Online Multivariate Optimization

Bhuvesh Kumar
13203

Govind Gopakumar
16111009

Vikas Jain
13788

Abstract

In a general classification setting, our notions of performance need to be quantified using multiple measures. In some cases like classification problems with label imbalance, it proves to be useful to optimize over non-decomposable measures such as F-measure, G-mean, and H-mean. In our work, we consider concave performance measures and propose an online primal-dual method `OPIUM` to learn an online learning problem in an adversarial setting to optimize the learning objective which is based on these concave performance measures. We applied Online Mirror Descent technique in our primal-dual solver method as in saddle point optimization problem[1] to obtain a sublinear regret bound similar to the case of stochastic optimization setting[4] with concave performance measures. We also employ and analyze our proposed algorithm with a general united framework[2] for learning under delayed feedback and establish the applicability of our algorithm even when faced with delays.

1 Introduction

Every learning problems boils down to optimizing a performance measure. Common performance measures include Accuracy, Precision, Specificity, squared error loss etc. These common additive performance measure are not well-behaved in some particular learning settings like classification with label imbalance. Therefore, non-additive performance measures like prec@k, F-measure, G-mean etc. find their application in realizing more practical learning setting.

Recently, Narasimhan et al. propose a stochastic primal-dual method (`SPADE`) to optimize concave and pseudo-linear performance measure in a stochastic learning setting. Motivated by this work, we develop methods for optimizing concave performance measure when face with an adaptive adversary or even with delayed feedback.

In this work, we consider the learning objectives that are based on concave performance measures. These measures can be written as concave functions of true positive (TPR) and negative (TNR) rates and include G-mean, H-mean etc[4]. We exploit the dual structure of these functions via their Fenchel dual to linearize them in terms of the TPR, TNR variables. Our proposed method tunes the primal as well as dual variables in this linearization in a single step and maximizes the weighted TPR-TNR combination. These updates are done in an online fashion using online mirror descent steps. We also analyzed our algorithm under delayed feedback using the general black-box algorithm proposed by [2] which takes a non-delayed feedback OCO algorithm and run it in delayed feedback setting.

2 Problem Setting

We are considering a normal binary classification setting. Here, we are given points from $\mathcal{X} \subset \mathbb{R}^d$, which are the points we wish to classify. We classify using linear classifiers, $w \in \mathcal{W} \subset \mathbb{R}^d$. For the sake of our objective function, we will define reward functions similar to those in [4]. Following their convention, we have,

$$r^+(w,x,y) = \frac{1}{p} r(y, \langle w, x\rangle)\mathbf{1}(y=1)$$

and a similar version for the negative class. Here, p denotes the fraction of positive examples in the dataset that we wish to learn over. Our objective function that we wish to maximize, is of course the performance measure $P_\psi(w) = \psi(P(w), N(w))$. Here, both the arguments specify the TPR and the TNR rate, and ψ is a concave link function that links these two together in the performance measure. In addition, we shall use the idea of a Fenchel conjugate / dual of the link function under consideration as,

$$\psi^*(\alpha,\beta) = \inf_{u,v\in\mathbb{R}} (\alpha u + \beta v - \psi(u,v))$$

This is the standard definition of a the dual of a function.

3 Optimizing Concave Performances in the face of adversity

3.1 Current literature

Existing methods to deal with these performance measures do promise good rates of convergence as well as error bounds. The work of [3] provides us with a batch gradient descent like technique which works on ranking objectives, and proves a vanishing regret bound in the non-stochastic adversarial setting. We also have the work of [4] that provides a stochastic optimization setting to optimize these non-decomposable performance metrics, but their technique works in the stochastic non-adversarial setting. We would like to form an extension that works in the non-stochastic, adversarial setting. A general technique is indeed outlined in [5], but their method (owing to the general nature) is not suitable to specific problems, indeed, at every iteration their algorithm might seek to do a step that is not computationally efficient. We therefore would like a solution that performs as fast as existing methods, but do work in case of non-stochastic adversarial settings as well. This is where we turn to the technique of mirror descent, and the formulation of the problem as a saddle point optimization problem.

Notice that the technique in [4] motivates a natural saddle point formulation. They optimize iteratively on the dual and the primal of the function, and we would like to see if existing techniques that work on such a setting can be applied directly.

3.2 Mirror Descent

In this section, we shall describe the optimization technique called `Mirror Descent`. The description as well as the regret bound proofs have been adapted from [1] and the associated blog post.

Our objective remains the minimization of a convex function, denoted as f. We shall assume for the purposes of analysis that this function is L-Lipschitz. Now, consider an α-stronlgy convex function, Φ. This will be used as a "mirror map", which is in fact where the technique derives its name from.

We use our "mirror map" to map a point in our primal space, $\mathcal{X}$, to some point in the "dual" space $\nabla\Phi(x)$. The idea is that doing gradient descent makes sense when the gradients lie in the same space as the original space. This is not a valid assumption to make if our original space is not a Hilbert space (in which gradients lie in the same space, a consequence of the Riesz Representation theorem). We use the notion of our mirror map to do gradient descent. When we map a point using the mirror map, we can then proceed to take a gradient descent step in this dual space, and then use an inverse map to get back a point in our primal space. This inverse mapping is guaranteed if we make certain assumptions on the map Φ, which are correct in our particular setting.

Formally, the "mirror descent" step is taken as,

$$\nabla\Phi(y^{t+1}) = \nabla\Phi(x^t) - \eta\nabla f(x^t)$$

Since the inverse mapped point may lie outside our constraint set $\mathcal{X}$, we use the general idea of a "Bregman Divergence" to map it back. That is, our final updated point is of the form,

$$x^{t+1} = \arg\min_{x \in \mathcal{X}} D_\Phi(x^t, y^{t+1})$$

The Bregman Divergence used here is defined in terms of our mirror map. It is given by,

$$D_\Phi(x, y) = \Phi(x) - \Phi(y) - \langle \nabla\Phi(y), x - y \rangle$$

Theorem 1. *For a convex, L-Lipschitz objective function f, with an α strongly convex mirror map, MD will satisfy,*

$$f(\frac{1}{t}\sum_{i=1}^{t} x_i) - \min f(x) \leq O(\frac{1}{\sqrt{t}})$$

Proof.

$$\begin{aligned}
f(x_i) - f(x) &\leq g_i^T(x_i - x) && \text{(Convexity)} \\
&\leq \frac{1}{\eta}(\nabla\Phi(x_i) - \nabla\Phi(y_{i+1}))^T(x_i - x) && \text{(Update rule)} \\
&\leq \frac{1}{\eta}(D_\Phi(x, x_i) + D_\Phi(x_i, y_{i+1}) - D_\Phi(x, y_{i+1})) && \text{(Definition of } B_\Phi)
\end{aligned}$$

From the fact that x_{i+1} is the optimal point under the projection, we can write,

$$\begin{aligned}
(\nabla\Phi(x_{i+1}) - \nabla\Phi(y_{i+1}))^T(x_{i+1} - x_i) \leq 0 \\
D_\Phi(x, y_{i+1}) \geq D_\Phi(x, x_{i+1}) + D_\Phi(x_{i+1}, y_{i+1})
\end{aligned}$$

Plugging this into the equation above, we can see that there will be two sets of terms. The first, will be a telescoping sum much like in regular analysis, $D_\Phi(x, x_i) - D_\Phi(x, x_{i+1})$. We will now bound the second part, $D_\Phi(x_i, y_{i+1}) - D_\Phi(x_{i+1}, y_{i+1})$

$$\begin{aligned}
D_\Phi(x_i, y_{i+1}) - D_\Phi(x_{i+1}, y_{i+1}) &= \Phi(x_i) - \Phi(x_{i+1}) - \langle \nabla\Phi(y_{i+1}), x_i - x_{i+1} \rangle \\
&\leq \langle \nabla\Phi(x_i) - \nabla\Phi(y_{i+1}), x_i - x_{i+1} \rangle - \frac{\alpha}{2}\|x_i - x_{i+1}\|^2 && \text{(Strong convexity of } \Phi) \\
&\leq \langle \eta g_i, x_i - x_{i+1} \rangle - \frac{\alpha}{2}\|x_i - x_{i+1}\|^2 && \text{(From update step)} \\
&\leq \eta L\|x_{i+1} - x_i\| - \frac{\alpha}{2}\|x_i - x_{i+1}\|^2 && \text{(Lipschitzness of f)} \\
&\leq \frac{(\eta L)^2}{2\alpha}
\end{aligned}$$

Putting it all together, we can obtain,

$$\sum_{i=1}^{t}(f(x_i) - f(x)) \leq \frac{D_\Phi(x, x_1)}{\eta} + \frac{\eta L^2 t}{2\alpha}$$

□

We can now see how this technique can be used to work in a "saddle point" setting. The saddle point setting is one where we are required to optimize across two sets simultaneously. We need to find the minima with respect to one set, while finding the maxima with respect to another. Formally,

Consider two sets, $X \subset \mathbb{R}^m$ and $Y \subset \mathbb{R}^n$. We are given an objective function, $\phi(.,.)$ such that $\phi(., y)$ is convex, $\phi(x, .)$ is concave.

Find $z^* = (x^*, y^*)$ such that,

$$\phi(x^*, y^*) = \inf_{x \in X} \sup_{y \in Y} \phi(x, y) = \sup_{y \in Y} \inf_{x \in X} \phi(x, y)$$

Let us make an observation,

$$\begin{aligned}
\phi(x, y) - \phi(x', y) &\leq g_x^T (x - x') \\
-\phi(x, y) - (-\phi(x, y')) &\leq (-g_y)^T (y - y') \\
&\text{If we take } g_z = (g_x, -g_y) \\
\max_{y' \in Y} \phi(x, y') - \min_{x' \in X} \phi(x', y) &\leq g_z^T (z - z')
\end{aligned}$$

This motivates us to use `MD` to solve this problem too. Notice how similar the initial equation we have arrived at here is to the initial equation we arrived at while analyzing `MD`.

Theorem 2. *Let Φ_X, Φ_Y be mirror maps defined on both the spaces X, Y. Consider the combined map, $\Phi(x, y) = \Phi_X(x) + \Phi_Y(y)$. This works on the "combined" space z. Mirror descent with $\eta = O(\frac{1}{\sqrt{t}})$ satisfies,*

$$\max_{y \in Y} \phi(\frac{1}{t} \sum_{i=1}^{t} x_i, y) - \min_{x \in X} \phi(x, \frac{1}{t} \sum_{i=1}^{t} y_i) \leq O(\frac{1}{\sqrt{t}})$$

Proof. We will just outline how this problem reduces to the earlier simpler version of `MD`, the proof will follow in the same manner.

First, consider the norm on the combined field,

$$\begin{aligned}
\|z\|_Z &= \sqrt{\frac{\|x\|_X^2}{\kappa_X} + \frac{\|y\|_Y^2}{\kappa_Y}} \\
\|g_t\|_Z &\leq \sqrt{\frac{L_x^2}{\kappa_X} + \frac{L_Y^2}{\kappa_Y}}
\end{aligned}$$

We can proceed in the same manner,

$$\begin{aligned}
\phi(\frac{1}{t} \sum_{i=1}^{t} x_i, y) - \phi(x, \frac{1}{t} \sum_{i=1}^{t} y_i) &\leq \frac{1}{t} \sum_{i=1}^{t} \phi(x_i, y) - \phi(x, y_i) \\
&\leq \frac{1}{t} \sum_{i=1}^{t} g_i^T (z_i - z)
\end{aligned}$$

$\square$

4 An Online Primal Dual Method for Optimizing Concave Performance Measures

From the analysis of `MD` and its extension to the saddle point setting, we can see how this lends itself to our objective function in case of optimizing concave performance measures. Hence, we provide below the algorithm for optimizing concave performance measures. Note that we are using the mirror map defined by the L_2 norm when computing the Bregman divergence. This of course, recovers the standard `OGD` method as a special case of `MD`. Our method would work even in the other standard mirror maps that have been used for `MD`.

Algorithm 1: Online PrImal DUal Method (OPIUM)

$w_0 \leftarrow 0, t \leftarrow 1$
while *data stream has points* **do**
  Receive data point (x_t, y_t)
  if $y_t > 0$ **then**
    $((\alpha, \beta), w_{t+1}) \leftarrow \mathtt{MD}(\psi^*, r^+)$
  end
  else
    $((\alpha, \beta), w_{t+1}) \leftarrow \mathtt{MD}(\psi^*, r^-)$
  end
  $t \leftarrow t + 1$
end
return $\bar{w} = \frac{1}{t}\sum_{\tau=1}^{t} w_\tau$

Here, we are simply doing `MD` on the combined objective, after having cast the original problem as a saddle point optimization problem.

4.1 Convergence Analysis for OPIUM

From the convergence analysis of `MD` in the saddle point setting, we obtain,

$$\max_{y \in Y} \phi(\frac{1}{t}\sum_{i=1}^{t} x_i, y) - \min_{x \in X} \phi(x, \frac{1}{t}\sum_{i=1}^{t} y_i) \leq O(\frac{1}{\sqrt{t}})$$

Here, the combined objective is,

$$\begin{aligned} \phi(x, y) &= \phi((\alpha, \beta), w) \\ &= -\psi^*(\alpha, \beta) + r_\psi(w) \end{aligned}$$

using the `MD` analysis,

$$\begin{aligned} \sum \phi((\hat{\alpha}, \hat{\beta}), w^*) - \inf \phi((\alpha, \beta), \hat{w}) &\leq O(\sqrt{t}) \\ \sum -\psi^*(\hat{\alpha}, \hat{\beta}, w^*) + r_\psi(w^*) - \inf_{\alpha,\beta}(-\psi^*(\alpha, \beta, \hat{w}) + r_\psi(\hat{w})) &\leq O(\sqrt{t}) \\ \sum -\psi^*(\hat{\alpha}, \hat{\beta}, w^*) + r_\psi(w^*) &\leq \inf_{\alpha,\beta} \sum \big(-\psi^*(\alpha, \beta, \hat{w}) + r_\psi(\hat{w})\big) + O(\sqrt{t}) \end{aligned}$$

$$\begin{aligned} \frac{1}{T}\sum \alpha_t r^+(w^*) + \beta_t r^-(w^*) - \psi^*(\alpha_t, \beta_t) &\leq \inf \sum \alpha \frac{r^+(w_t)}{T} + \beta \frac{r^-(w_t)}{T} - \psi^*(\alpha, \beta) + O(\frac{1}{\sqrt{T}}) \\ &\leq \psi\left(\sum \frac{r^+(w_t)}{T}, \sum \frac{r^-(w_t)}{T}\right) + O(\frac{1}{\sqrt{T}}) \end{aligned}$$

The last step follows from the definition of the dual.

Now, we shall lower bound the quantity on the left side,

$$\begin{aligned} \frac{1}{T}\sum \alpha_t r^+(w^*) + \beta_t r^-(w^*) - \psi^*(\alpha_t, \beta_t) &= \hat{\alpha} r^+(w^*) + \hat{\beta} r^-(w^*) - \frac{1}{T}\psi^*(\alpha_t, \beta_t) \\ &\geq \hat{\alpha} r^+(w^*) + \hat{\beta} r^-(w^*) - \psi^*(\hat{\alpha}, \hat{\beta}) \\ &\geq \inf_{\alpha,\beta}\big(\alpha r^+(w^*) + \beta r^-(w^*) - \psi^*(\alpha, \beta)\big) \end{aligned}$$

So far, we have only used basic properties like the convexity of the dual function, and the definition of the fenchel dual. We see that we recover pretty much the exact proof of the technique described in [4], apart from the OTB conversion bounds that they applied. What we can see is that applying the mirror descent technique on the saddle point version of the problem recovers pretty much exactly the same technique as in [4].

5 Delayed Feedback

In an online learning setting, it may be possible that the feedback to the current prediction made is not available instantaneously but after some time steps. This setting is commonly recognized in the community as learning with *delayed feedback*. In this project, we also analyzed how our proposed algorithm for optimizing concave performance measure in an online setting would work when presented with delayed feedbacks.

Recently, a lot of methods have been proposed for handling the case of delayed feedback explicitly. While many works typically focused on extending specific machine learning tasks in an online setting to various delayed feedback scenarios on a case by case basis, some have proposed general methods for the online learning algorithms under delayed feedbacks. In this project, we consider one such work which propose a unified theoretical framework for analyzing full-information online learning algorithms under delayed feedback[2]. In the subsections, we will first describe their algorithm and then we will analyze our algorithm when presented with their method under delayed feedback.

The problem setting and the learners for Delayed-feedback Online Convex Optimization is provided in Algo. 2:

Algorithm 2: Delayed-feedback Online Convex Optimization

The enviroment chooses a sequence of convex loss functions $f_1, \dots, f_T \in \mathcal{F}$.
for *each time step* $t = 1, 2, \dots, T$ **do**
  The learner makes a prediction $x_t \in \mathcal{X}$
  The learner incurs a loss $f_t(x_t)$ and receives the set of feedbacks $H_t = \{f_{t'} : t' + \tau_{t'} = t\}$ where $\tau_{t'}$ is the delay in the feedback of time step t'
end
Goal: minimize $\sup_{x \in \mathcal{X}} R_T(x)$

5.1 Single Instance Black-box Reduction[2]

Suppose we have a deterministic online learning algorithm for non-delayed setting (call it `BASE`). Without doing any modification in `BASE`, at every time step, `BASE` is only fed with the feedback that has already arrived. In simpler terms, prediction at every time step is made only using the feedback that has already arrived till that time step and whenever feedback is arrived `BASE` is updated. This scheme is called `SOLID` "Single-instance Online Learning In Delayed environments". The complete algorithm from [2] is shown in Algo 3.

Algorithm 3: Single-Instance Online Learning In Delayed environments (SOLID) [2]

Set $x \leftarrow$ first prediction of BASE
for *each time step* $t = 1, 2, \dots$ **do**
  Set $x_t \leftarrow x$ as the prediction for the current time step.
  Receive the set of feedback H_t that arrives at the end of time step t.
  for *each* $f_s \in H_t$ **do**
    Update BASE with f_s.
    $x \leftarrow$ the next prediction of BASE.
  end
end

5.2 Providing OPIUM as BASE in SOLID

We will now look into the assumption and setting that `SOLID` assumes for the `BASE` algorithm for it to work and verify that `OPIUM` satisfies those assumptions. The assumptions and settings of `SOLID` are as follows:

1. `BASE` solves an **online convex optimization problem** under non-delayed feedback. It is **not** necessary for the objective function to be either strongly convex or smooth. Also, the gradients of the objective function can be **sparse**.
2. The predictions are made over a closed, convex, non-empty subset of $\mathcal{X}$ of a Hilbert space $\mathbb{X}$ over the reals. The function $\Phi : S \mapsto \mathbb{R}$ which induces the Bregman Divergence $D_\Phi(x, y)$ is an α **strongly convex** (w.r.t. some norm $\|\cdot\|$), **differentiable** and **non-negative function** over a convex closed set $S \subset \mathbb{X}$ with a non-empty interior S°.
3. The adversary provides a **convex** loss function $f_t \in \mathcal{F} \subset \{f : \mathcal{X} \mapsto \mathbb{R}\}$ at each time $t > 0, t \in \mathbb{Z}$. The sequence of loss functions can be chosen in an **adversarial manner**. (Note: Though f_t is assigned after each prediction at t, it is provided to the learner after a delay).
4. The delays $\{\tau_t\}_{t=1}^T$ can be **unbounded** and they can **reorder the feedbacks**, i.e., the feedback f_t of the interaction at time step $t < t'$ might arrive after feedback $f_{t'}$

The settings of our proposed algorithm `OPIUM` are similar and the assumptions needed are also satisfied.

5.3 Convergence Analysis

Considering the aforementioned settings and assumptions `SOLID` makes for the `BASE` algorithm, Joulani et al. analyzed the Mirror-Descent class of Online Convex Optimization algorithms and a regret bound under delayed feedback setting is provided. Since, our algorithm principally applying Mirror-Descent for saddle point optimization, we can directly apply the regret bounds proved in [2] for Mirror-Descent class of OCO algorithms.

The following theorem provides the regret bound for the Mirror-Descent class of OCO algorithms under delayed feedback setting:

Theorem 3. *Suppose all the aforementioned assumptions hold and we ran `SOLID` in a delayed-feedback environment. Let $r_t, t = 0, 1, \ldots, T$ denotes the regularizers (in case of MD, regularizers are the functions which induce Bregman Divergence) that `BASE` uses in its simulated non-delayed run inside `SOLID`, and let $\|\cdot\|_{(t)}$ denote the associated strong-convexity norms. Let R_T denote the regret of `SOLID` in its delayed-feedback environment.*

If BASE is an Mirror Descent Online convex Optimization algorithm, then

$$R_T \leq \sum_{t=1}^{T} D_{\tilde{r}_t}(x, \tilde{x_s}) + \frac{1}{2}\sum_{t=1}^{T} \|\tilde{f}_t'(\tilde{x_t}\|_{(t),*}^2 + \sum_{t=1}^{T}\sum_{j=t-\tilde{\tau}_t}^{t-1} \|\tilde{f}_t'(\tilde{x_j}\|_{(j),*}^2 \|\tilde{f}_j'(\widetilde{x_{j+1}}\|_{(j),*}^2 \tag{1}$$

The bound provided in the equation 1 seems somewhat unwieldy. Joulani et al. simplifies the above bound to get a more indicative result.

$$R_T \leq \frac{2R^2}{\tilde{\eta}_t} + \frac{G^2}{2}\sum_{t=1}^{T} \tilde{\eta}_t \left(1 + 2\tilde{\tau}_t\right)$$

Where R is constant such that $\tilde{\eta}_T \sum_{t=1}^{n} D_\Phi(x^*, x_t) \leq 2R^2$, G is the upper bound on the norm of the gradient, $\tilde{\tau}_t$ is difference between the time of t_{th} prediction by the BASE and the time of feedback received for that prediction.

The final regret in terms of order of number of time steps can be written as $\mathcal{O}(\sqrt{T + 2\mathcal{T}})$ where $\mathcal{T} = \sum_{t=1}^{T} \tilde{\tau}_t$.

6 Conclusions

In this project, we propose an algorithm OPIUM for optimizing concave performance measures in an online learning setting where the loss incurred by the learner at each time step can be adversarial in nature. We extend the work by Narasmhan et al. [4], who proposed an algorithm for learning non-concave performance measures in a stochastic learning setting, using Mirror Descent algorithm for the parameters update. We analyze the proposed algorithm OPIUM and obtain similar sublinear regret bound as of the stochastic setting in [4]. We also analyzed our algorithm under delayed feedback using the general black-box algorithm proposed by [2] which takes a non-delayed feedback OCO algorithm and run it in delayed feedback setting.

7 Future Work

In this project the concave performance measure we looked at had the Lipschitz property in the link function which is necessary for Mirror Descent to work. As a future work we want to explore ways to optimize concave performance measures with Non-Lipschitz link functions. It's evident that direct application of Mirror descent would not work on these measure we might have to change the approach all together to work on them. We haven't done any experimental verification of our methods hence, it's necessary that we set up experiments to test the performance of our methods in adversarial settings. For that, we might even have to set up an adversary.

Lastly, we want to eplore that if there is a possibility of getting faster rates of convergence by introducing smoothness/strongly convex surrogates for these rewards.

References

[1] Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[2] Pooria Joulani, András György, and Csaba Szepesvári. Delay-tolerant online convex optimization: Unified analysis and adaptive-gradient algorithms. In *AAAI*, volume 16, pages 1744–1750, 2016.

[3] Purushottam Kar, Harikrishna Narasimhan, and Prateek Jain. Online and stochastic gradient methods for non-decomposable loss functions. In *Advances in Neural Information Processing Systems*, pages 694–702, 2014.

[4] Harikrishna Narasimhan, Purushottam Kar, and Prateek Jain. Optimizing non-decomposable performance measures: A tale of two classes. In *Proceedings of the International Conference in Machine Learning*, 2015.

[5] Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Beyond regret. In *COLT*, pages 559–594, 2011.