Research Project
Efficiency of Primal-Dual Hybrid Gradient method

Rustem Islamov* Olivier Fercoq †

October 31, 2021

Abstract

We study the empirical performance of Primal-Dual Hybrid Gradient (PDHG) method on different class of optimization problems arising in machine learning and other areas. We demonstrate numerical superiority of PDHG in various scenarios. Besides, we compare the performance of PDHG with other gradient type methods.

Contents

*Institut Polytechnique de Paris, Palaiseau, France
†Télécom Paris, Institut Polytechnique de Paris, Palaiseau, France.

1 Introduction

Primal-dual methods are widely used for solving optimization problems with constraints. Encoding such nonsmooth constrains we replace original problem with the problem of finding saddle points of the Lagrangian. More precisely we consider the problem of the form

$$\min_{x\in\mathcal{X}}\Big[f(x)+f_2(x)+g\square g_2(Ax)\Big]. \tag{1}$$

Here f and g are convex functions for which the proximal operators are easily computable; $A:\mathcal{X}\to\mathcal{Y}$ is a linear operator. We assume that we have an access to g^* and g_2^* Fenchel-Legendre conjugates of g and g_2 respectively. Finally, f_2 and g_2^* are convex functions with L_f and L_{g^*} Lipschitz gradients. As it was stated before, we consider primal-dual methods which are searching for a saddle point of of the Lagrangian, which has the following form

$$L(x,y)=f(x)+f_2(x)+\langle Ax,y\rangle-g^*(y)-g_2^*(y). \tag{2}$$

The point (x^*,y^*) is called a saddle point for the Lagrangian (2) if it satisfies

$$L(x,y^*)\le L(x^*,y^*)\le L(x^*,y)\quad\forall x\in\mathcal{X},y\in\mathcal{Y}. \tag{3}$$

We assume throughout the paper that at least one saddle point exists. It can be guaranteed using Slater's constraint qualification condition.

There exist different ways to measure the convergence of primal-dual algorithms: duality gap, Karush-Kuhn-Tacker (KKT) error, metrical subregularity [Rockafellar and Wets, 1998], smoothed gap [Tran-Dinh et al., 2018]. Recently Quadratic Error Bound has been introduced in [Fercoq, 2021] which properly reflect the behaviour of PDHG. This regularity assumption holds for a wide range of problems such as strongly convex-concave problem or linear programming.

2 Preliminaries

In this section we introduce necessary definitions and notation which will be used throughout the paper.

2.1 Notation

We denote $\mathcal{X}$ the primal space, $\mathcal{Y}$ the dual space. The proximal operator of a function f is given by $\operatorname{prox}_f(x')=\arg\min_{x'}\left[f(x')+\frac{1}{2}\|x-x'\|^2\right]$. We will use the indicator function ι_C of a convex set C which is defined as follows

$$\iota_C:=\begin{cases}0 & \text{if } x\in C\\ +\infty & \text{if } x\notin C.\end{cases}$$

Besides, Fenchel-Legendre conjugate f^* of a function f is defined by

$$f^*(y)=\sup_{x\in\mathcal{X}}\left[\langle x,y\rangle-f(x)\right].$$

2.2 Definitions

First we define the epigraph of a function f.

Definition 2.1. *Let $f:\mathcal{X}\to\mathbb{R}\cup\{+\infty\}$. The epigraph of f, denoted by epi f, is the subset of $\mathcal{X}\times\mathbb{R}$ defined by*

$$epi\ f=\{(x,t)\in\mathcal{X}\times\mathbb{R}:t\ge f(x)\}.$$

Knowing what the epigraph is, we can define the definition of convex function.

Definition 2.2. *A function $f : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$ is convex if its epigraph is a convex set.*

More restricted class of convex functions is a class of so called strongly-convex functions.

Definition 2.3. *A function $f : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$ is μ-strongly convex if $f - \frac{\mu}{2}\|x\|^2$ is convex.*

Now we define what strongly convex-concave Lagrangian function means.

Definition 2.4. *The Lagrangian function is μ-strongly convex-concave, if $x \mapsto L(x, y)$ is μ-strongly convex for all y, and $y \mapsto -L(x, y)$ is μ-strongly convex for all x.*

Moreover, we will work with L-smooth functions.

Definition 2.5. *Let $f : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$ is L-smooth, if it is continuously differentiable and $\|\nabla f(x) - \nabla f(x')\| \leq L\|x - x'\|$ for all x, x'.*

3 Brief description of existing methods

Now we describe the methods we will compare later in numerical experiments.

3.1 Primal-Dual Hybrid Gradient

We start with Primal-Dual Hybrid Gradient (PDHG) method which is defined by Algorithm 1. This algorithm was designed to find a saddle point of the Lagrangian (2) (see [Fercoq, 2021] for more detailed description and convergence theory of the method under different optimality measures). Quadratic Error Bound (QEB) was introduced in [Fercoq, 2021] which unifies existing optimilaty measures for PDHG analysis like strong convexity [Chambolle and Pock, 2011] and metrical subregularity [Liang et al., 2016]. [Fercoq, 2021] shows linear convergence of PDHG under Quadratic Error Bound regularity condition.

Algorithm 1 Primal-Dual Hybrid Gradient (PDHG)

Parameters: stepsizes τ, σ
Initialization: $x_0 \in \mathcal{X}, y_0 \in \mathcal{Y}$
for $k = 0, 1, 2, \ldots, K$ **do**
  $\bar{x}_{k+1} = \text{prox}_{\tau f}\left(x_k - \tau \nabla f_2(x^k) - \tau A^\top y_k\right)$
  $\bar{y}_{k+1} = \text{prox}_{\sigma g^*}\left(y_k - \sigma \nabla f_2(y_k) + \sigma A \bar{x}_{k+1}\right)$
  $x_{k+1} = \bar{x}_{k+1} - \tau A^\top (\bar{y}_{k+1} - y_k)$
  $y_{k+1} = \bar{y}_{k+1}$
end for
return (x_K, y_K)

3.2 Restarted Averaged Primal-Dual Hybrid Gradient

One of the popular techniques to make the performance of a certain method better is to average all iterates and issue this mean as the output of averaged method. This technique could be applied on PDHG which lead to Averaged PDHG (Algorithm 2). [Fercoq, 2021] suggests to restart APDHG to get even better performance (see Algorithm 3). Under certain assumptions (for example, Quadratic Error Bound) restarted ADPHG coverges linearly (see Proposition 13 from [Fercoq, 2021]).

Algorithm 2 Averaged Primal-Dual Hybrid Gradient (APDHG)

Parameters: stepsizes τ, σ
Initialization: $x_0 \in \mathcal{X}, y_0 \in \mathcal{Y}$
for $k = 0, 1, 2, \ldots, K-1$ **do**
  $\bar{x}_{k+1} = \text{prox}_{\tau f}\left(x_k - \tau \nabla f_2(x^k) - \tau A^\top y_k\right)$
  $\bar{y}_{k+1} = \text{prox}_{\sigma g^*}\left(y_k - \sigma \nabla f_2(y_k) + \sigma A \bar{x}_{k+1}\right)$
  $x_{k+1} = \bar{x}_{k+1} - \tau A^\top (\bar{y}_{k+1} - y_k)$
  $y_{k+1} = \bar{y}_{k+1}$
end for
$\tilde{x}_K = \frac{1}{K} \sum_{l=0}^{K-1} \bar{x}_{l+1} \quad \tilde{y}_K = \frac{1}{K} \sum_{l=0}^{K-1} \bar{y}_{l+1}$
return $(\tilde{x}_K, \tilde{y}_K)$

Algorithm 3 restarted Averaged Primal-Dual Hybrid Gradient (rAPDHG)

Parameters: stepsizes $\tau, \beta_k, t_0 \in \mathbb{R}$, integer period K
Initialization: $x_0 \in \mathcal{X}, y_0 \in \mathcal{Y}$
for $t = 0, 1, 2, \ldots, T-1$ **do**
  $x_{(t+1)K} = \text{APDHG}(x_{tK}, K)$ run APDHG for K iterations starting from x_{tK}
end for
return x_{TK}

3.3 FISTA and its extensions

Next, we consider Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [Beck and Teboulle, 2009]. This method was designed to solve problems of the form (so called composite problem)

$$\min_{x \in \mathcal{X}} \Big[f(x) + g(x) \Big],$$

where g is convex l.s.c. function with easily computable proximal operator, and f is a convex function with Lipschitz gradient. [Beck and Teboulle, 2009] shows optimal $\mathcal{O}(\frac{1}{k^2})$ convergence of FISTA. Later [Fercoq and Qu, 2019] analyses restarted version of FISTA (rFISTA) and proves optimal linear convergence. Besides, [Chambolle and Pock, 2016] considers strongly convex case of the composite problem, and proves optimal linear convergence of FISTA-PC (modification of FISTA for strongly convex composite problem). However, we would like to note that rFISTA is provably works for more wide class of problems satisfying QEB. All three methods are presented by Algorithms 4 and 5. For FISTA-PC we use constants μ_f, μ_g for strong convexity parameters of f and g respectively.

We would like to point out that FISTA and restarted FISTA provably converge for much wider class of functions (satisfying Quadratic Error Bound) than FISTA-PC that converges only in strongly convex case. Besides, in practice it is much difficult to estimate strong convexity parameter than the Lipschitz constant of the gradient. Thus it could be complicated to find best parameters for appropriate performance of FISTA-PC rather for rFISTA.

3.4 Primal-dual method with random extrapolation and coordinate descent

In this section we investigate Primal-dual method with random extrapolation and coordinate descent (PURE-CD) [Alacaoglu et al., 2020]. This method was created to solve problems of the form

$$\min_{x \in \mathcal{X}} \Big[f(x) + g(x) + h(Ax) \Big],$$

Algorithm 4 Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) and FISTA-PC

Parameters: stepsizes $\tau, \beta_k, t_0 \in \mathbb{R}$, $q = \tau\mu / 1+\tau\mu_g$
Initialization: $x_0 = x_{-1} \in \mathcal{X}$
for $k = 0, 1, 2, \ldots, K-1$ **do**
  $y_k = x_k + \beta_k (x_k - x_{k-1})$
  $x_{k+1} = \operatorname{prox}_{\tau g}(y_k - \tau \nabla f(y_k))$
  $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}, \quad \beta_{k+1} = \frac{t_k - 1}{t_{k+1}}$ FISTA update
  $t_{k+1} = \frac{1-qt_k^2+\sqrt{(1-qt_k^2)^2+4t_k^2}}{2}, \quad \beta_{k+1} = \frac{t_k-1}{t_{k+1}} \frac{1+\tau\mu_g - t_{k+1}\tau\mu}{1-\tau\mu_f}$ FISTA-PC update
end for
return x_K

Algorithm 5 restarted Fast Iterative Shrinkage-Thresholding Algorithm (rFISTA)

Parameters: stepsizes $\tau, \beta_k, t_0 \in \mathbb{R}$, integer period K
Initialization: $x_0 = x_{-1} \in \mathcal{X}$
for $t = 0, 1, 2, \ldots, T-1$ **do**
  $x_{(t+1)K} = \text{FISTA}(x_{tK}, K)$ run FISTA for K iterations starting from x_{tK}
end for
return x_{TK}

where $f, g : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$ and $h : \mathcal{Y} \to \mathbb{R} \cup \{+\infty\}$ are proper, lower semicontinuous, convex functions, $A : \mathcal{X} \to \mathcal{Y}$ is a linear operator. We assume that Euclidean spaces $\mathcal{X}$ and $\mathcal{Y}$ can be represented as $\mathcal{X} = \prod_{i=1}^n \mathcal{X}_i$ and $\mathcal{Y} = \prod_{j=1}^m \mathcal{Y}_j$. In the simplest case, when $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathbb{R}^m$, this is obvouisly true. Moreover, f is assumed to have coordinatewise Lipschitz continuous gradients and g, h admit easily computable proximal operators. PURE-CD as PDHG works in $\mathcal{X} \times \mathcal{Y}$. The pseudocode of PURE-CD is represented in Algorithm 6.

In this section we use additional notation. We consider the proximal operator of a function g with respect to positive semidefinite matrix V

$$\operatorname{prox}_{V,g}(x) = \arg\min_{x'} \left[g(x') + \frac{1}{2} \left\| x' - x \right\|_{V^{-1}}^2 \right].$$

Here $\|u\|_A$ is defined as follows

$$\|u\|_A = \sqrt{\langle Au, u \rangle},$$

where A is a positive definite matrix. In case of diagonal matrix V each diagonal entire of V is used as a stepsize for corresponding coordinate. Moreover, we are required to define the following notation

$$J(i) = \{j \in [m] : A_{ji} \neq 0\}.$$

The Lagrangian function of this problem has the form

$$L(x, y) = f(x) + g(x) + \langle Ax, y \rangle - h^*(y).$$

4 Optimization problems

4.1 Ridge regression

The first problem we consider is Ridge regression. The problem has the following form

$$\min_{x \in \mathcal{X}} \left[\frac{1}{2} \|Ax - b\|^2 + \frac{\lambda}{2} \|x\|^2 \right],$$

Algorithm 6 Primal-Dual method with Random Extrapolation and Coordinate Descent (PURE-CD)

Parameters: Diagonal matrices $\theta, \tau, \sigma > 0$
Initialization: $x_0 \in \mathcal{X}, y_0 \in \mathcal{Y}$
for $k = 0, 1, 2, ...K-1$ **do**
  $\bar{y}_{k+1} = \text{prox}_{\sigma,h^*}\left(y_k + \sigma A x_k\right)$
  $\bar{x}_{k+1} = \text{prox}_{\tau,g}\left(x_k - \tau\left(\nabla f(x_k) + A^\top \bar{y}_{k+1}\right)\right)$
  sample $i_{k+1} \in [n]$ with $\mathbb{P}\left(i_{k+1} = i\right) = p_i$
  $x_{k+1}^{i_{k+1}} = \bar{x}_{k+1}^{i_{k+1}}$
  $x_{k+1}^{j} = x_k^j \ \forall j \neq i_{k+1}$
  $y_{k+1}^{j} = \bar{y}_{k+1}^{j} + \sigma_j \theta_j \left(A(x_{k+1} - x_k)\right)_j \ \ \forall j \in J(i_{k+1})$
  $y_{k+1}^{j} = y_k^j \ \forall j \notin J(i_{k+1})$
end for
return (x_K, y_K)

where λ is positive regularization parameter. Such regularization is used when linear system $Ax = b$ has infinite number of solutions. We choose the solution with the smallest norm. Usually ℓ_2 is applied if data suffers from multicollinearity. Referring to (1), we set $g(Ax) = \frac{1}{2}\left\|Ax - b\right\|^2$, i.e. $g(z) = \frac{1}{2}\left\|z - b\right\|^2$, and $f(x) = \frac{\lambda}{2}\|x\|^2$. Other functions are zero.

First, we need to find g^*.

Lemma 4.1. *The Fenchel-Legendre conjugate of $g = \frac{1}{2}\left\|x - b\right\|^2$ is given by*

$$g^*(y) = \frac{1}{2}\|y\|^2 + \langle y, b\rangle .$$

Proof. We write the definition of the Fenchel-Legendre conjugate

$$g^*(y) = \sup_x \left[\langle y, x\rangle - \frac{1}{2}\left\|x - b\right\|^2\right].$$

This is the strongly convex problem, thus the solution is unique. By the Fermat rule we get

$$y - x + b = 0 \Rightarrow x = y + b.$$

Finally, putting the above into the definition of Fenchel-Legendre conjugate we derive

$$\begin{aligned} g^*(y) &+ \langle y, y + b\rangle - \frac{1}{2}\left\|y + b - b\right\|^2 \\ &= \frac{1}{2}\|y\|^2 + \langle y, b\rangle . \end{aligned}$$

□

Now we need to find the explicit form of proximal operators for f and g^*.

Lemma 4.2. *The proximal operator of τf, where $f = \frac{\lambda}{2}\|x\|^2$, is given by*

$$\text{prox}_{\tau f}(x) = \frac{x}{1 + \tau\lambda}.$$

Proof. We write the definition of a proximal operator

$$\text{prox}_{\tau f}(x) = \arg\min_{x'} \left[\frac{\tau\lambda}{2}\|x'\|^2 + \frac{1}{2}\left\|x' - x\right\|^2\right].$$

By the Fermat rule we obtain

$$\tau\lambda x' + x' - x = 0 \Rightarrow x' = \frac{x}{1 + \tau\lambda}.$$

□

Lemma 4.3. *The proximal operator of σg^*, where $g^*(y) = \frac{1}{2}\|y\|^2 + \langle y, b\rangle$, is given by*

$$\text{prox}_{\sigma g^*}(x) = \frac{x - \sigma b}{1 + \sigma}.$$

Proof. We write the definition of a proximal operator

$$\text{prox}_{\sigma g^*}(x) = \arg\min_{x'} \left[\frac{\sigma}{2}\|x'\|^2 + \sigma \left\langle x', b \right\rangle + \frac{1}{2}\left\|x' - x\right\|^2 \right].$$

By the Fermat rule we get

$$\sigma x' + \sigma b + x' - x = 0 \Rightarrow x' = \frac{x - \sigma b}{1 + \sigma}.$$

□

The Lagrangian function of Ridge regression problem has the following problem

$$L(x, y) = \frac{\lambda}{2}\|x\|^2 + \langle Ax, y\rangle - \frac{1}{2}\|y\|^2 - \langle y, b\rangle .$$

It is λ-strongly convex in x and 1-strongly concave in y. Finally, f is L-smooth with $L = \lambda_{\max}(A^\top A)$, where $\lambda_{\max}(M)$ denotes the largest eigenvalue of M.

4.2 Elastic net regression

Now we consider Elastic net regression problem of the form

$$\min_{x \in \mathcal{X}} \left[\frac{1}{2}\left\|Ax - b\right\|^2 + \lambda_1 \|x\|_1 + \frac{\lambda_2}{2}\|x\|^2 \right],$$

where λ_1, λ_2 are positive regularization constants. Use of ℓ_2 regularization has the same meaning as for Ridge regression. Besides, we also add thresholding via ℓ_1 regularization. This means that we select only important features corresponding to large values of x and throw away others. In this case we use the following notation: $f(x) = \lambda_1\|x\|_1$, $f_2(x) = \frac{\lambda_2}{2}\|x\|^2$, and $g(z) = \frac{1}{2}\|z - b\|^2$, i.e. $g(Ax) = \frac{1}{2}\left\|Ax - b\right\|^2$.

We already now the explicit form of the Fenchel-Legendre conjugate of g (see Lemma 4.1). Moreover, we don't have to use proximal operator of f_2, but f_2 is L-smooth with $L = \lambda_2$. The only thing that is still unknown is the proximal operator of $f = \|\cdot\|_1$.

Lemma 4.4. *The proximal operator of $\tau f(x)$, where $f(x) = \lambda_1\|x\|_1$, is given by*

$$\left[\text{prox}_{\tau f}(x)\right]_i = \begin{cases} x_i - \tau\lambda_1 & \text{if } x_i > \tau\lambda_1 \\ 0 & \text{if } x_i \in [-\tau\lambda_1, \tau\lambda_1] \\ x_i + \tau\lambda_1 & \text{if } x_i < -\tau\lambda_1 \end{cases}.$$

Proof. Recall that the subdifferntial of $|x|$ can be given in the following way

$$\partial|x| = \begin{cases} 1 & \text{if } x > 0 \\ [-1, 1] & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}.$$

Now we write the definition of proximal operator of τf

$$\text{prox}_{\tau f}(x) = \arg\min_{x'} \left[\tau\lambda_1\|x'\|_1 + \frac{1}{2}\left\|x' - x\right\|^2 \right].$$

Note, that the subproblem inside $\arg\min$ is separable, thus the solution can be found for each component separately

$$\left[\text{prox}_{\tau f}(x)\right]_i = \arg\min_{x_i'} \left[\tau\lambda_1|x_i'| + \frac{1}{2}(x_i' - x_i)^2\right].$$

Since both functions in $\arg\min$ are convex with domain $\mathbb{R}$, then the Fermat rule can be written as follows

$$0 \in \tau\lambda_1 \partial|x_i'| + x_i' - x_i.$$

Now we consider all three cases. If $x_i' > 0$, then

$$0 = \tau\lambda_1 + x_i' - x_i \Rightarrow x_i' = x_i - \tau\lambda_1.$$

We get that this case could be realized if $x_i > \tau\lambda_1$. Now let be $x_i < 0$, then

$$0 = -\tau\lambda_1 + x_i' - x_i \Rightarrow x_i' = x_i + \tau\lambda_1.$$

This case is realized, if $x_i < -\tau\lambda_1$. Finally, if $x_i = 0$, then we obtain

$$0 \in [-\tau\lambda_1, \tau\lambda_1] + 0 - x_i \Rightarrow x_i \in [-\tau\lambda_1, \tau\lambda_1].$$

Combining all the above we derive

$$x_i' = \begin{cases} x_i - \tau\lambda_1 & \text{if } x_i > \tau\lambda_1 \\ 0 & \text{if } x_i \in [-\tau\lambda_1, \tau\lambda_1] \\ x_i + \tau\lambda_1 & \text{if } x_i < -\tau\lambda_1 \end{cases},$$

that concludes the proof. □

Note that the result above can be written as follows:

$$\text{prox}_{\tau\lambda_1\|x\|_1}(x) = \text{sign}(x)\max\{|x| - \tau\lambda_1, \mathbf{0}\},$$

where all functions work element-wise. For example, for $x \in \mathbb{R}^d$ we have

$$\begin{aligned} &\text{sign} : \mathbb{R}^d \to \mathbb{R}^d, \quad [\text{sign}(x)]_i = \text{sign}(x_i)\ \forall i \in [d], \\ &|\cdot| : \mathbb{R}^d \to \mathbb{R}^d, \quad [|x|]_i = |x_i|, \\ &\max : \mathbb{R}^d \to \mathbb{R}^d, \quad [\max\{x, \mathbf{0}\}]_i = \max\{x_i, 0\}, \end{aligned}$$

where $\mathbf{0}$ is a vector of zeros. Such explicit form allows efficient implementation of this proximal operator in practice.

Finally, we write the explicit of form of the Lagrangian function of this problem

$$L(x, y) = \lambda_1\|x\|_1 + \frac{\lambda_2}{2}\|x\|^2 + \langle Ax, y\rangle - \frac{1}{2}\|y\|^2 - \langle y, b\rangle.$$

This function is λ_2-strongly convex in x and 1-strongly concave in y.

4.3 Logistic regression

Next, we study one of the most popular models in classic machine learning, which is Logistic regression model for binary classification. This linear classifier returnes the probability that the object belongs to a certain class. Let $\{(x_i, y_i)\}_{i=1}^N$ be a data set, where each pair consists of a feature vector $x_i \in \mathbb{R}^d$ and class label $y_i \in \{-1, 1\}$. We consider ℓ_2 regularized problem

$$\min_w \left[\sum_{i=1}^N \log\left(1 + \exp(-y_i x_i^\top w)\right) + \frac{\lambda}{2}\|x\|^2 \right],$$

where $w \in \mathbb{R}^d$ denotes the parameters of the model. Now we define a function of the form

$$g(z) = \sum_{i=1}^N \log\left(1 + \exp(-y_i z_i)\right) = \sum_{i=1}^N h(y_i z_i), \quad z \in \mathbb{R}^N, \tag{4}$$

where $h(t) = \log\left(1 + \exp(-t)\right)$. Using g we can write the main part in the original problem as $g(Xw)$, where

$$X = \begin{pmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_N^\top \end{pmatrix} \in \mathbb{R}^{N \times d}$$

is a feature matrix. We define the regularization term as f, and we already know the explicit form of proximal operator of τf. Now we need to find the Fenchel-Legendre conjugate of g.

Lemma 4.5. *The Fenchel-Legendre conjugate of g which is defined in (4) has the following form*

$$ads$$

Proof. We write the definition

$$g^*(\varphi) = \sup_z \left[\langle \varphi, z \rangle - \sum_{i=1}^N \log\left(1 + \exp(-y_i z_i)\right) \right].$$

The subproblem inside sup is separable, hence we can solve it componentwise. The i-th component has the form as follows

$$g_i^*(\varphi_i) = \sup_{z_i} \left[\varphi_i z_i - \log\left(1 + \exp(-y_i z_i)\right) \right],$$

and $g^*(\varphi) = \sum_{i=1}^N g_i^*(\varphi_i)$. Let consider the first case $y_i = -1$, then

$$g_i^*(\varphi_i) = \sup_{z_i} \left[\varphi_i z_i - \log\left(1 + \exp(z_i)\right) \right].$$

Writing the Fermat rule we obtain

$$\varphi_i - \frac{e^{z_i}}{1 + e^{z_i}} = 0 \Rightarrow \varphi_i = \frac{e^{z_i}}{1 + e^{z_i}}.$$

We clearly see that g_i^* is well defined for $\varphi_i \in (0, 1)$. Taking log of both sides we get

$$\log \varphi_i = z_i - \log(1 + e^{z_i}). \tag{5}$$

Besides, we have

$$1 - \varphi_i = 1 - \frac{e^{z_i}}{1 + e^{z_i}} = \frac{1}{1 + e^{z_i}},$$

thus we also get

$$\log(1-\varphi_i) = -\log(1+e^{z_i}). \tag{6}$$

Subtracting (6) from (5) we obtain

$$\log\varphi_i - \log(1-\varphi_i) = z_i.$$

Combining all the above we finally derive

$$\begin{aligned} g_i^*(\varphi_i) &= \varphi_i z_i - \log\left(1+\exp(z_i)\right) \\ &= \varphi_i\left(\log(\varphi_i) - \log(1-\varphi_i)\right) - \log\left(1+\frac{\varphi_i}{1-\varphi_i}\right) \\ &= \varphi_i\log\varphi_i + (1-\varphi_i)\log(1-\varphi_i). \end{aligned}$$

Now assume that $y_i = 1$, then

$$g_i^*(\varphi_i) = \sup_{z_i}\left[\varphi_i z_i - \log\left(1+\exp(-z_i)\right)\right].$$

Writing the Fermat rule again we get

$$\varphi_i + \frac{e^{-z_i}}{1+e^{-z_i}} = 0 \Rightarrow -\varphi_i = \frac{e^{-z_i}}{1+e^{-z_i}}.$$

Taking log of both sides we get

$$\log(-\varphi_i) = -z_i - \log(1+e^{-z_i}). \tag{7}$$

Besides, we have

$$1+\varphi_i = 1 - \frac{e^{-z_i}}{1+e^{-z_i}} = \frac{1}{1+e^{-z_i}}.$$

So we also have the equality

$$\log(1+\varphi_i) = -\log(1+e^{-z_i}). \tag{8}$$

Subtracting (7) from (8) we get

$$\log(1+\varphi_i) - \log(-\varphi_i) = z_i.$$

Finally, we have

$$\begin{aligned} g_i^*(\varphi_i) &= \varphi_i z_i - \log(1+\exp(-z_i)) \\ &= \varphi_i\left(\log(1+\varphi_i) - \log(-\varphi_i)\right) - \log\left(1+\frac{-\varphi_i}{1+\varphi_i}\right) \\ &= (1+\varphi_i)\log(1+\varphi_i) - \varphi_i\log(-\varphi_i). \end{aligned}$$

One formula that combines both cases has the form

$$g_i^*(\varphi_i) = -y_i\varphi_i\log\left(-y_i\varphi_i\right) + (1+y_i\varphi_i)\log(1+y_i\varphi_i).$$

The whole conjugate of g has the following form

$$g^*(\varphi_i) = \sum_{i=1}^{N} -y_i\varphi_i\log\left(-y_i\varphi_i\right) + (1+y_i\varphi_i)\log(1+y_i\varphi_i). \tag{9}$$

□

Now we need to find a proximal operator of g^*.

Lemma 4.6. *The proximal operator of a function g, which has the form* (9)*, can be written as*

ss

Proof. We again begin with writing the definition of a proximal operator of g^*

$$\operatorname{prox}_{\sigma g^*}(\varphi) = \arg\min_{\varphi'} \left[\sum_{i=1}^{N} -y_i\varphi_i' \log\left(-y_i\varphi_i'\right) + (1+y_i\varphi_i')\log(1+y_i\varphi_i') + \frac{1}{2}\left\|\varphi'-\varphi\right\|^2 \right].$$

This optimization problem is separable, thus we consider only one component

$$\arg\min_{\varphi_i'} \left[-y_i\varphi_i' \log\left(-y_i\varphi_i'\right) + (1+y_i\varphi_i')\log(1+y_i\varphi_i') + \frac{1}{2}(\varphi_i'-\varphi_i)^2 \right].$$

Writing the Fermat rule we have

$$\begin{aligned} 0 &= -y_i(1+\log(-y_i\varphi_i')) + y_i(1+\log(1+y_i\varphi_i')) + \varphi_i' - \varphi_i \\ &= -y_i\log(-y_i\varphi_i') + y_i\log(1+y_i\varphi_i') + \varphi_i' - \varphi_i. \end{aligned}$$

□

4.4 Support vector machine

The objective of the support vector machine algorithm is to find an optimal hyperplane in an d-dimensional space, where d is the number of features, that distinctly classifies the data points. The optimilaty is characterized by the fact that the distance from the closest point to a hyperplane is the largest. Let $\{(x_i, y_i)\}_{i=1}^N$ be a data set, where $x_i \in \mathbb{R}^d$ is a feature vector, and $y_i \in \{-1, 1\}$ is a class label. From mathematical point of view, ℓ_1 regularized SVM problem can be formulated as follows

$$\min_w \left[\sum_{i=1}^{N} \max\left(0, 1 - x_i y_i^\top w\right) + \lambda\|w\|_1 \right].$$

We set $f(w) = \lambda\|w\|_1$, and $g\left(D(y)Xw\right)$, where

$$g(z) = \sum_{i=1}^{N} \max(0, 1-z_i) \quad \forall z \in \mathbb{R}^N,$$

and

$$X = \begin{pmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_N^\top \end{pmatrix} \in \mathbb{R}^{N\times d}, \quad D(b) = \begin{pmatrix} y_1 & 0 & \dots & 0 \\ 0 & y_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & y_N \end{pmatrix} \in \mathbb{R}^{N\times N}.$$

Proximal operator of τf was already found in Lemma 4.4. Now we need to find the Fenchel-Legendre conjugate of g.

Lemma 4.7. *The Fenchel-Legendre of g has the following form*

$$g^*(\varphi) = \langle \varphi, \mathbf{1}\rangle + \iota_{[-1,0]^N}(\varphi)$$

Proof. First, note that $g(z) = \sum\limits_{i=1}^{N} \max(0, 1-z_i) = \sum\limits_{i=1}^{N} l(z_i)$. Then we can write the Fenchel-Legendre conjugate of g via

$$g^*(\varphi) = \sum_{i=1}^{N} l^*(\varphi_i).$$

Thus we need to find $l^*(t)$

$$\begin{aligned} l^*(t) &= \sup_s \left[st - \max(0, 1-s)\right] \\ &= \max\left[\sup_{s<1}\left(st - (1-s)\right), \sup_{s\geq 1}(st)\right] \\ &= \max\left[\sup_{s<1}\left(s(t+1) - 1\right), \sup_{s\geq 1}(st)\right]. \end{aligned}$$

Now we analyse several cases in order to find explicit form for l^*. If $t > 0$, then st is not upper bounded above for $s \geq 1$, thus $l^*(t) = +\infty$. If $s < -1$, then $s(t+1)$ is not upper bounded above for $s < 1$, thus $l^*(t) = +\infty$. Let $t \in [-1, 0]$, then

$$\sup_{s\geq 1} st = t, \quad \sup_{s<1} s(t+1) - 1 = t.$$

Combining everything, we derive the explicit form of $l^*(t)$

$$l^*(t) = t + \iota_{[-1,0]}(t).$$

Finally, we have the following form for g^*

$$g^*(\varphi) = \sum_{i=1}^{N} \varphi_i + \iota_{[-1,0]}(\varphi_i) = \langle \varphi, \mathbf{1}\rangle + \iota_{[-1,0]^N}(\varphi), \tag{10}$$

where $\mathbf{1} \in \mathbb{R}^N$ is the vector of ones. □

Knowing the explicit form of g^* we are ready to compute a proximal operator for σg^*.

Lemma 4.8. *The proximal operator for σg^*, where g^* is defined in* (10)*, is given by*

$$\left[\mathrm{prox}_{\sigma g^*}(\varphi)\right]_i = \begin{cases} \varphi_i - \sigma & \text{if } \varphi_i \in [-1+\sigma, \sigma] \\ 0 & \text{if } \varphi_i > \sigma \\ -1 & \text{if } \varphi_i < -1 + \sigma \end{cases}.$$

Proof. We write the definition of a proximal operator

$$\mathrm{prox}_{\sigma g^*}(\varphi) = \arg\min_{\varphi'} \left[\sigma \left\langle \varphi', \mathbf{1}\right\rangle + \iota_{[-1,0]^N}(\varphi') + \frac{1}{2}\left\|\varphi' - \varphi\right\|^2\right].$$

The optimization subproblem inside $\arg\min$ is separable, hence we can solve it componentwise. We consider the problem of the form

$$\arg\min_{\varphi_i'} \left[\sigma\varphi_i' + \iota_{[-1,0]}(\varphi_i') + \frac{1}{2}(\varphi_i' - \varphi_i)^2\right].$$

This problem is equivalent to the following one

$$\min_{\varphi_i'} \left[\sigma\varphi_i' + \frac{1}{2}(\varphi_i' - \varphi_i)^2\right] \quad \text{s.t } \varphi_i' \leq 0, -1 - \varphi_i' \leq 0.$$

Note, that the function we minimize is a second degree polynomial with positive senior coefficient. Thus we can easily find a solution of this problem. Note, that if we solve unconstrained problem, then the solution is given by

$$\left(\varphi_i'\right)_* = \frac{-(\sigma - \varphi_i)}{2 \cdot 1/2} = \varphi_i - \sigma.$$

If $\varphi_i - \sigma \in [-1, 0]$, then global solution is also the solution of constrained problem. If $\varphi_i - \sigma > 0$, then $(\varphi_i')_* = 0$. If $\varphi_i - \sigma < -1$, then $(\varphi_i')_* = -1$. Combining all the above we obtain

$$\left(\varphi_i'\right)_* = \begin{cases} \varphi_i - \sigma & \text{if } \varphi_i \in [-1+\sigma, \sigma] \\ 0 & \text{if } \varphi_i > \sigma \\ -1 & \text{if } \varphi_i < -1 + \sigma \end{cases}.$$

□

The proximal operator of σg^* can be shortly written as the thresholding function.

Remark 4.9. *For clarification, when we work with SVM problem the role of arbitrary matrix A in (2) plays the matrix $D(b)A$, since SVM problem is written as*

$$\min_w \left[g(D(b)Aw) + \lambda \|w\|_1 \right].$$

Remark 4.10. *For PURE-CD method proximal operators are defined with respect to function and positive definite matrix. In general, it is not possible to give explicit form of the proximal operator (for example, if the function is ℓ_1 norm). However, for the method itself we only need to know the explicit form of proximal operator with diagonal matrix. This could be seen as the generalization of all proximal operators we have computed above, but with its own stepsize τ for each coordinate. Since all proximal operators are defined element-wise, this generalization is straightforward.*

5 Experiments

In this section we present experimental results of comparison of the methods described in Section 3.

5.1 Parameters setting and data sets

Rustem: to do

6 How to choose stepsizes if strong convexity parameters are unknown?

We work with the problem of the form

$$L(x,y) = f(x) + \langle Ax, y\rangle - g^*(y),$$

i.e. we assume that f_2 and g_2 are zero functions.

6.1 Adaptive local estimation

Based on the paper [Vladarean et al., 2021] we propose the following way how to handle the issue of unknown strong convexity parameters. From the definition of strong convexity

$$\varphi(x_1) \geq \varphi(x_2) + \langle \nabla\varphi(x_2), x_1 - x_2\rangle + \frac{\mu}{2}\|x_1 - x_2\|^2$$

we may estimate local strong convexity parameter that we need to compute x_{k+1} and y_{k+1} as follows

$$\mu_f^k = \frac{f(x_k) - f(x_{k-1}) - \langle \nabla f(x_{k-1}), x_k - x_{k-1}\rangle}{\frac{1}{2}\|x_k - x_{k-1}\|^2},$$
$$\mu_{g^*}^k = \frac{g^*(y_k) - g^*(y_{k-1}) - \langle \nabla g^*(y_{k-1}), y_k - y_{k-1}\rangle}{\frac{1}{2}\|y_k - y_{k-1}\|^2}.$$

Using these estimators we define constants τ_k and σ_k in the following way

$$\tau_k = \sqrt{\frac{\mu_{g^*}^k}{\mu_f^k}}\frac{1}{\|A\|}, \quad \sigma_k = \sqrt{\frac{\mu_f^k}{\mu_{g^*}^k}}\frac{1}{\|A\|},$$

or

$$\tau_k = \min\left\{\sqrt{\frac{\mu_{g^*}^k}{\mu_f^k}}\frac{1}{\|A\|}, \tau_{k-1}\sqrt{1 + \frac{\mu_{g^*}^{k-1}}{\mu_{g^*}^{k-2}}\frac{\mu_f^{k-2}}{\mu_f^{k-1}}}\right\}, \quad \sigma_k = \min\left\{\sqrt{\frac{\mu_f^k}{\mu_{g^*}^k}}\frac{1}{\|A\|}, \sigma_{k-1}\sqrt{1 + \frac{\mu_{g^*}^{k-2}}{\mu_{g^*}^{k-1}}\frac{\mu_f^{k-1}}{\mu_f^{k-2}}}\right\}.$$

6.2 Adaptive restart of PDHG

Definition 6.1. *We say that a function $f : \mathcal{X} \to \mathbb{R}\cup\{+\infty\}$ has a quadratic error bound if there exists η and an open region $\mathcal{R}$ that contains* $\arg\min f$ *such that for all $x \in \mathcal{R}$,*

$$f(x) \geq \min f + \frac{\eta}{2}\mathrm{dist}(x, \arg\min f)^2. \tag{11}$$

We shall use the acronym f has η-QEB.

Although this is more general than strong convexity, the quadratic error bound is not enough for saddle point problems. For example, for a large class of problems with linear constraints (i.e. $y \to L(x,y)$ is linear) QEB is not satisfied in y. To resolve this issue, we may resort to metric regularity.

Definition 6.2. *A set-valued function $F : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is metrically subregular at z for b if there exists $\eta > 0$ and a neighbourhood $N(z)$ of z such that $\forall\ z' \in N(z)$,*

$$dist(F(z', b)) \geq \eta\mathrm{dist}(z', F^{-1}(b)). \tag{12}$$

We denote $C(z) = [\partial f(z), \partial g^*(y)]$ and $M(z) = [A^\top y, -Ax]$. The Largrangian's subgradient is then $\tilde{\partial}L(z) = (C + M)(z)$. We put a tulde to emphasize the fact that the dual component if the negative of the subgradient.

Proposition 6.3. *If L is μ-strongly convex-concave, then $\tilde{\partial}L$ us μ-metrically sub-regular at z^* for 0, where z^* is the unique saddle point of L.*

PDHG can be conveniently seen as a fixed point algorithm $z_{k+1} = T(z_k)$ where T is defined as follows

$$\begin{aligned} \bar{x} &= \operatorname{prox}_{\tau f}(x - \tau A^\top y), & \bar{y} &= \operatorname{prox}_{\sigma g^*}(y + \sigma A\bar{x}) \\ x^+ &= \bar{x} - \tau A^\top(\bar{y} - y), & y^+ &= \bar{y} \\ T(x,y) &= (x^+, y^+). \end{aligned} \tag{13}$$

In strongly convex-concave case we are able to prove linear convergence of PDHG in the norm $\|\cdot\|_V$.

Proposition 6.4. *If L is μ-strongly convex-concave in the norm $\|\cdot\|_V$, then the iterates of PDHG satisfy for all k,*

$$\|z_{k+1} - z^*\|_V^2 \leq \left(1 + \frac{\mu}{1 + \mu/\Gamma}\right)^{-1} \|z_k - z^*\|_V^2, \tag{14}$$

where z^* is the unique saddle point of L and $\Gamma = (1 - \alpha_f)(1 - \sqrt{\gamma})$.

For $z = (x, y) \in \mathcal{Z}$, we denote $\|z\|_V^2 = (\tau^{-1}\|x\|^2 + \sigma^{-1}\|y\|^2)^{1/2}$. Let stepsizes satisfy $\gamma = \sigma\tau\|A\|^2 < 1$, $\tau L_f/2 \leq \alpha_f < 1$, $\alpha_g = \sigma L_{g^*}/2 \leq 1$, and $\sigma L_{g^*}/2 \leq \alpha_f(1 - \sigma\tau\|A\|^2)$. Using Proposition 6.3 we get another Proposition.

Proposition 6.5. *If $\tilde{\partial}L$ is metrically sub-regular at z^* for 0 and for all $z^* \in \mathcal{Z}^*$ with constant $\eta > 0$, then $(I - T)$ is metrically sub-regular at z^* for 0 and for all $z^* \in \mathcal{Z}^*$ with constant $\frac{\eta}{\sqrt{3}\eta + 2 + 2\sqrt{3}\max\{\alpha_g, \alpha_f\}}$, and PDHG converges linearly with rate $\left(1 - \frac{\eta^2(1-\alpha_f)(1-\sqrt{\gamma})}{\left(\sqrt{3}\eta + 2 + 2\sqrt{3}\max\{\alpha_f, \alpha_g\}\right)^2}\right)$.*

Let us assume that f and g^* are strongly convex function, but we do not know the strong convexity parameter of f. In this case L is strongly convex-concave. By Propositions 6.5 and 6.3 we get that $\tilde{\partial}L$ is μ-metrically sub-regular at z^* for 0, and $(I - T)$ is η-metrically sub-regular, where

$$\eta = \frac{\mu}{\sqrt{3}\mu + 2 + 2\sqrt{3}\max\{\alpha_g, \alpha_f\}}.$$

This implies the following

$$\|T(z) - z\|^2 \geq \eta^2 \|z - z^*\|^2. \tag{15}$$

Moreover, from Lemma 2 of [Fercoq, 2021] we get for $z' = z^*$ (note that z^* is a fixed point of T)

$$\begin{aligned} \lambda \|z - T(z) - z^* + T(z^*)\|^2 &\leq \|z - z^*\|_V^2 - \|T(z) - T(z^*)\|_V^2 - 2\mu_f \|\bar{x} - \bar{x}^*\|^2 - 2\mu_{g^*} \|\bar{y} - \bar{y}^*\|^2 \\ \lambda \|z - T(z)\|^2 &\leq \|z - z^*\|_V^2, \end{aligned} \tag{16}$$

where

$$\lambda = 1 - \alpha_f - \frac{\alpha_g - (1-\gamma)\alpha_f}{2} - \sqrt{(1-\alpha_f)^2\gamma + \left((1-\gamma)\alpha_f - \alpha_g\right)^2/4}.$$

Using the above in Proposition 6.4 we get

$$\begin{aligned} \lambda \|z_{k+1} - T(z_{k+1})\|^2 \leq \|z_{k+1} - z^*\|_V^2 &\leq \left(1 + \frac{\mu}{1 + \mu/\Gamma}\right)^{-1} \|z_k - z^*\|_V^2 \\ &\leq \eta^2 \left(1 + \frac{\mu}{1 + \mu/\Gamma}\right)^{-1} \|T(z_k) - z_k\|^2. \end{aligned} \tag{17}$$

We may use (17) as criterion for restarted PDHG.

References

Ahmet Alacaoglu, Olivier Fercoq, and Volkan Cevher. Random extrapolation for primal-dual coordinate descent. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

Amir Beck and Marc Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *IMA Journal of Numerical Analysis 39(4), 2069–2095*, 2009.

Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision 40(1), 120–145*, 2011.

Antonin Chambolle and Thomas Pock. An introduction to continuous optimization for imaging. *Acta Numerica, Cambridge University Press (CUP)*, 2016.

Olivier Fercoq. Quadratic error bound of the smoothed gap and the restarted averaged primal-dual hybrid gradient. 2021. URL `https://hal.archives-ouvertes.fr/hal-03228252`.

Olivier Fercoq and Zheng Qu. Adaptive restart of accelerated gradient methods under local quadratic growth condition. *SIAM Journal on Imaging Sciences, 2(1), 183–202*, 2019.

Jingwei Liang, Jalal Fadili, and Gabriel Peyré. Convergence rates with inexact non-expansive operators. *Mathematical Programming 159(1-2), 403–434*, 2016.

R. Tyrrell Rockafellar and Roger J.-B. Wets. *Variational Analysis*. Springer Verlag, Heidelberg, Berlin, New York, 1998.

Quoc Tran-Dinh, Olivier Fercoq, and Volkan Cevher. A smooth primal-dual optimization framework for nonsmooth composite convex minimization. *SIAM Journal on Optimization 28(1), 96–134*, 2018.

Maria-Luiza Vladarean, Yura Malitsky, and Volkan Cevher. A first-order primal-dual method with adaptivity to local smoothness. In *Proceedings of the 34th Advances in Neural Information Processing Systems 34 pre-proceedings (NeurIPS 2021)*, 2021.