Randomized Block Cubic Newton Method

Nikita Doikov [1] **Peter Richtárik** [2 3 4]

Abstract

We study the problem of minimizing the sum of three convex functions: a differentiable, twice-differentiable and a non-smooth term in a high dimensional setting. To this effect we propose and analyze a randomized block cubic Newton (RBCN) method, which in each iteration builds a model of the objective function formed as the sum of the *natural* models of its three components: a linear model with a quadratic regularizer for the differentiable term, a quadratic model with a cubic regularizer for the twice differentiable term, and perfect (proximal) model for the nonsmooth term. Our method in each iteration minimizes the model over a random subset of blocks of the search variable. RBCN is the first algorithm with these properties, generalizing several existing methods, matching the best known bounds in all special cases. We establish $\mathcal{O}(1/\epsilon)$, $\mathcal{O}(1/\sqrt{\epsilon})$ and $\mathcal{O}(\log(1/\epsilon))$ rates under different assumptions on the component functions. Lastly, we show numerically that our method outperforms the state-of-the-art on a variety of machine learning problems, including cubically regularized least-squares, logistic regression with constraints, and Poisson regression.

1. Introduction

In this paper we develop an efficient randomized algorithm for solving an optimization problem of the form

$$\min_{x \in Q} F(x) \stackrel{\text{def}}{=} g(x) + \phi(x) + \psi(x), \tag{1}$$

where $Q \subseteq \mathbb{R}^N$ is a closed convex set, and g, ϕ and ψ are convex functions with different smoothness and structural properties. Our aim is to capitalize on these different properties in the design of our algorithm. We assume that g has Lipschitz gradient[1], ϕ has Lipschitz Hessian, while ψ is allowed to be nonsmooth, albeit "simple".

1.1. Block structure

Moreover, we assume that the N coordinates of x are partitioned into n blocks of sizes $N_1, \dots, N_n$, with $\sum_i N_i = N$, and then write $x = (x_{(1)}, \dots, x_{(n)})$, where $x_{(i)} \in \mathbb{R}^{N_i}$. This block structure is typically dictated by the particular application considered. Once the block structure is fixed, we further assume that ϕ and ψ are *block separable*. That is, $\phi(x) = \sum_{i=1}^n \phi_i(x_{(i)})$ and $\psi(x) = \sum_{i=1}^n \psi_i(x_{(i)})$, where ϕ_i are twice differentiable with Lipschitz Hessians, and ψ_i are closed convex (and possibly nonsmooth) functions.

Revealing this block structure, problem (1) takes the form

$$\min_{x \in Q} F(x) \stackrel{\text{def}}{=} g(x) + \sum_{i=1}^n \phi_i(x_{(i)}) + \sum_{i=1}^n \psi_i(x_{(i)}). \tag{2}$$

We are specifically interested in the case when n is *big*, in which case it make sense to update a small number of the block in each iteration only.

1.2. Related work

There has been a substantial and growing volume of research related to second-order and block-coordinate optimization. In this part we briefly mention some of the papers most relevant to the present work.

A major leap in second-order optimization theory was made since the cubic Newton method was proposed by Griewank (1981) and independently rediscovered by Nesterov & Polyak (2006), who also provided global complexity guarantees.

Cubic regularization was equipped with acceleration by Nesterov (2008), adaptive stepsizes by (Cartis et al., 2011a;b) and extended to a universal framework by Grapiglia & Nesterov (2017). The universal schemes can automatically adjust to the implicit smoothness level of the objective. Cubically regularized second-order schemes for solving systems of nonlinear equations were developed by Nesterov

[1]National Research University Higher School of Economics, Moscow, Russia [2]King Abdullah University of Science and Technology, Thuwal, Saudi Arabia [3]University of Edinburgh, Edinburgh, United Kingdom [4]Moscow Institute of Physics and Technology, Dolgoprudny, Russia. Correspondence to: Nikita Doikov <nikitad101@gmail.com>, Peter Richtárik <peter.richtarik@kaust.edu.sa, peter.richtarik@ed.ac.uk>.

February 10, 2018

[1]Our assumption is bit more general than this; see Assumptions 1, 2 for details.

(2007) and randomized variants for stochastic optimization were considered by Tripuraneni et al. (2017); Ghadimi et al. (2017); Kohler & Lucchi (2017).

Despite their attractive global iteration complexity guarantees, the weakness of second-order methods in general, of cubic Newton in particular, is their high computational cost per iteration. This issue remains the subject of active research. For successful theoretical results related to the approximation of the cubic step we refer to (Agarwal et al., 2016) and (Carmon & Duchi, 2016).

At the same time, there are many successul attempts to use *block coordinate* randomization to accelerate first-order (Tseng & Yun, 2009; Richtárik & Takáč, 2014; 2016) and second-order (Qu et al., 2016) methods. By this work we are addressing the issue of combining the latter technique with cubic regularization, to get *a second-order method with proven global complexity guarantees and with a low cost per iteration*.

A powerful advance in convex optimization theory was the advent of *composite* or *proximal* first-order methods (see (Nesterov, 2013) as a modern reference). This technique has become available as an algorithmic tool in block coordinate setting as well (Richtárik & Takáč, 2014; Qu et al., 2016). Our aim in this work is the development of a *composite cubically regularized second-order method*.

1.3. Contributions

We propose a new randomized second-order proximal algorithm for solving convex optimization problems of the form (2). Our method, *Randomized Block Cubic Newton (RBCN)* (see Algorithm 1) treats the three functions appearing in (1) differently, according to their nature.

Our method is a *randomized block method* because in each iteration we update a random subset of the n blocks only. This facilitates faster convergence, and is suited to problems where n is very large. Our method is *proximal* because we keep the functions ψ_i in our model, which is minimized in each iteration, without any approximation. Our method is a *cubic Newton* method because we approximate each ϕ_i using a cubically-regularized second order model.

We are not aware of *any method* that can solve (2) via using the most appropriate models of the three functions (quadratic with a constant Hessian for g, cubically regularized quadratic for ϕ and no model for ψ), not even in the case $n = 1$.

Our approach generalizes several existing results:

- In the case when $n = 1$, $g = 0$ and $\psi = 0$, RBCN reduces to the cubically-regularized Newton method of Nesterov & Polyak (2006). Even when $n = 1$, RBCN can be seen as an extension of this method to *composite* optimization. For $n > 1$, RBCN provides an extension of the algorithm in Nesterov & Polyak (2006) to the *randomized block coordinate* setting, popular for high-dimensional problems.
- In the special case when $\phi = 0$ and $N_i = 1$ for all i, RBCN specializes to the stochastic Newton (SN) method of Qu et al. (2016). Applied to the empirical risk minimization problem (see Section 7), our method has a dual interpretation (see Algorithm 2). In this case, our method reduces to the stochastic dual Newton method (SDNA) also described in (Qu et al., 2016). Hence, RBCN can be seen as an extension of SN and SDNA to blocks of arbitrary sizes, and to the inclusion of the twice differentiable term ϕ.
- In the case when $\phi = 0$ and the simplest over approximation of g is assumed: $0 \preceq \nabla^2 g(x) \preceq LI$, the composite block coordinate gradient method Tseng & Yun (2009) can be applied to solve (1). Then our method extend it in two directions: add twice-differentiable terms ϕ and use more tightest model for g with all the global curvature information (if available).

We prove high probability global convergence guarantees under several regimes, summarized next:

- Under no additional assumptions on g, ϕ and ψ beyond convexity (and either boundedness of Q, or boundedness of the level sets of F on Q), we prove the rate
$$\mathcal{O}\left(\frac{n}{\tau\epsilon}\right),$$
where τ is the mini-batch size (see Theorem 1).
- Under certain conditions combining the properties of g with the way the random blocks are sampled, formalized by the assumption $\beta > 0$ (see (12) for the definition of β), we obtain the rate
$$\mathcal{O}\left(\frac{n}{\tau \max\{1,\beta\}\sqrt{\epsilon}}\right)$$
(see Theorem 2). In the special case when $n = 1$, we necessarily have $\tau = 1$ and $\beta = \mu/L$ (reciprocal of the condition number of g) we get the rate $\mathcal{O}(\frac{L}{\mu\sqrt{\epsilon}})$. If g is quadratic and $\tau = n$, then $\beta = 1$ and resulting complexity $\mathcal{O}(\frac{1}{\sqrt{\epsilon}})$ recovers the rate of cubic Newton established by Nesterov & Polyak (2006).
- Finally, if g is strongly convex, the above result can be improved (see Theorem 3) to
$$\mathcal{O}\left(\frac{n}{\tau \max\{1,\beta\}} \log\frac{1}{\epsilon}\right).$$

1.4. Contents

The rest of the paper is organized as follows. In Section 2 we introduce the notation and elementary identities needed to efficiently handle the block structure of our model. In Section 3 we make the various smoothness and convexity assumptions on g and ϕ_i formal. Section 4 is devoted to the description of the block sampling process used in our method, along with some useful identities. In Section 5 we describe formally our randomized block cubic Newton (RBCN) method. Section 6 is devoted to the statement and description of our main convergence results, summarized in the introduction. Missing proofs are provided in the appendix. In Section 7 we show how to apply our method to the empirical risk minimization problem. Applying RBCN to its dual leads to Algorithm 2. Finally, our numerical experiments on synthetic and real datasets are described in Section 8.

2. Block structure

To model a block structure, we decompose the space $\mathbb{R}^N$ into n subspaces in the following standard way. Let $\mathbf{U} \in \mathbb{R}^{N\times N}$ be a column permutation of the $N \times N$ identity matrix $\mathbf{I}$ and let a decomposition $\mathbf{U} = [\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_n]$ be given, where $\mathbf{U}_i \in \mathbb{R}^{N\times N_i}$ are n submatrices, $N = \sum_{i=1}^n N_i$. Subsequently, any vector $x \in \mathbb{R}^N$ can be uniquely represented as $x = \sum_{i=1}^n \mathbf{U}_i x_{(i)}$, where $x_{(i)} \stackrel{\text{def}}{=} \mathbf{U}_i^T x \in \mathbb{R}^{N_i}$.

In what follows we will use the standard Euclidean inner product: $\langle x, y\rangle \stackrel{\text{def}}{=} \sum_i x_i y_i$, Euclidean norm of a vector: $\|x\| \stackrel{\text{def}}{=} \sqrt{\langle x, x\rangle}$ and induced spectral norm of a matrix: $\|\mathbf{A}\| \stackrel{\text{def}}{=} \max_{\|x\|=1} \|\mathbf{A}x\|$. Using block decomposition, for two vectors $x, y \in \mathbb{R}^N$ we have:

$$\langle x, y\rangle = \left\langle \sum_{i=1}^n \mathbf{U}_i x_{(i)}, \sum_{j=1}^n \mathbf{U}_j y_{(j)} \right\rangle = \sum_{i=1}^n \langle x_{(i)}, y_{(i)}\rangle.$$

For a given nonempty subset S of $[n] \stackrel{\text{def}}{=} \{1, \dots, n\}$ and for any vector $x \in \mathbb{R}^N$ we denote by $x_{[S]} \in \mathbb{R}^N$ the vector obtained from x by retaining only blocks $x_{(i)}$ for which $i \in S$ and zeroing all other:

$$x_{[S]} \stackrel{\text{def}}{=} \sum_{i\in S} \mathbf{U}_i x_{(i)} = \sum_{i\in S} \mathbf{U}_i \mathbf{U}_i^T x.$$

Furthermore, for any matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ we write $\mathbf{A}_{[S]} \in \mathbb{R}^{N\times N}$ for the matrix obtained from $\mathbf{A}$ by retaining only elements whose indices are both in some coordinate blocks from S, formally:

$$\mathbf{A}_{[S]} \stackrel{\text{def}}{=} \left(\sum_{i\in S} \mathbf{U}_i \mathbf{U}_i^T\right) \mathbf{A} \left(\sum_{i\in S} \mathbf{U}_i \mathbf{U}_i^T\right).$$

Note that these definitions imply that

$$\langle \mathbf{A}_{[S]} x, y\rangle = \langle \mathbf{A} x_{[S]}, y_{[S]}\rangle, \quad x, y \in \mathbb{R}^N.$$

Let us define also for a fixed block decomposition the *block-diagonal* operator, which, up to permutation of coordinates, retains main diagonal blocks and nullifies off-diagonal:

$$\operatorname{blockdiag}(\mathbf{A}) \stackrel{\text{def}}{=} \sum_{i=1}^n \mathbf{U}_i \mathbf{U}_i^T \mathbf{A} \mathbf{U}_i \mathbf{U}_i^T = \sum_{i=1}^n \mathbf{A}_{[\{i\}]}.$$

Finally, denote $\mathbb{R}^N_{[S]} \stackrel{\text{def}}{=} \left\{x_{[S]} \,|\, x \in \mathbb{R}^N\right\}$. This is a linear subspace of $\mathbb{R}^N$ composed of vectors which are zero in blocks $i \notin S$.

3. Assumptions

In this section we formulate our main assumptions about differentiable components of (2) and provide some basic examples.

3.1. Smoothness assumptions

We assume that $g : \mathbb{R}^N \to \mathbb{R}$ is a differentiable function and all $\phi_i : \mathbb{R}^{N_i} \to \mathbb{R}$, $i \in [n]$ are twice differentiable. Thus, at any point $x \in \mathbb{R}^N$ we should be able to compute all the gradients $\{\nabla g(x), \nabla\phi_1(x_{(1)}), \dots, \nabla\phi_n(x_{(n)})\}$ and the Hessians $\{\nabla^2\phi_1(x_{(1)}), \dots, \nabla^2\phi_n(x_{(n)})\}$, or at least its actions to arbitrary vector h of appropriate dimension.

Next, we formalize our assumptions about convexity and a level of smoothness. Speaking informally, we look at g as at something which is similar to quadratic and all the ϕ_i are arbitrary twice-differentiable and smooth.

Assumption 1 (Convexity) *There is a positive semidefinite matrix $\mathbf{G} \succeq 0$ such that for all $x, h \in \mathbb{R}^N$:*

$$g(x+h) \geq g(x) + \langle \nabla g(x), h\rangle + \frac{1}{2}\langle \mathbf{G}h, h\rangle, \qquad (3)$$

$$\phi_i(x_{(i)} + h_{(i)}) \geq \phi_i(x_{(i)}) + \langle \nabla\phi_i(x_{(i)}), h_{(i)}\rangle, \quad i \in [n].$$

Plugging $\mathbf{G} = 0$ into the (3) we get definition of just *convex* functions. For strictly positive definite $\mathbf{G} \succ 0$ the objective will be *strongly convex* with a constant $\mu \stackrel{\text{def}}{=} \lambda_{\min}(\mathbf{G}) > 0$.

Note, that for all ϕ_i we ask only about convexity. However, if on occasion we know that any of ϕ_i is strongly convex: $\lambda_{\min}(\nabla^2\phi_i(y)) \geq \mu_i > 0$ for all $y \in \mathbb{R}^{N_i}$, we can *move* this strong convexity to g by subtracting $\frac{\mu_i}{2}\|x_{(i)}\|^2$ from ϕ_i and adding it to g. This extra knowledge may in some particular cases improve convergence guarantees for our algorithm, but does not change actual computations at all.

Assumption 2 (Smoothness of g) *There is a positive semidefinite matrix $\mathbf{A} \succeq 0$ such that for all $x, h \in \mathbb{R}^N$:*

$$g(x+h) \leq g(x) + \langle \nabla g(x), h \rangle + \frac{1}{2}\langle \mathbf{A}h, h \rangle. \quad (4)$$

The main example of g, which we are keeping in mind during the paper is a quadratic function $g(x) = \frac{1}{2}\langle \mathbf{M}x, x \rangle$ with a symmetric positive semidefinite $\mathbf{M} \in \mathbb{R}^{N \times N}$ for which both (3) and (4) hold with $\mathbf{G} = \mathbf{A} = \mathbf{M}$.

Of course, any convex g with Lipschitz-continuous gradient with a constant $L \geq 0$ satisfies (3) and (4) with $\mathbf{G} = 0$ and $\mathbf{A} = L\mathbf{I}$ (Nesterov, 2004).

Assumption 3 (Smoothness of ϕ_i) *For every $i \in [n]$ there is a nonnegative constant $\mathcal{H}_i \geq 0$ such that the Hessian of ϕ_i is Lipschitz-continuous:*

$$\|\nabla^2 \phi_i(x+h) - \nabla^2 \phi_i(x)\| \leq \mathcal{H}_i \|h\|, \quad (5)$$

for all $x, h \in \mathbb{R}^{N_i}$.

Examples of functions which satisfy (5) with a known Lipschitz constant of Hessian $\mathcal{H}$ are *quadratic*: $\phi(t) = \|Ct - t_0\|^2$ ($\mathcal{H} = 0$ for all the parameters), *cubed norm*: $\phi(t) = (1/3)\|t - t_0\|^3$ ($\mathcal{H} = 2$, see Lemma 5 in (Nesterov, 2008)), *logistic regression loss*: $\phi(t) = \log(1 + \exp(t))$ ($\mathcal{H} = 1/(6\sqrt{3})$, see Proposition 1 in the appendix).

For a fixed set of indices $S \subset [n]$ denote

$$\phi_S(x) \stackrel{\text{def}}{=} \sum_{i \in S} \phi_i(x_{(i)}), \quad x \in \mathbb{R}^N.$$

It is easy to see that:

$$\langle \nabla \phi_S(x), h \rangle = \sum_{i \in S} \langle \nabla \phi_i(x_{(i)}), h_{(i)} \rangle, \quad x, h \in \mathbb{R}^N,$$
$$\langle \nabla^2 \phi_S(x) h, h \rangle = \sum_{i \in S} \langle \nabla^2 \phi_i(x_{(i)}) h_{(i)}, h_{(i)} \rangle, \quad x, h \in \mathbb{R}^N.$$

Lemma 1 *If Assumption 3 holds, then for all $x, h \in \mathbb{R}^N$ we have the following second-order approximation bound:*

$$\left| \phi_S(x+h) - \phi_S(x) - \langle \nabla \phi_S(x), h \rangle - \frac{1}{2}\langle \nabla^2 \phi_S(x) h, h \rangle \right| \leq \max_{i \in S} \{\mathcal{H}_i\} \cdot \|h_{[S]}\|^3. \quad (6)$$

For further usage denote $\mathcal{H}_F \stackrel{\text{def}}{=} \max_{i=1}^{n} \mathcal{H}_i$.

4. Sampling of blocks

Let us state some basic properties of probability sampling $\hat{S}$, which is essentially a distribution over subsets of indices $S \subset [n]$. For a fixed block-decomposition we associate the probability matrix $\mathbf{P} \in \mathbb{R}^{N \times N}$ to sampling $\hat{S}$: every element of $\mathbf{P}$ is the probability of choosing a pair of blocks, which contains indices of this element. Denoting by $\mathbf{E} \in \mathbb{R}^{N \times N}$ the matrix of all ones, we have $\mathbf{P} = \mathbb{E}\left[\mathbf{E}_{[\hat{S}]}\right]$. We will use in our analysis the following

Assumption 4 (Uniform sampling) *It holds*

$$\mathbb{P}(i \in \hat{S}) = \mathbb{P}(j \in \hat{S}) = p, \quad i, j \in [n].$$

This means that the diagonal of $\mathbf{P}$ is constant: $\mathbf{P}_{ii} = p$ for all $i \in [N]$. Denote $\tau \stackrel{\text{def}}{=} \mathbb{E}[|\hat{S}|] = np$. It is easy to see that (Corollary 3.1 in (Qu & Richtárik, 2016)):

$$\mathbb{E}\left[\mathbf{A}_{[\hat{S}]}\right] = \mathbf{A} \circ \mathbf{P}, \quad (7)$$

where $\circ$ denotes the Hadamard product.

If $\hat{S}$ is τ-nice (this is a special type of uniform samplings; one in which we pick subsets of size τ uniformly at random), then (see Lemma 4.3 in (Qu & Richtárik, 2016))

$$\mathbf{P} = \frac{\tau}{n}\left((1-\gamma)\,\mathrm{blockdiag}(\mathbf{E}) + \gamma \mathbf{E}\right), \quad (8)$$

where $\gamma = (\tau - 1)/(n-1)$.

In particular, the above results in the following:

Lemma 2 *For a τ-nice sampling $\hat{S}$ we have*

$$\mathbb{E}[\mathbf{A}_{[\hat{S}]}] = \frac{\tau}{n}\left(1 - \frac{\tau - 1}{n-1}\right)\mathrm{blockdiag}(\mathbf{A}) + \frac{\tau(\tau-1)}{n(n-1)}\mathbf{A}.$$

Proof: Combine (7) and (8). ■

5. Algorithm

Due to the problem structure (2) and by smoothness of the components – (4) and (5), for a fixed subset of indices $S \subset [n]$, it is natural to consider the following *model* $M_{H,S}(x, y)$ of our objective F around a point $x \in \mathbb{R}^N$, which is a combination of first-order model for g with a global curvature information provided by matrix $\mathbf{A}$, second-order model with cubic regularization (following (Nesterov & Polyak, 2006)) for the ϕ-component and non-differentiable terms ψ_i as it is:

$$M_{H,S}(x; y) \stackrel{\text{def}}{=} F(x) + \langle (\nabla g(x))_{[S]}, y \rangle + \frac{1}{2}\langle \mathbf{A}_{[S]} y, y \rangle + \\ + \langle (\nabla \phi(x))_{[S]}, y \rangle + \frac{1}{2}\langle (\nabla^2 \phi(x))_{[S]} y, y \rangle + \frac{H}{6}\|y_{[S]}\|^3 + \\ + \sum_{i \in S}\Big(\psi_i(x_{(i)} + y_{(i)}) - \psi_i(x_{(i)})\Big). \quad (9)$$

By (4) and (6) we get a global upper bound:

$$F(x+y) \leq M_{H,S}(x; y), \quad x \in \mathbb{R}^N,\ y \in \mathbb{R}^N_{[S]},$$

for large enough H, at least for $H \geq \max\limits_{i \in S} \mathcal{H}_i$. Moreover, the value of all summands from $M_{H,S}(x;y)$ depends only on subset of blocks $\{y_{(i)} | i \in S\}$ and therefore a minimizer

$$T_{H,S}(x) \stackrel{\text{def}}{=} \operatorname*{argmin}_{\substack{y \in \mathbb{R}^N_{[S]} \\ \text{s.t. } x+y \in Q}} M_{H,S}(x;y) \tag{10}$$

can be found effectively for small $|S|$ and as long as Q is *simple* (for example, affine). Denote a minimum of the cubic model by $M^*_{H,S}(x) \stackrel{\text{def}}{=} M_{H,S}(x; T_{H,S}(x))$. This is formalized as Algorithm 1.

Algorithm 1 RBCN: Randomized Block Cubic Newton

Parameters: sampling distribution $\hat{S}$
Initialization: choose initial point $x^0 \in Q$
for $k = 0, 1, 2, \ldots$ **do**
  Sample $S_k \sim \hat{S}$
  Find $H_k \in (0,\, 2\mathcal{H}_F]$ such that
    $F(x^k + T_{H_k,S_k}(x^k)) \leq M^*_{H_k,S_k}(x^k)$
  Make a step $x^{k+1} := x^k + T_{H_k,S_k}(x^k)$
end for

6. Convergence results

In this subsection we analyze convergence rate of the Algorithm 1 for a general class of convex problems and for more specific strongly convex case. We will be focusing on the family of *uniform samplings* only, but generalizations to other sampling distributions are also possible.

6.1. Convex loss

We start from a general situation, when the term $g(x)$ and all the $\phi_i(x_{(i)})$ and $\psi_i(x_{(i)}), i \in [n]$ are convex but not necessary strongly convex.

Denote by D a maximum distance from an optimum point x^* to the initial level set:

$$D \stackrel{\text{def}}{=} \sup\Big\{ \|x - x^*\| \;\Big|\; x \in Q,\, F(x) \leq F(x^0) \Big\}.$$

Theorem 1 *Let Assumptions 1, 2, 3, 4 hold. Let solution $x^* \in Q$ of the problem (1) exists and level sets are bounded: $D < +\infty$. Choose required accuracy $\varepsilon > 0$ and confidence level $\rho \in (0,1)$. Then after proceeding*

$$K \geq \frac{2}{\varepsilon} \frac{n}{\tau} \left(1 + \log \frac{1}{\rho}\right) \max\Big\{ LD^2 + \mathcal{H}_F D^3,\, F(x^0) - F^* \Big\} \tag{11}$$

iterations of the Algorithm 1, where $L \stackrel{\text{def}}{=} \lambda_{\max}(\mathbf{A})$, we have

$$\mathbb{P}\Big(F(x^K) - F^* \leq \varepsilon \Big) \;\geq\; 1 - \rho.$$

Given theoretical result provides global sublinear rate of convergence, with iteration complexity of the order $O\big(1/\varepsilon\big)$.

Note that for a case $\phi(x) \equiv 0$ we can put $\mathcal{H}_F = 0$, and Theorem 1 in this situation restates well-known result about convergence of composite gradient-type block-coordinate methods (see, for example, (Richtárik & Takáč, 2014)).

6.2. Strongly convex loss

Here we study the case when the matrix $\mathbf{G}$ from the convexity assumption (3) is strictly positive definite: $\mathbf{G} \succ 0$, which means that the objective F is *strongly convex* with a constant $\mu \stackrel{\text{def}}{=} \lambda_{\min}(\mathbf{G}) > 0$.

Denote by β a *condition number* for the function g and sampling distribution $\hat{S}$: the maximum nonnegative real number such that

$$\beta \cdot \mathbb{E}_{S \sim \hat{S}}\big[\mathbf{A}_{[S]}\big] \;\preceq\; \frac{\tau}{n} \mathbf{G}. \tag{12}$$

If (12) holds for *all* nonnegative β we put by definition $\beta \equiv +\infty$.

A simple lower bound exists: $\beta \geq \frac{\mu}{L} > 0$, where $L = \lambda_{\max}(\mathbf{A})$, as in Theorem 1. However, because (12) depends not only on g, but also on sampling distribution $\hat{S}$, it is possible that $\beta > 0$ even if $\mu = 0$ (for example, $\beta = 1$ if $\mathbb{P}(S = [n]) = 1$ and $\mathbf{A} = \mathbf{G} \neq 0$).

The following theorems describe global iteration complexity guarantees of the order $O\big(1/\sqrt{\varepsilon}\big)$ and $O\big(\log(1/\varepsilon)\big)$ for the Algorithm 1 in the cases $\beta > 0$ and $\mu > 0$ correspondingly, which is an improvement of general $O\big(1/\varepsilon\big)$.

Theorem 2 *Let Assumptions 1, 2, 3, 4 hold. Let solution $x^* \in Q$ of the problem (1) exists, level sets are bounded: $D < +\infty$ and assume that β, which is defined by (12) is greater than zero. Choose required accuracy $\varepsilon > 0$ and confidence level $\rho \in (0,1)$. Then after proceeding*

$$K \geq \frac{2}{\sqrt{\varepsilon}} \frac{n}{\tau} \frac{1}{\sigma} \left(2 + \log \frac{1}{\rho}\right) \sqrt{\max\Big\{ \mathcal{H}_F D^3,\, F(x^0) - F^* \Big\}} \tag{13}$$

iterations of the Algorithm 1, where $\sigma \stackrel{\text{def}}{=} \min\{\beta, 1\} > 0$, we have

$$\mathbb{P}\Big(F(x^K) - F^* \leq \varepsilon \Big) \;\geq\; 1 - \rho.$$

Theorem 3 *Let Assumptions 1, 2, 3, 4 hold. Let solution $x^* \in Q$ of the problem (1) exists and $\mu \stackrel{\text{def}}{=} \lambda_{\min}(\mathbf{G})$ is strictly positive. Then after proceeding*

$$K \geq \frac{3}{2} \log\left(\frac{F(x^0) - F^*}{\varepsilon \rho} \right) \frac{n}{\tau} \frac{1}{\sigma} \sqrt{\max\left\{ \frac{\mathcal{H}_F D}{\mu}, 1 \right\}} \tag{14}$$

iterations of the Algorithm 1, we have

$$\mathbb{P}\Big(F(x^K) - F^* \leq \varepsilon \Big) \;\geq\; 1 - \rho.$$

Provided complexity estimates make us able to analyze which parameters of the problem directly affect to convergence rate of the algorithm.

The first bound (13) gives $\sqrt{D_0/\varepsilon}$ improvement over (11) but multiplies iterations by additional factor $\sigma^{-1} = (\min\{\beta, 1\})^{-1}$, which is growing up when the condition number β (which depends on g and sampling distribution $\hat{S}$) is becoming smaller.

In opposite and limit case, when the *quadratic* part of the objective is vanished ($g(x) \equiv 0 \Rightarrow \sigma = 1$) and $\psi = 0$, the Algorithm 1 is turned to be a parallelized block-independent minimization of ϕ_i components of the objective via cubic-regularized Newton steps (Nesterov & Polyak, 2006), and estimate (13), thus, generalizes a known result about its convergence in a nonrandomized ($\tau = n,\ \rho \to 1$) setting.

The second bound (14) guarantees a linear rate of convergence, which means logarithmic dependence on required accuracy ε for the number of iterations, and the main complexity factor becomes a product of two terms: $\sigma^{-1} \cdot \max\{\mathcal{H}_F D/\mu, 1\}^{1/2}$. In a case $\phi(x) = 0$ we can put $\mathcal{H}_F = 0$ and, by that, to get a restate of the randomized Newton-type method and its linear rate of convergence from (Qu et al., 2016)

Despite of the fact that linear rate is asymptotically better than sublinear, and $O\big(1/\sqrt{\varepsilon}\big)$ is asymptotically better than $O\big(1/\varepsilon\big)$, comparing algorithms, we need to take into account other factors, which slow down convergence rate. Thus, while $\mu = \lambda_{\min}(\mathbf{G}) \to 0$, estimate (13) is becoming better than (14) as well as (11) is becoming better than (13) while $\beta \to 0$.

6.3. Implementation issues

Let us explain how one step of the method can be performed, which requires minimizing of the cubic model (10), possibly with some simple convex constrains.

The first and the classical approach (which was proposed in (Nesterov & Polyak, 2006) and before for trust-region methods in (Conn et al., 2000)) works with unconstrained ($Q \equiv \mathbb{R}^N$) and differentiable case ($\psi(x) \equiv 0$). It needs firstly to find a root of a special one-dimensional nonlinear equation (this can be done, for example, by simple Newton iterations). After that, to produce a step we just solve one linear system. Then, total complexity of solving the subproblem can be estimated as $O(d^3)$ arithmetical operations, where d is the dimension of subproblem, $d = |S|$ in our case. Since some matrix factorization is used, the cost of the Cubic regularized Newton step is actually similar by efficiency to the classical Newton one. See also (Gould et al., 2010) for detailed analysis. For the case of affine constrains, the same procedure can be applied, example of which is presented by Lemma 3 in the next section.

Another approach is based on finding inexact solution of the subproblem by fast approximate eigenvalue computations (Agarwal et al., 2016) or by applying gradient descent (Carmon & Duchi, 2016). Both of these schemes provide global convergence guarantees and are Hessian-free, thus they need only a procedure of multiplying quadratic part of (9) to arbitrary vector, without storing the full matrix. The latter approach is the most universal one and can be spread to composite case, by using proximal gradient method and its accelerated variant (Nesterov, 2013).

To find parameter H_k on every iteration there are basically two strategies: a *constant choice* $H_k := \max_{i \in S_k}\{\mathcal{H}_i\}$ or $H_k := \mathcal{H}_F$, if Lipschitz constants of the Hessians are known, or simple *adaptive procedure*, which performs a truncated binary search and has a logarithmic cost per one step of the method. Example of such procedure can be found in primal-dual Algorithm 2 from the next section.

6.4. Extension of problem class

Randomized cubic model (9), which has been considered and analyzed before, arises naturally from the separable structure (2) and by our smoothness assumptions (4), (5). Let us discuss an interpretation of the Algorithm 1 in terms of general problem $\min\limits_{x \in \mathbb{R}^N} F(x)$ with twice-differentiable F (omitting non-differentiable component for simplicity). One can state and minimize a model $m_{H,S}(x; y) \equiv F(x) + \langle (\nabla F(x))_{[S]}, y\rangle + \frac{1}{2}\langle (\nabla^2 F(x))_{[S]} y, y\rangle + \frac{H}{6}\|y_{[S]}\|^3$ which is just a *sketched* version of the originally proposed Cubic Newton method (Nesterov & Polyak, 2006). For alternative sketched variants of Newton-type methods but without cubic regularization see (Pilanci & Wainwright, 2015).

The latter model $m_{H,S}(x; y)$ coincides with used by us model $M_{H,S}(x; y)$ in a case when inequality (4) from the smoothness assumption for g turns to be exact equality, i.e. if the function g is quadratic with the Hessian matrix $\nabla^2 g(x) \equiv \mathbf{A}$. Thus, we may use $m_{H,S}(x; y)$ instead of $M_{H,S}(x; y)$, which remains cheap computational cost of each step for small $|S|$ but does not give any convergence guarantees for the general F, to the best of our knowledge, unless $S = [n]$. However, it can be a workable approach, when the separable structure (2) is not provided.

Note also, that Assumption 3 about Lipschitz-continuity of the Hessian is not too restrictive, recent result (Grapiglia & Nesterov, 2017) shows that Newton-type methods with cubic regularization and with a standard procedure of *adaptive* estimate of $\mathcal{H}_F$ on each step, automatically fit the actual level of smoothness of the Hessian without any additional changes in the algorithm.

Moreover, step (10) of the method as a global minimum of $M_{H,S}(x; y)$ is well-defined and can be computed even in nonconvex cases (Nesterov & Polyak, 2006) what allows

to apply the method to nonconvex objective as well, but without known theoretical guarantees for $S \neq [n]$.

7. Empirical risk minimization

One of the most popular examples of optimization problems in machine learning is *empirical risk minimization* problem, which in many cases can be formulated as follows:

$$\min_{w\in\mathbb{R}^d}\left[P(w) \equiv \frac{1}{m}\sum_{i=1}^{m}\phi_i(b_i^T w) + \lambda g(w)\right], \qquad (15)$$

where ϕ_i are convex *loss functions*, g is a *regularizer*, variables w are *weights* of a model and m is a size of a dataset.

7.1. Constrained problem reformulation

Let us consider the case, when dimension d of the problem (15) is very *huge* and $d \gg m$. This asks us to use some coordinate-randomization technique. Note, that formulations (15) does not directly fit our problem setup (2), but we can easily restate it as following constrained optimization problem, by introducing new variables $\alpha_i \equiv b_i^T w$:

$$\min_{\substack{w\in\mathbb{R}^d\\ \alpha\in\mathbb{R}^m}}\left[\frac{1}{m}\sum_{i=1}^{m}\phi_i(\alpha_i)+\lambda g(w)+\sum_{i=1}^{m}\mathbb{I}\{\alpha_i = b_i^T w\}\right]. \quad (16)$$

Following our framework, on every step we will sample a random subset of coordinates $S \subset [d]$ of weights w, build the cubic model of the objective (assuming that ϕ_i and g satisfy (4), (5)):

$$M_{H,S}(w,\alpha;y,h) \equiv \lambda\left(\langle(\nabla g(w))_{[S]}, y\rangle + \frac{1}{2}\langle \mathbf{A}_{[S]}y, y\rangle\right)+$$
$$+\frac{1}{m}\left(\sum_{i=1}^{m}\left(\phi_i'(\alpha_i)h_i + \frac{1}{2}\phi_i''(\alpha_i)h_i^2\right) + \frac{H}{6}\|h\|^3\right) + P(w)$$

and minimize it by y and h on the affine set:

$$(y^*, h^*) := \operatorname*{argmin}_{\substack{y\in\mathbb{R}^d_{[S]}, h\in\mathbb{R}^m\\ \text{s.t. } h=\mathbf{B}y}} M_{H,S}(w,\alpha;y,h), \qquad (17)$$

where rows of matrix $\mathbf{B} \in \mathbb{R}^{m\times d}$ are b_i^T. Then, updates of the variables are: $w^+ := w + y^*$ and $\alpha^+ := \alpha + h^*$.

The following lemma is addressing the issue of how to solve (17), which is required on every step of the method. Its proof can be found in the appendix.

Lemma 3 *Denote by $\hat{\mathbf{B}} \in \mathbb{R}^{m\times|S|}$ the submatrix of $\mathbf{B}$ with row indices from S, by $\hat{\mathbf{A}} \in \mathbb{R}^{|S|\times|S|}$ the submatrix of $\mathbf{A}$ with elements whose both indices are from S, by $b_1 \in \mathbb{R}^{|S|}$ the subvector of $\nabla g(w)$ with element indices from S. Denote vector $b_2 \equiv\big(\phi_i'(\alpha_i)\big)_{i=1}^m$ and $b \equiv m\lambda b_1 + \hat{\mathbf{B}}^T b_2$. Define the family of matrices $\mathbf{Z}(\tau) : \mathbb{R}_+ \to \mathbb{R}^{|S|\times|S|}$:*

$$\mathbf{Z}(\tau) \stackrel{\text{def}}{=} m\lambda\hat{\mathbf{A}} + \hat{\mathbf{B}}^T\Big(\operatorname{diag}(\phi_i''(\alpha_i)) + \frac{H\tau}{2}\mathbf{I}\Big)\hat{\mathbf{B}}.$$

Then the solution (y^*, h^*) of (17) can be found from equations: $\mathbf{Z}(\tau^*)y_S^* = -b,\ h^* = \hat{\mathbf{B}}y_S^*$, where $\tau^* \geq 0$ satisfies one-dimensional nonlinear equation: $\tau^* = \|\hat{\mathbf{B}}(\mathbf{Z}(\tau^*))^\dagger b\|$ and $y_S^* \in \mathbb{R}^{|S|}$ is the subvector of the solution y^* with element indices from S.

Thus, after finding the root of nonlinear *one-dimensional* equation, we need to solve $|S| \times |S|$ linear system to compute y^*, and do one matrix-vector multiplication, with the matrix of size $m \times |S|$ to find h^*. Matrix $\mathbf{B}$ usually has a sparse structure when m is big, which also should be used in effective implementation.

7.2. Maximizing the dual problem

Another approach to solving optimization problem (15) is to maximize its Fenchel dual (Rockafellar, 1997):

$$\max_{\alpha\in\mathbb{R}^m}\left[D(\alpha) \equiv \frac{1}{m}\sum_{i=1}^{m} -\phi_i^*(-\alpha_i) - \lambda g^*\Big(\frac{1}{\lambda m}\mathbf{B}^T\alpha\Big)\right], \qquad (18)$$

where g^* and $\{\phi_i^*\}$ are *the Fenchel conjugate* functions of g and $\{\phi_i\}$ respectively, $f^*(s) \stackrel{\text{def}}{=} \sup_x[\,\langle s, x\rangle - f(x)\,]$ for arbitrary f. It is know (Bertsekas, 1978), that if ϕ_i is twice-differentiable in a neighborhood of y and $\nabla^2\phi_i(y) \succ 0$ in this area, then its Fenchel conjugate ϕ_i^* is also twice-differentiable in some neighborhood of $s = \nabla\phi_i(y)$ and it holds: $\nabla^2\phi_i^*(s) = (\nabla^2\phi_i(y))^{-1}$.

Then, in a case of smooth differentiable g^* and twice-differentiable $\phi_i^*, i \in [m]$ we can apply our framework to (18), by doing cubic steps in random subsets of the dual variables $\alpha \in \mathbb{R}^m$. The primal $w \in \mathbb{R}^d$ corresponded to particular α can be computed from the stationary equation

$$w = \nabla g^*\left(\frac{1}{\lambda m}\mathbf{B}^T\alpha\right),$$

which holds for solutions of primal (15) and dual (18) problems in a case of strong duality. Let us assume that the function g is 1-strongly convex (which is of course true for ℓ_2-regularizer $1/2\|w\|_2^2$), then for $G(\alpha) \equiv \lambda g^*\Big(\frac{1}{\lambda m}\mathbf{B}^T\alpha\Big)$ the uniform bound for the Hessian exists: $\nabla^2 G(\alpha) \preceq \frac{1}{\lambda m^2}\mathbf{B}^T\mathbf{B}$ and we can build, as before, the following cubic

model and its minimizer (setting $Q \equiv \bigcap_{i=1}^m \operatorname{dom} \phi_i^*$):

$$M_{H,S}(\alpha, h) \equiv -D(\alpha) + \lambda \left\langle \nabla g^* \left(\frac{1}{\lambda m} \mathbf{B}^T \alpha \right), h_{[S]} \right\rangle + \\ + \frac{1}{2\lambda m^2} \|\mathbf{B} h_{[S]}\|^2 + \frac{1}{m} \sum_{i \in S} \Big[-\langle \nabla \phi_i^*(-\alpha_i), h_i \rangle + \\ + \frac{1}{2} \langle \nabla^2 \phi_i^*(-\alpha_i) h_i, h_i \rangle \Big] + \frac{H}{6} \|h_{[S]}\|^3; \; S \subset [m],$$

$$T_{H,S}(\alpha) \equiv \operatorname*{argmin}_{\substack{h \in \mathbb{R}^m_{[S]} \\ \text{s.t. } \alpha + h \in Q}} M_{H,S}(\alpha, h),$$

$$M^*_{H,S}(\alpha) \equiv M_{H,S}(\alpha, T_{H,S}(\alpha)).$$

Because in general we may not know exact Lipschitz-constant for the Hessians, we do an adaptive search for estimating H. Resulting primal-dual scheme is presented in the Algorithm 2.

When a small subset of coordinates S is used, the most expensive operations become: computation of the objective at current point $D(\alpha)$ and matrix-vector product $\mathbf{B}^T \alpha$. Both of them can be significantly optimized by storing already computed values in memory and updating only changed information on every step.

Algorithm 2 Stochastic Dual Cubic Newton Ascent (SDCNA)

Parameters: sampling distribution $\hat{S}$
Initialization: choose initial $\alpha^0 \in Q$ and $H_0 > 0$
for $k = 0, 1, 2, \ldots$ **do**
  Make a primal update $w^k := \nabla g^* \left(\frac{1}{\lambda m} \mathbf{B}^T \alpha^k \right)$
  Sample $S_k \sim \hat{S}$
  While $M^*_{H_k, S_k}(\alpha^k) > -D(\alpha^k + T_{H_k, S_k}(\alpha^k))$ do
    $H_k := 1/2 \cdot H_k$
  Make a dual update $\alpha^{k+1} := \alpha^k + T_{H_k, S_k}(x^k)$
  Set $H_{k+1} := 2 \cdot H_k$
end for

8. Numerical experiments

8.1. Synthetic

We consider the following synthetic regression task:

$$\min_{x \in \mathbb{R}^N} \frac{1}{2} \|\mathbf{A}x - b\|_2^2 + \sum_{i=1}^N \frac{c_i}{6} |x_i|^3$$

with randomly generated parameters and for different N. On each problem of this type we run Algorithm 1 and evaluate total computational time until reaching 10^{-12} accuracy in function residual. It turns out that using middle-size blocks of coordinates on each step is the best choice in terms of total computational time, comparing it with small coordinate subsets and with full-coordinate method.

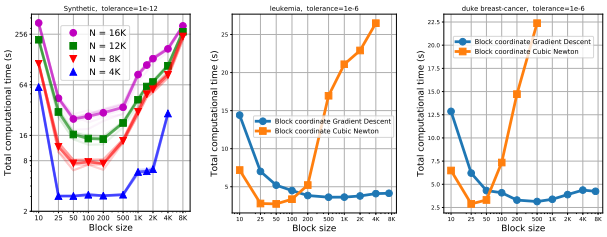

Figure 1. Time it takes to solve a problem for different sampling block sizes. Left: synthetic problem. Center and right: logistic regression for real data.

8.2. Logistic regression

In this experiment we train ℓ_2-regularized logistic regression model for classification task with two classes by its constrained reformulation (16) and compare the Algorithm 1 with the Block coordinate Gradient Descent (see, for example, (Tseng & Yun, 2009)) on the datasets[2]: *leukemia* ($m = 38, d = 7129$) and *duke breast-cancer* ($m = 44, d = 7129$). We see that using coordinate blocks of size $25 - 50$ for the Cubic Newton outperforms all other cases of both methods in terms of total computational time. Increasing block size further starts to significantly slow down the method because of high cost of every iteration.

8.3. Poisson regression

In this experiment we train Poisson model for regression task with integer responses by the primal-dual Algorithm 2 and compare it with SDCA (Shalev-Shwartz & Zhang, 2013) and SDNA (Qu et al., 2016) methods on synthetic ($m = 1000, d = 200$) and real data[3] ($m = 319, d = 20$). Our method reaches required accuracy in much smaller number of epochs but remains computational efficiency of one step as SDNA method has.

In the following set of experiments with Poisson regression we take real datasets[4]: *australian* ($m = 690, d = 14$), *breast-cancer* ($m = 683, d = 10$), *splice* ($m = 1000, d = 60$), *svmguide3* ($m = 1243, d = 21$) and use for response a random vector $y \in (\mathbb{N} \cup \{0\})^m$ from the standard Poisson distribution.

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

[3] https://www.kaggle.com/pablomonleon/montreal-bike-lanes

[4] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

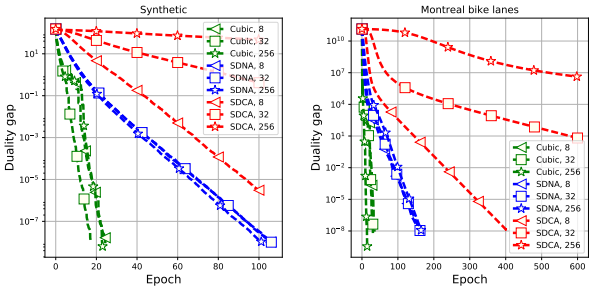

Figure 2. Comparison of Algorithm 2 (marked as Cubic) with SDNA and SDCA methods for minibatch sizes $\tau = 8, 32, 256$, training Poisson regression. Left: synthetic. Right: real data.

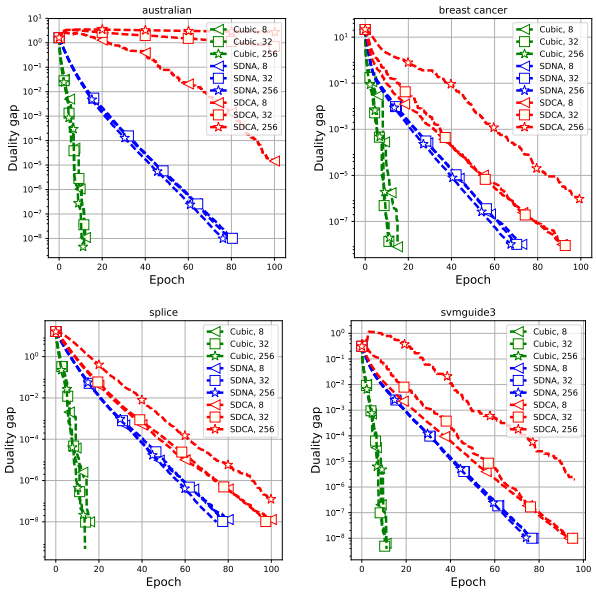

Figure 3. Comparison of Algorithm 2 (marked as Cubic) with SDNA and SDCA methods for minibatch sizes $\tau = 8, 32, 256$, training Poisson regression.

We see that in all the cases our method (Algorithm 2) outperforms state-of-the-art analogues in terms of number of data accesses.

8.4. Experimental setup

In the experiments with synthetic cubically regularized regression we generate a data in the following way: sample firstly a matrix $U \in \mathbb{R}^{10\times N}$, each entry of which is identically distributed from $\mathcal{N}(0,1)$, then put $A := U^T U \in \mathbb{R}^{N\times N}$, $b := -U^T\xi$, where $\xi \in \mathbb{R}^{10}$ is a normal vector: $\xi_i \sim \mathcal{N}(0,1)$ for each $1 \le i \le 10$, and $c_j := 1 + |v_j|$ where $v_j \sim \mathcal{N}(0,1)$ for all $1 \le j \le N$. Running the Algorithm 1 we use for H_k the known Lipschitz constants of the Hessians: $H_k := \max_{i\in S_k} c_i$.

In the experiments with ℓ_2-regularized[5] logistic regression we use the *Armijo rule* for computing step length in the Block coordinate gradient descent and the *constant choice* for the parameters H_k in the Algorithm 1 provided by Proposition 1.

In the experiments with Poisson regression for the methods SDNA and SDCA we use *damped Newton method* as a computational subroutine to compute one method step, using it until reaching 10^{-12} accuracy for the norm of the gradient in the corresponding subproblem and the same accuracy for solving inner cubic subproblem in the Algorithm 2. For the synthetic experiment with Poisson regression, we generate data matrix $B \in \mathbb{R}^{m\times d}$ as independent samples from standard normal distribution $\mathcal{N}(0,1)$ and corresponding vector of responses $y \in (\mathbb{N}\cup\{0\})^m$ from the standard Poisson distribution (in which *mean* parameter equals to 1).

[5] Regularization parameter λ in all the experiments with logistic and Poisson regression set to $1/m$.

References

Agarwal, Naman, Allen-Zhu, Zeyuan, Bullins, Brian, Hazan, Elad, and Ma, Tengyu. Finding approximate local minima for nonconvex optimization in linear time. *arXiv preprint arXiv:1611.01146*, 2016.

Bertsekas, Dimitri P. Local convex conjugacy and Fenchel duality. In *Preprints of 7th Triennial World Congress of IFAC, Helsinki, Finland*, volume 2, pp. 1079–1084, 1978.

Carmon, Yair and Duchi, John C. Gradient descent efficiently finds the cubic-regularized non-convex newton step. *arXiv preprint arXiv:1612.00547*, 2016.

Cartis, Coralia, Gould, Nicholas IM, and Toint, Philippe L. Adaptive cubic regularisation methods for unconstrained optimization. part I: motivation, convergence and numerical results. *Mathematical Programming*, 127(2):245–295, 2011a.

Cartis, Coralia, Gould, Nicholas IM, and Toint, Philippe L. Adaptive cubic regularisation methods for unconstrained optimization. part II: worst-case function-and derivative-evaluation complexity. *Mathematical programming*, 130 (2):295–319, 2011b.

Conn, Andrew R, Gould, Nicholas IM, and Toint, Philippe L. *Trust region methods*. SIAM, 2000.

Ghadimi, Saeed, Liu, Han, and Zhang, Tong. Second-order methods with cubic regularization under inexact information. *arXiv preprint arXiv:1710.05782*, 2017.

Gould, Nicholas IM, Robinson, Daniel P, and Thorne, H Sue. On solving trust-region and other regularised subproblems in optimization. *Mathematical Programming Computations*, 2(1):21–57, 2010.

Grapiglia, Geovani Nunes and Nesterov, Yu. Regularized Newton methods for minimizing functions with Hölder continuous Hessians. *SIAM Journal on Optimization*, 27 (1):478–506, 2017.

Griewank, Andreas. The modification of Newton's method for unconstrained optimization by bounding cubic terms. Technical report, Technical Report NA/12, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, 1981.

Kohler, Jonas Moritz and Lucchi, Aurelien. Sub-sampled cubic regularization for non-convex optimization. *arXiv preprint arXiv:1705.05933*, 2017.

Nesterov, Yu. Accelerating the cubic regularization of Newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008.

Nesterov, Yu. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

Nesterov, Yurii. Introductory lectures on convex optimization., 2004.

Nesterov, Yurii. Modified gauss–newton scheme with worst case guarantees for global performance. *Optimisation Methods and Software*, 22(3):469–483, 2007.

Nesterov, Yurii and Polyak, Boris T. Cubic regularization of Newton's method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

Pilanci, Mert and Wainwright, Martin J. Randomized sketches of convex programs with sharp guarantees. *IEEE Transactions on Information Theory*, 61(9):5096–5115, 2015.

Qu, Zheng and Richtárik, Peter. Coordinate descent with arbitrary sampling II: expected separable overapproximation. *Optimization Methods and Software*, 31(5):858–884, 2016.

Qu, Zheng, Richtárik, Peter, Takáč, Martin, and Fercoq, Olivier. SDNA: stochastic dual Newton ascent for empirical risk minimization. In *International Conference on Machine Learning*, pp. 1823–1832, 2016.

Richtárik, Peter and Takáč, Martin. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1-2):1–38, 2014.

Richtárik, Peter and Takáč, Martin. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, 156(1-2):433–484, 2016.

Rockafellar, R Tyrrell. Convex analysis. Princeton landmarks in mathematics, 1997.

Shalev-Shwartz, Shai and Zhang, Tong. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb): 567–599, 2013.

Tripuraneni, Nilesh, Stern, Mitchell, Jin, Chi, Regier, Jeffrey, and Jordan, Michael I. Stochastic cubic regularization for fast nonconvex optimization. *arXiv preprint arXiv:1711.02838*, 2017.

Tseng, Paul and Yun, Sangwoon. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1-2):387–423, 2009.

Appendix

A. Proof of Lemma 1

Proof: Let us show that the Hessian of $\phi_S(x)$ is Lipschitz-continuous with a constant $\max\limits_{i\in S}\{\mathcal{H}_i\}$. Then (6) will be fulfilled (see Lemma 1 in (Nesterov & Polyak, 2006)). So,

$$\begin{aligned}
\left\|\nabla^2\phi_S(x+h)-\nabla^2\phi_S(x)\right\|^2 &= \max_{\|y\|=1}\left\|\left(\nabla^2\phi_S(x+h)-\nabla^2\phi_S(x)\right)y\right\|^2 \\
&= \max_{\|y\|=1}\sum_{i\in S}\left\|\left(\nabla^2\phi_i(x_{(i)}+h_{(i)})-\nabla^2\phi_i(x_{(i)})\right)y_{(i)}\right\|^2 \\
&\overset{(5)}{\leq} \max_{\|y\|=1}\sum_{i\in S}\left(\mathcal{H}_i^2\cdot\|h_{(i)}\|^2\cdot\|y_{(i)}\|^2\right) \\
&= \max_{i\in S}\left\{\mathcal{H}_i^2\cdot\|h_{(i)}\|^2\right\} \\
&\leq \max_{i\in S}\{\mathcal{H}_i^2\}\cdot\|h_{[S]}\|^2.
\end{aligned}$$

■

B. Auxiliary results

Lemma 4 *Let Assumptions 1, 2, 3, 4 hold. Let minimum of the problem (1) exists and $x^*\in Q$ is its arbitrary solution: $F(x^*)=F^*$. Then for the random sequence $\{F(x^k)\}_{k\geq 0}$ generated by the Algorithm 1 we have a bound:*

$$\mathbb{E}\left[F(x^{k+1})\,|\,x^k\right] \leq F(x^k) - \frac{\alpha\tau}{n}\left(F(x^k)-F^*\right) - \frac{\alpha}{2}\left\langle\left(\frac{\tau}{n}\mathbf{G}-\alpha\mathbb{E}\left[\mathbf{A}_{[S_k]}\right]\right)(x^k-x^*),x^k-x^*\right\rangle \tag{19}$$

$$+\frac{\alpha^3\tau}{n}\cdot\frac{\mathcal{H}_F}{2}\cdot\|x^k-x^*\|^3, \tag{20}$$

for all $\alpha\in[0,1]$.

Proof: From the Assumption 1 we have the following bound, $\mathbf{G}\succeq 0$:

$$\langle\nabla g(x),z\rangle \leq g(x+z)-g(x)-\frac{1}{2}\langle\mathbf{G}z,z\rangle, \qquad x,z\in\mathbb{R}^N. \tag{21}$$

Denote $h^k\equiv T_{H_k,S_k}(x^k)$. Then $x^{k+1}=x^k+h^k$. By a way of choosing H_k we have:

$$\begin{aligned}
F(x^{k+1}) &\leq M^*_{H_k,S_k}(x^k) \\
&\equiv F(x^k) + \langle(\nabla g(x^k))_{[S_k]},h^k\rangle + \frac{1}{2}\langle\mathbf{A}_{[S_k]}h^k,h^k\rangle \\
&\quad + \frac{1}{2}\sum_{i\in S_k}\left(\langle\nabla\phi_i(x^k_{(i)}),h^k_{(i)}\rangle+\nabla^2\phi_i(x^k_{(i)})h^k_{(i)},h^k_{(i)}\rangle\right) + \frac{H_k}{6}\|h^k_{[S_k]}\|^3 \\
&\quad + \sum_{i\in S_k}\left(\psi_i(x^k_{(i)}+h^k_{(i)}) - \psi_i(x^k_{(i)})\right).
\end{aligned}$$

At the same time, because h^k is a minimizer of the cubic model $M_{H_k,S_k}(x^k, y)$, we can change h^k in the previous bound to arbitrary $y \in \mathbb{R}^N$, such that $x^k + y \in Q$:

$$F(x^{k+1}) \;\leq\; F(x^k) + \langle (\nabla g(x^k))_{[S_k]}, y\rangle \;+\; \frac{1}{2}\langle \mathbf{A}_{[S_k]} y, y\rangle \;+\; \sum_{i\in S_k}\Big(\langle \nabla\phi_i(x^k_{(i)}), y_{(i)}\rangle + \frac{1}{2}\langle \nabla^2\phi_i(x^k_{(i)})y_{(i)}, y_{(i)}\rangle\Big) \;+$$

$$+ \;\; \frac{H_k}{6}\|y_{[S_k]}\|^3 + \sum_{i\in S_k}\Big(\psi_i(x^k_{(i)} + y_{(i)}) - \psi_i(x^k_{(i)})\Big) \;\leq$$

$$\leq \;\; F(x^k) + \langle (\nabla g(x^k))_{[S_k]}, y\rangle \;+\; \frac{1}{2}\langle \mathbf{A}_{[S_k]} y, y\rangle \;+\; \sum_{i\in S_k}\Big(\phi_i(x^k_{(i)} + y_{(i)}) - \phi_i(x^k_{(i)})\Big) \;+$$

$$+ \;\; \frac{H_k + \mathcal{H}_F}{6}\|y_{[S_k]}\|^3 \;+\; \sum_{i\in S_k}\Big(\psi_i(x^k_{(i)} + y_{(i)}) - \psi_i(x^k_{(i)})\Big), \tag{22}$$

where the last inequality is given by Lemma 1. Note that

$$\mathbb{E}\big[\|y_{[S_k]}\|^3\big] \;\leq\; \mathbb{E}\big[\|y\|\cdot\|y_{[S_k]}\|^2\big] \;=\; \|y\|\cdot\mathbb{E}\left[\sum_{i\in S_k}\|y_{(i)}\|^2\right] \;=\; \|y\|\cdot\left(\frac{\tau}{n}\sum_{i=1}^n\|y_{(i)}\|^2\right) \;=\; \frac{\tau}{n}\|y\|^3.$$

Thus, by taking conditional expectation $\mathbb{E}\big[\cdot \,|\, x^k\big]$ from (22) we have

$$\mathbb{E}\big[F(x^{k+1}) \,|\, x^k\big] \;\leq\; F(x^k) + \frac{\tau}{n}\langle \nabla g(x^k), y\rangle \;+\; \frac{1}{2}\langle \mathbb{E}\big[\mathbf{A}_{[S_k]}\big] y, y\rangle \;+$$

$$+ \frac{\tau}{n}\sum_{i=1}^n\Big(\phi_i(x^k_{(i)} + y_{(i)}) \;+\; \psi_i(x^k_{(i)} + y_{(i)}) \;-\; \phi_i(x^k_{(i)}) \;-\; \psi_i(x^k_{(i)})\Big) \;+\; \frac{\tau}{n}\cdot\frac{\mathcal{H}_F}{2}\|y\|^3,$$

which is valid for arbitrary $y \in \mathbb{R}^N$ such that $x^k + y \in Q$. Restricting y to the segment: $y = \alpha(x^* - x^k)$, where $\alpha \in [0, 1]$, by convexity of ϕ_i and ψ_i we get:

$$\mathbb{E}\big[F(x^{k+1}) \,|\, x^k\big] \;\leq\; F(x^k) + \frac{\alpha\tau}{n}\langle \nabla g(x^k), x^* - x^k\rangle \;+\; \frac{\alpha^2}{2}\langle \mathbb{E}\big[\mathbf{A}_{S_k}\big](x^* - x^k), x^* - x^k\rangle +$$

$$+ \frac{\alpha\tau}{n}\sum_{i=1}^n\Big(\phi_i(x^*_{(i)}) + \psi_i(x^*_{(i)}) - \phi_i(x^k_{(i)}) - \psi_i(x^k_{(i)})\Big) \;+\; \frac{\alpha^3\tau}{n}\cdot\frac{\mathcal{H}_F}{2}\|x^* - x^k\|^3.$$

Finally, using inequality (21) for $x \equiv x^k$ and $z \equiv x^* - x^k$ we get the state of the lemma. ∎

The following technical tool is useful for analyzing a convergence of the random sequence $\xi_k \equiv F(x^k) - F^*$ with high probability. It is a generalization of result from (Richtárik & Takáč, 2014).

Lemma 5 *Let $\xi_0 > 0$ be a constant and consider a nonnegative nonincreasing sequence of random variables $\{\xi_k\}_{k\geq 0}$ with the following property, for all $k \geq 0$:*

$$\mathbb{E}\big[\xi_{k+1} \,|\, \xi_k\big] \;\leq\; \xi_k - \frac{\xi_k^p}{c}, \tag{23}$$

where $c > 0$ is a constant and $p \in \{3/2,\, 2\}$. Choose confidence level $\rho \in (0, 1)$.

Then if we set $0 < \varepsilon < \min\{\xi_0, c^{1/(p-1)}\}$ and

$$K \;\geq\; \frac{c}{\varepsilon^{p-1}}\left(\frac{1}{p-1} + \log\frac{1}{\rho}\right) - \frac{c}{\xi_0^{p-1}(p-1)}, \tag{24}$$

we will have

$$\mathbb{P}(\xi_K \leq \varepsilon) \;\geq\; 1 - \rho. \tag{25}$$

Proof: Technique of the proof is similar to corresponding one of the Theorem 1 from (Richtárik & Takáč, 2014). For a fixed $0 < \varepsilon < \min\{\xi_0, c^{1/(p-1)}\}$ define a new sequence of random variables $\{\xi_k^\varepsilon\}_{k\geq 0}$ by the following way:

$$\xi_k^\varepsilon \;=\; \begin{cases} \xi_k, & \text{if } \xi_k > \varepsilon, \\ 0, & \text{otherwise.} \end{cases}$$

It satisfies

$$\xi_k^\varepsilon \leq \varepsilon \quad \Leftrightarrow \quad \xi_k \leq \varepsilon, \qquad k \geq 0,$$

therefore, by Markov inequality:

$$\mathbb{P}(\xi_k > \varepsilon) \;=\; \mathbb{P}(\xi_k^\varepsilon > \varepsilon) \;\leq\; \frac{\mathbb{E}[\xi_k^\varepsilon]}{\varepsilon},$$

and hence it suffices to show that

$$\theta_K \leq \rho\varepsilon,$$

where $\theta_k \stackrel{\text{def}}{=} \mathbb{E}[\xi_k^\varepsilon]$. From the conditions of the lemma we get

$$\mathbb{E}\big[\xi_{k+1}^\varepsilon \,|\, \xi_k^\varepsilon\big] \;\leq\; \xi_k^\varepsilon - \frac{(\xi_k^\varepsilon)^p}{c}, \qquad \mathbb{E}\big[\xi_{k+1}^\varepsilon \,|\, \xi_k^\varepsilon\big] \;\leq\; \Big(1 - \frac{\varepsilon^{p-1}}{c}\Big)\xi_k^\varepsilon, \qquad k \geq 0,$$

and by taking expectations and using convexity of $t \mapsto t^p$ for $p > 1$ we obtain

$$\theta_{k+1} \quad \leq \quad \theta_k - \frac{\theta_k^p}{c}, \qquad k \geq 0, \tag{26}$$

$$\theta_{k+1} \quad \leq \quad \Big(1 - \frac{\varepsilon^{p-1}}{c}\Big)\theta_k, \qquad k \geq 0. \tag{27}$$

Consider now two cases, whether $p = 2$ or $p = 3/2$, and find a number k_1 for which we get $\theta_{k_1} \leq \varepsilon$.

1. $p = 2$, then

$$\frac{1}{\theta_{k+1}} - \frac{1}{\theta_k} \;=\; \frac{\theta_k - \theta_{k+1}}{\theta_{k+1}\theta_k} \;\geq\; \frac{\theta_k - \theta_{k+1}}{\theta_k^2} \;\stackrel{(26)}{\geq}\; \frac{1}{c},$$

   thus we have $\frac{1}{\theta_k} \geq \frac{1}{\theta_0} + \frac{k}{c} = \frac{1}{\xi_0} + \frac{k}{c}$, and choosing $k_1 \geq \frac{c}{\varepsilon} - \frac{c}{\xi_0}$ we obtain $\theta_{k_1} \leq \varepsilon$.

2. $p = 3/2$, then

$$\frac{1}{\theta_{k+1}^{1/2}} - \frac{1}{\theta_k^{1/2}} \;=\; \frac{\theta_k^{1/2} - \theta_{k+1}^{1/2}}{\theta_{k+1}^{1/2}\theta_k^{1/2}} \;=\; \frac{\theta_k - \theta_{k+1}}{(\theta_k^{1/2} + \theta_{k+1}^{1/2})\theta_{k+1}^{1/2}\theta_k^{1/2}} \;\geq\; \frac{\theta_k - \theta_{k+1}}{2\theta_k^{3/2}} \;\stackrel{(26)}{\geq}\; \frac{1}{2c},$$

   thus we have $\frac{1}{\theta_k^{1/2}} \geq \frac{1}{\theta_0^{1/2}} + \frac{k}{2c}$, and choosing $k_1 \geq \frac{2c}{\varepsilon^{1/2}} - \frac{2c}{\xi_0^{1/2}}$ we get $\theta_{k_1} \leq \varepsilon$.

Therefore, for both cases $p \in \{3/2, 2\}$ it is enough to choose

$$k_1 \quad \geq \quad \frac{c}{p-1}\Big(\frac{1}{\varepsilon^{p-1}} - \frac{1}{\xi_0^{p-1}}\Big),$$

for which we get $\theta_{k_1} \leq \varepsilon$. Finally, letting $k_2 \geq \frac{c}{\varepsilon^{p-1}} \log\frac{1}{\rho}$ and $K \geq k_1 + k_2$, we have

$$\theta_K \;\leq\; \theta_{k_1+k_2} \;\stackrel{(27)}{\leq}\; \Big(1 - \frac{\varepsilon^{p-1}}{c}\Big)^{k_2}\theta_{k_1} \;\leq\; \exp\Big(-\frac{k_2\varepsilon^{p-1}}{c}\Big)\theta_{k_1} \;\leq\; \rho\varepsilon.$$

■

C. Proof of Theorem 1

Proof: From the bound (19), using $\mathbf{G} \succeq 0$ and $\mathbb{E}\big[\mathbf{A}_{[S_k]}\big] \preceq \frac{\tau L}{n}\mathbf{I}$ we get, for all $\alpha \in [0,1]$:

$$\mathbb{E}\big[F(x^{k+1}) \,|\, x^k\big] \;\;\leq\;\; F(x^k) - \frac{\alpha\tau}{n}\Big(F(x^k) - F^*\Big) + \frac{\alpha^2\tau}{2n}\left(L\|x^k - x^*\|^2 + \alpha\mathcal{H}_F\|x^k - x^*\|^3\right).$$

Thus, for the random sequence $\xi_k \equiv F(x^k) - F^*$ we have a bound, for all $\alpha \in [0,1]$:

$$\mathbb{E}\big[\xi_{k+1} \,|\, \xi_k\big] \;\leq\; \xi_k - \frac{\alpha\tau\xi_k}{n} + \frac{\alpha^2\tau}{2n}D_0, \tag{28}$$

where $D_0 \equiv \max\big\{LD^2 + \mathcal{H}_F D^3,\, \xi_0\big\}$. Minimum of the right hand side is attained at $\alpha^* = \frac{\xi_k}{D_0} \leq 1$, which substituting to (28) gives: $\mathbb{E}[\xi_{k+1}|\xi_k] \;\leq\; \xi_k - \frac{\tau\xi_k^2}{2nD_0}$. Applying Lemma 5 complete the proof. ■

D. Proof of Theorem 2

Proof: From the bound (19), restricting α to the segment $[0,\sigma]$ and using (12) we get:

$$\mathbb{E}\big[\xi_{k+1} \,|\, \xi_k\big] \;\leq\; \xi_k - \frac{\alpha\tau\xi_k}{n} + \frac{\alpha^3\tau}{2n}\mathcal{H}_F\|x^k - x^*\|^3, \tag{29}$$

where $\xi_k \equiv F(x^k) - F^*$ as before. To get the first complexity bound, we rough the right hand side, denoting $D_0 \equiv \max\big\{\mathcal{H}_F D^3, \xi_0\big\}$:

$$\mathbb{E}\big[\xi_{k+1} \,|\, \xi_k\big] \;\leq\; \xi_k - \frac{\alpha\tau\xi_k}{n} + \frac{\alpha^3\tau}{2n}\frac{D_0}{\sigma^2},$$

minimum of which is attained at $\alpha^* = \sigma\sqrt{\frac{2}{3}\frac{\xi_k}{D_0}} \leq \sigma$. Therefore we obtain

$$\mathbb{E}[\xi_{k+1}|\xi_k] \;\leq\; \xi_k - (2/3)^{3/2}\frac{\tau\sigma\xi_k^{3/2}}{nD_0},$$

and applying Lemma 5 complete the proof. ■

E. Proof of Theorem 3

Proof: Because of strong convexity, we know $\beta > 0$ and all the conditions of the Theorem 2 are satisfied. Using inequality $F(x^k) - F^* \geq \frac{\mu}{2}\|x^k - x^*\|^2$ for (29), we have for every $\alpha \in [0,\sigma]$:

$$\mathbb{E}\big[\xi_{k+1} \,|\, \xi_k\big] \;\leq\; \left(1 - \frac{\alpha\tau}{n} + \frac{\alpha^3\tau}{n}\frac{\mathcal{H}_F D}{\mu}\right)\xi_k.$$

Minimum of the right hand side is attained at $\alpha^* = \sigma\min\Big\{\sqrt{\frac{\mu}{3\mathcal{H}_F D}}, 1\Big\}$, substituting of which and taking total expectation gives a recurrence

$$\begin{aligned}
\mathbb{E}\big[\xi_{k+1}\big] &\leq \left(1 - \frac{2\tau\sigma}{3n}\sqrt{\min\Big\{\frac{\mu}{\mathcal{H}_F D}, 1\Big\}}\right)\mathbb{E}[\xi_k] \\
&\leq \;\ldots \\
&\leq \exp\left(-(k+1)\frac{2\tau\sigma}{3n}\sqrt{\min\Big\{\frac{\mu}{\mathcal{H}_F D}, 1\Big\}}\right)\xi_0.
\end{aligned}$$

Thus, choosing K large enough, by Markov inequality we have $\mathbb{P}(\xi_K > \varepsilon) \leq \mathbb{E}[\xi_K]\varepsilon^{-1} \overset{(14)}{\leq} \rho$. ■

F. Proof of Lemma 3

Using notation from the statement of the lemma and multiplying everything by m, we can formulate our target optimization subproblem as follows:

$$\min_{\substack{x\in\mathbb{R}^{|S|},h\in\mathbb{R}^m \\ \text{s.t. } h=\hat{\mathbf{B}}x}} \left[m\lambda\langle b_1, x\rangle \;+\; \frac{m\lambda}{2}\langle \hat{\mathbf{A}}x, x\rangle \;+\; \langle b_2, h\rangle \;+\; \frac{1}{2}\langle \mathbf{D}h, h\rangle \;+\; \frac{H}{6}\|h\|^3 \right], \tag{30}$$

where $\mathbf{D}\in\mathbb{R}^{m\times m}$ is a diagonal matrix: $\mathbf{D}\equiv \mathrm{diag}(\phi_i''(\alpha_i))$. We also denote a subvector of y as x to avoid confusion. The minimum of (30) satisfies the following KKT system:

$$\begin{cases} m\lambda b_1 \;+\; m\lambda\hat{\mathbf{A}}x \;+\; \hat{\mathbf{B}}^T\mu \;=\; 0, \\ b_2 \;+\; \mathbf{D}h \;+\; \frac{H}{2}\|h\|h \;-\; \mu \;=\; 0, \\ h \;=\; \hat{\mathbf{B}}x, \end{cases} \tag{31}$$

where $\mu\in\mathbb{R}^m$ is a vector of slack variables. From the second and the third equations we get:

$$\mu \;=\; b_2 \;+\; \mathbf{D}\hat{B}x \;+\; \frac{H}{2}\|\hat{\mathbf{B}}x\|\hat{\mathbf{B}}x,$$

plugging of which into the first one gives:

$$\underbrace{\left(m\lambda\hat{\mathbf{A}} \;+\; \hat{\mathbf{B}}^T\left(\mathbf{D}+\frac{H}{2}\|\hat{\mathbf{B}}x\|\right)\hat{\mathbf{B}}\right)}_{\equiv\, \mathbf{Z}(\|\hat{\mathbf{B}}x\|)} x \;=\; -\underbrace{(m\lambda b_1 + \hat{\mathbf{B}}^T b_2)}_{\equiv\, b}.$$

Thus, if we have a solution $\tau^*\geq 0$ of the one-dimensional equation:

$$\tau^* \;=\; \|\hat{\mathbf{B}}(\mathbf{Z}(\tau^*))^\dagger b\|,$$

then we can set

$$x^* \;:=\; -(\mathbf{Z}(\tau^*))^\dagger b, \qquad h^* \;:=\; \hat{\mathbf{B}}x^*, \qquad \mu^* \;:=\; b_2 + \mathbf{D}\hat{\mathbf{B}}x^* + \frac{H}{2}\|\hat{\mathbf{B}}x^*\|\hat{\mathbf{B}}x^*.$$

It is easy to check that (x^*, h^*, μ^*) are solutions of (31) and therefore of (30) as well.

G. Lipschitz constant of the Hessian of logistic loss

Proposition 1 *Loss function for logistic regression $\phi(t) := \log(1+\exp(t))$ has Lipschitz-continuous Hessian with constant $\mathcal{H}_\phi = 1/(6\sqrt{3})$. Thus, it holds, for all $t, s\in\mathbb{R}$:*

$$|\phi''(t) - \phi''(s)| \quad\leq\quad \mathcal{H}_\phi |t-s|. \tag{32}$$

Proof:

To prove (32) it is enough to show: $|\phi'''(t)|\leq \mathcal{H}_\phi$, for all $t\in\mathbb{R}$. Direct calculations give:

$$\phi'(t) \;=\; \frac{1}{1+\exp(-t)}, \qquad \phi''(t) \;=\; \phi'(t)\cdot(1-\phi'(t)), \qquad \phi'''(t) \;=\; \phi''(t)\cdot(1-2\phi'(t)).$$

Let us find extreme values of the function $g(t) := \phi'''(t)$ for which we have $\lim\limits_{t\to-\infty} g(t) \;=\; \lim\limits_{t\to+\infty} g(t) = 0$.

Stationary points of $g(t)$ are solutions of the equation

$$g'(t^*) \;=\; \phi^{(4)}(t^*) \;=\; \phi''(t^*)\cdot\left[(1-2\phi'(t^*))^2 - 2\phi'(t^*)\cdot(1-\phi'(t^*))\right] \;=\; 0$$

which consequently should satisfy $\phi'(t^*) = \frac{1}{2} \pm \frac{1}{\sqrt{12}}$ and therefore:

$$g(t^*) \;=\; \phi'''(t^*) \;=\; \left(\frac{1}{2}+\frac{1}{\sqrt{12}}\right)\cdot\left(\frac{1}{2}-\frac{1}{\sqrt{12}}\right)\cdot\left(\pm\frac{1}{\sqrt{3}}\right) \;=\; \pm\frac{1}{6\sqrt{3}},$$

from what we get: $|\phi'''(t)|\leq 1/(6\sqrt{3})$. ■