,Q) \leq L\|\mu_P - \mu_Q\|_1 \leq 2L\epsilon T/\eta$. The second difference between the proofs is in the proof of lemma 8, estimating the tail of the random variable $Z := \sqrt{2/d}\sum_{i=d/2+1}^{3d/4}(\gamma_{i+d/4} - \gamma_i)$. Originally, we have used the fact that the normalized sum of normal random variables is a standard normal random variable. Analogously, sums and differences of logconcave random variables are also logconcave (Prékopa, 1973). Then, from its definition, Z is a symmetric logconcave random variable that is isotropic, namely it has mean 0 and variance 1. Let f denote the PDF of Z, and F its CDF. We have the following properties of Z:

- For any isotropic logconcave density $f(0) \geq 1/8$ (Lovász & Vempala, 2007, Lemma 5.5).
- By symmetry $F(0) = 1/2$ and for any $s \in \mathbb{R}$, $F(s) = 1 - F(s)$.

From the first bullet and by logconcavity, for any $s < 0$, $\log F(s) \leq \log F(0) + (\log F)'\big|_{t=0} \cdot s$. We note that $(\log F)'\big|_{t=0} = f(0)/F(0) \geq 1/4$, which means that $F(s) \leq (1/2)\exp(s/4)$. Then for any $s > 0$, $F(s) = 1 - F(-s) \geq 1 - \frac{1}{2}\exp(-s/4)$.

Then the regret of the second adversary satisfies

$$\begin{aligned}\mathbb{E}[\text{Regret}_2] &\geq \frac{Td}{8}\left(1 - \frac{1}{2}\exp\left(-\frac{\sqrt{2d}}{16\eta}\right)\right) - \frac{Td}{16} \\ &= \frac{Td}{16}\left(1 - \exp\left(-\frac{\sqrt{2d}}{16\eta}\right)\right)\end{aligned}$$

which we can apply with the rest of the proof, including the technical lemma, since $1 - \exp(-x)$ is nondecreasing and concave (see supplementary material for more detail).

5. Experiments

We evaluated our algorithm, Neu & Bartók's algorithm and Koolen et al.'s algorithm on the shortest path problem. We constructed a directed graph with 53 vertices and 201 edges. The graph has a grid-like structure; where an undirected edge appears there is an edge in the opposite direction. Additionally, we have added a source vertex s and a destination vertex t. Both s and t are connected to the rest of the graph only through outgoing or incoming edges respectively. The graph is shown in figure 1.

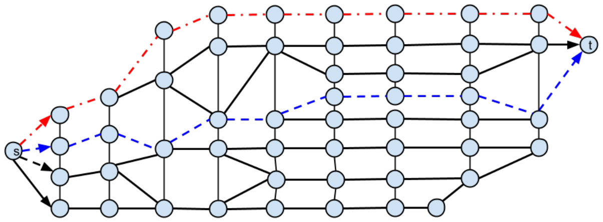

Figure 1. Graph for experiments. A directed edge indicates that an edge exists in the direction it is drawn. An undirected edge indicates there are two directed edges in its place, one for every direction. The two best paths are shown colored and dashed.

The three algorithms we compare require to tune a parameter η. The minimax optimal choice of η depends on k, which in this case is the length of the longest $s-t$ path — which is NP-hard and even NP-hard to approximate (Björklund et al., 2004). Therefore, we resort to choosing η based on the empirical performance of each of the algorithms against our adversary.

We have built the adversary so that the choices the algorithms make are neither completely arbitrary (the loss of all edges are random and independent) nor fixed on all rounds (there is a path with least cumulative loss at all rounds). The loss on each edge is picked uniformly at random in $\{0, 1\}$. However, we override this random loss in the following manner. We have picked two $s-t$ paths in advance (colored and dashed in figure 1). In every round, we choose one of these paths in an alternating fashion. We set of a loss of 0.75 to each of its edges and 0 to the edges of the other path. We do that so that these two paths are always optimal on average compared and yet switching often between the two would result in high regret.

We choose a nominal η by evaluating each algorithm on our problem with $T = 100$ on values of η ranging from e^n where n an integer is between 0 and 10 (we choose η from e^{-n} for Koolen et al.'s algorithm). We run our experiments with horizons T between 1 and 100. We choose the nominal η and multiply it by $\sqrt{T/100}$ (divide by $\sqrt{T/100}$ for Koolen et al.'s algorithm). In figure 2, we plot the average runtime of each algorithm as well as the average regret of each algorithm, as a function of T.

As one can see from the plots, both FTPL algorithms have approximately the same runtime and the same regret and are much faster than the RFTL algorithm. As expected, both FTPL algorithms perform slightly worse than the RFTL algorithm in terms of regret.

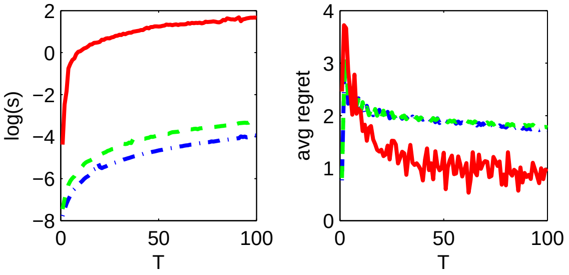

Figure 2. On the left, average runtime in seconds as a function of T (in log scale). On the right, average regret as a function of T. In dot-dashed blue is our algorithm, in dashed green Neu & Bartók's and in solid red is Koolen et al.'s algorithm.

6. Discussion and future work

Many real-life problems are, in fact, both online and combinatorial in nature. In this paper we consider algorithms for this setting. From a statistical perspective, this problem has been solved in Koolen et al. (2010), that have given a regret bound of $O(k\sqrt{T}\log(d/k))$ and a matching lower bound.

In our work we argue that this problem is far from being solved from a computational perspective. As can be seen in our experiments, FTPL algorithms are computationally efficient. We have shown a new FTPL algorithm with state of the art regret bound. Furthermore, we have shown a lower bound for most FTPL algorithms found in the literature.

It is easy to see that our lower bound doesn't hold for non-IID perturbations. Focusing on the construction of the lower bound, if the perturbations of the last $d/2$ coordinates are dependent (i.e. the first $d/4$ are all equal and the second $d/4$ are all equal) then the probability of switching between the sets of $d/4$ coordinates on odd rounds (lemma 8) becomes independent of d (instead of proportional to $\sqrt{d}$ in the IID case). We conjecture that FTPL with non-IID perturbations can achieve optimal regret.

We conjecture that $O(d^{5/4}\sqrt{T})$ is the correct rate. Intuitively, when the additional dependence on k (or $d^{1/4}$ as in the lower bound) appears in the regret bound, the set $\mathcal{X}$ cannot be too large. This is in contrast to the k-sets problem, for which $\mathcal{X}$ is the largest set for a given k and for which our algorithm has optimal regret. We conjecture that our upper bound fails to capture the interplay between the size of $\mathcal{X}$ and the appearance of the additional $d^{1/4}$ factor.

We ask whether the logconcavity assumption is necessary. We can still prove that this lower bound holds asymptotically if T is fixed and d goes to infinity (using the Central Limit Theorem). However, we argue that the regime when $T \gg d$ is much more interesting.

References

Abernethy, Jacob, Lee, Chansoo, Sinha, Abhinav, and Tewari, Ambuj. Online linear optimization via smoothing. *The Journal of Machine Learning Research*, 35: 807–823, 2014.

Ailon, Nir. Improved bounds for online learning over the permutahedron and other ranking polytopes. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, pp. 29–37, 2014.

Audibert, Jean-Yves, Bubeck, Sébastien, and Lugosi, Gábor. Regret in online combinatorial optimization. *Mathematics of Operations Research*, 39(1):31–45, 2013.

Björklund, Andreas, Husfeldt, Thore, and Khanna, Sanjeev. Approximating longest directed paths and cycles. In *Automata, Languages and Programming*, pp. 222–233. Springer, 2004.

Blum, Avrim. *On-line algorithms in machine learning*. Springer, 1998.

Bubeck, Sébastien. Introduction to online optimization. *Lecture Notes*, 2011.

Cesa-Bianchi, Nicolo and Lugosi, Gábor. *Prediction, learning, and games*. Cambridge University Press Cambridge, 2006.

Cesa-Bianchi, Nicolo, Freund, Yoav, Haussler, David, Helmbold, David P, Schapire, Robert E, and Warmuth, Manfred K. How to use expert advice. *Journal of the ACM (JACM)*, 44(3):427–485, 1997.

Cover, Thomas M. Universal portfolios. *Mathematical finance*, 1(1):1–29, 1991.

Devroye, Luc, Lugosi, Gábor, and Neu, Gergely. Prediction by random-walk perturbation. *arXiv preprint arXiv:1302.5797*, 2013.

Foster, Dean P and Vohra, Rakesh. Regret in the on-line decision problem. *Games and Economic Behavior*, 29 (1):7–35, 1999.

Hannan, James. Approximation to bayes risk in repeated play. *Contributions to the Theory of Games*, 3:97–139, 1957.

Hazan, Elad. The convex optimization approach to regret minimization. *Optimization for machine learning*, pp. 287, 2012.

Helmbold, David P and Warmuth, Manfred K. Learning permutations with exponential weights. In *Learning theory*, pp. 469–483. Springer, 2007.

Kalai, Adam and Vempala, Santosh. Geometric algorithms for online optimization. In *Journal of Computer and System Sciences*. Citeseer, 2002.

Kalai, Adam and Vempala, Santosh. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005.

Koolen, Wouter M, Warmuth, Manfred K, and Kivinen, Jyrki. Hedging structured concepts. In *COLT*, pp. 93–105, 2010.

Kuzmin, Dima and Warmuth, Manfred K. Optimum follow the leader algorithm. In *Learning Theory*, pp. 684–686. Springer, 2005.

Littlestone, Nick and Warmuth, Manfred K. The weighted majority algorithm. *Information and computation*, 108 (2):212–261, 1994.

Lovász, László and Vempala, Santosh. The geometry of logconcave functions and sampling algorithms. *Random Structures & Algorithms*, 30(3):307–358, 2007.

Neu, Gergely and Bartók, Gábor. An efficient algorithm for learning with semi-bandit feedback. In *Algorithmic Learning Theory*, pp. 234–248. Springer, 2013.

Prékopa, András. Logarithmic concave measures and functions. *Acta Scientiarum Mathematicarum*, 34(1):334–343, 1973.

Rakhlin, Sasha, Shamir, Ohad, and Sridharan, Karthik. Relax and randomize: From value to algorithms. In *Advances in Neural Information Processing Systems*, pp. 2141–2149, 2012.

Shalev-Shwartz, Shai. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

Takimoto, Eiji and Warmuth, Manfred K. Path kernels and multiplicative updates. *The Journal of Machine Learning Research*, 4:773–818, 2003.

Van Erven, Tim, Kotlowski, Wojciech, and Warmuth, Manfred K. Follow the leader with dropout perturbations. In *Proceedings of Conference on Learning Theory (COLT)*, 2014.

Zinkevich, Martin. Online convex programming and generalized infinitesimal gradient ascent. *AAAI*, 2003.