IMA Journal of Numerical Analysis (2019) **00**, 1–33
doi:10.1093/imanum/drz027

Subsampled inexact Newton methods for minimizing large sums of convex functions

Stefania Bellavia*
Department of Industrial Engineering, University of Florence, Viale Morgagni, 40/44, Florence 50134, Italy
*Corresponding author: stefania.bellavia@unifi.it

and

Nataša Krejić and Nataša Krklec Jerinkić
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, Novi Sad 21000, Serbia
natasak@uns.ac.rs natasa.krklec@dmi.uns.ac.rs

[Received on 23 January 2018; revised on 1 May 2019]

This paper deals with the minimization of a large sum of convex functions by inexact Newton (IN) methods employing subsampled functions, gradients and Hessian approximations. The conjugate gradient method is used to compute the IN step and global convergence is enforced by a nonmonotone line-search procedure. The aim is to obtain methods with affordable costs and fast convergence. Assuming strongly convex functions, R-linear convergence and worst-case iteration complexity of the procedure are investigated when functions and gradients are approximated with increasing accuracy. A set of rules for the forcing parameters and subsample Hessian sizes are derived that ensure local q-linear/q-superlinear convergence of the proposed method. The random choice of the Hessian subsample is also considered and convergence in the mean square, both for finite and infinite sums of functions, is proved. Finally, the analysis of global convergence with asymptotic R-linear rate is extended to the case of the sum of convex functions and strongly convex objective function. Numerical results on well-known binary classification problems are also given. Adaptive strategies for selecting forcing terms and Hessian subsample size, streaming out of the theoretical analysis, are employed and the numerical results show that they yield effective IN methods.

Keywords: inexact Newton; subsampled Hessian; superlinear convergence; global convergence; mean square convergence.

1. Introduction

The problem we consider is

$$\min f_{\mathcal{N}}(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x) \tag{1.1}$$

with $x \in \mathbb{R}^n$, N very large and all functions $f_i : \mathbb{R}^n \to \mathbb{R}$ convex and $f_{\mathcal{N}}$ strongly convex. We are also interested in the case of large dimension n. There are a number of important problems of this type. To start with, one can be interested in minimizing the objective function stated in the form of a mathematical expectation, $f(x) = E[F(x, w)]$, with w being a random variable from some probability space. Given

that the analytical expression for mathematical expectation is rarely available, one possibility is to approximate the expectation with the sample average approximation (SAA) function. In that case, a sample $\{w^1, \ldots, w^N\}$ is generated and the approximate objective function of the form (1.1) with $f_i(x) = F(x, w^i)$ is minimized. To ensure good approximation of the original objective function in general, one has to take a very large sample and thus calculating $f_{\mathcal{N}}(x)$, its gradient and Hessian is expensive.

Binary and multiclass classification problems, e.g., employing the softmax activation function and cross-entropy loss can also be expressed in the form (1.1). For a given (very large) set of data, we are interested in classifying the data according to a set of rules specified by the data features.

In the framework of classical optimization, (1.1) is a convex problem that can be solved by either a first-order or a second-order method. However, the size of N makes classical approaches prohibitively costly and thus the calls for specific methods. One possibility is to consider different subsampling schemes which are used to reduce the cost of calculating $f_{\mathcal{N}}$, its gradient and Hessian. There are many approaches in the literature, based on the idea of using a small sample subset at the beginning of the iterative process and increasing the sample size as the solution is approached, ranging from a simple heuristic approach (Byrd *et al.*, 2012; Friedlander & Schmidt, 2012) to elaborate schemes that take into account the progress achieved up to the current iteration (Bastin, 2004; Bastin *et al.*, 2006a,b; Birgin *et al.*, 2018; Polak & Royset, 2008; Deng & Ferris, 2009; Pasupathy, 2010; Krejić & Krklec, 2013; Krejić & Krklec Jerinkić, 2015; Krejić & Martínez, 2016).

Whatever scheduling one adopts, the next question to be discussed is the choice of method. First-order methods are attractive due to their low cost. One successful example is the stochastic gradient method that employs a smaller subset of gradient components and thus reduces the cost even further (Friedlander & Schmidt, 2012). On the other hand, several papers investigate the use of second-order methods in this framework and demonstrate advantages in some important problems if the second-order methods are correctly implemented (Byrd *et al.*, 2011, Byrd *et al.*, 2016; Erdogdu & Montanari, 2015; Xu *et al.*, 2016, Xu *et al.*, 2018; Pilanci & Wainwright, 2017; Bellavia *et al.*, 2018; Berahas *et al.*, 2018; Bollapragada *et al.*, 2018; Roosta-Khorasani & Mahoney, 2019). For a comprehensive discussion of this issue one can see Bottou *et al.* (2017) and references cited therein.

In this paper we focus on subsampled inexact Newton (IN) methods for (1.1) wrapped in a nonmonotone line-search strategy. In IN methods (Dembo *et al.*, 1982; Nash, 2000) the Newton equation is approximately solved and in the case of large-scale problems an iterative Krylov method is used to compute an approximate solution of the Newton equation. The convexity of the objective function allows us to use the conjugate gradient (CG) method (Kelley, 1995).

The choice of the nonmonotone strategy is motivated by the fact that the method uses approximate functions, at least initially, before the full sample is reached. Then, enforcing a strict decrease in the Armijo rule might require unnecessarily small steps. We adopt the nonmonotone line-search procedure introduced in Li & Fukushima (2000) and, assuming that each of the functions f_i is strongly convex, we prove R-linear convergence. Also the worst-case iteration complexity is investigated and it is proved that the worst-case complexity bound of this class of nonmonotone algorithms, analyzed in Grapiglia & Sachs (2017), is maintained provided that errors in gradient and function also decay with the R-rate. Namely the method requires at most $\mathcal{O}(\log(\epsilon^{-1}))$ iterations to reach $f(x^k) - f(x^*) < \epsilon$, where x^* is the minimizer of (1.1).

Then we turn our attention to the local properties of the method to obtain a local convergence rate faster than the R-linear convergence provided by first-order methods. The local convergence rate of IN methods with full Hessian depends on the choice of forcing terms that govern the error in solving each Newton linear system (Eisenstat & Walker, 1996). Here, as the Hessian of the objective function given in (1.1) might be prohibitively expensive to compute, we concentrate on the subsampled Hessian and on

the IN method that employs such Hessian approximations. We point out that in this context it is pointless to solve the Newton equation exactly as the Hessian is generally approximated with a lower accuracy than the function and gradient and the Newton model employed is actually a subsampled Newton model. Therefore, the use of the CG method, which allows to control the accuracy in the solution of the Newton equation, is advisable even if n is not large (Byrd *et al.*, 2012).

Assuming that the sample size scheduling is given for the objective function and the gradient, i.e., assuming that eventually one reaches the full sample size N, we analyze the local convergence of a subsampled Hessian IN method. The analysis provides bounds on the Hessian accuracy requirements that depend on the employed forcing terms. Adaptive forcing terms streaming out of the iterative process itself are derived as well. Furthermore, it is shown that the local method combines well with the non-monotone line search, i.e., the q-linear/q-superlinear convergence rate of the local method is preserved.

In the second part of the paper we consider a randomized method obtained by relaxing the conditions for Hessian subsampling. Hence, we prove the q-linear/q-superlinear convergence in the mean sense assuming that the Hessian approximation is good enough with high probability. The analysis yields the relation between the Hessian subsample size, the forcing term and the (computable) sampled gradient at each iteration. Q-linear convergence in the mean sense is proved for fixed forcing terms with a fixed Hessian subsample size, while superlinear convergence in the mean square (m.s.) is obtained for the forcing terms that approach zero and increasing Hessian subsample sizes.

Having in mind the binary classification problem and the fact that the number of training points is enlarged over time in many applications, we also consider the case of unbounded N, i.e., the case where the objective function is defined as the mathematical expectation. For this problem we obtain convergence in the m.s. as well.

Finally, the strong convexity assumption is relaxed similarly to the problem considered in Roosta-Khorasani & Mahoney (2019). A bound on the Hessian sample size, derived in Xu *et al.* (2018), is used to obtain Hessian approximations that are positive definite with some high probability and CG is adapted to deal with possibly singular problems. The convergence in the mean square is obtained for this problem as well.

From the performed theoretical analysis we derive adaptive rules for selecting both the forcing term sequence and the Hessian sample size. A particularly important feature of the proposed method is that the Hessian sample size is related to the current forcing term and approximated gradient norm, both quantities actually computable. Moreover, when q-superlinear convergence is sought, the Hessian sample size is adaptively chosen along the iterations; low accuracy and smaller Hessian sample size are generally used in the early stage of the method while the accuracy and the Hessian sample size increase in the last stage of the convergence. Finally, we note that the Hessian sample size is also allowed to decrease if too high accuracy is used at the previous iteration. Numerical results on binary classification problems give numerical evidence of the effectiveness of the proposed adaptive choices.

The paper is organized as follows. In Section 2 the method is introduced and the global and local analysis is performed using the standard deterministic reasoning for the case of strongly convex functions f_i. Convergence in the mean square is proved in Section 3, considering all three cases—a finite number of strongly convex functions f_i, an infinite number of strongly convex functions f_i and the last case with relaxed convexity assumptions. Some numerical results are presented in Section 4.

1.1 *Related work*

Our analysis is strictly related to that developed in Berahas *et al.* (2018), Bollapragada *et al.* (2018) and Roosta-Khorasani & Mahoney (2019), where convergence of inexact subsampled Newton methods

is investigated both in probability (Roosta-Khorasani & Mahoney, 2019) and expectation (Byrd *et al.*, 2012; Berahas *et al.*, 2018; Bollapragada *et al.*, 2018). We focus on the choice of the forcing terms, on the nonmonotone line-search strategy and on adaptive choices of Hessian sample size.

The local behavior of subsampled IN methods has been analyzed in Bollapragada *et al.* (2018) and Roosta-Khorasani & Mahoney (2019). Bounds to the accuracy required in the last stage of the procedure have been given, but the issue of developing an automatic transition between the initial stage of the procedure where a low accuracy in the Hessian approximation is enough, and the last stage where more accurate Hessian approximations are needed, is not investigated. However, in Roosta-Khorasani & Mahoney (2019) the analysis is carried out under weaker assumptions than those we used here as the function $f_{\mathcal{N}}$ is supposed to be strongly convex only in a neighborhood of the sought minimizer, without any assumptions on the convexity of functions f_i. A set of conditions on the gradient and the Hessian sample sizes that ensure local R-superlinear convergence in the expectation under the assumption on the variance of the error norms (bounded moment of iterates) is given in Bollapragada *et al.* (2018). The Hessian sample size is assumed to increase at each iteration, starting from a large enough initial sample size. In Byrd *et al.* (2012) an adaptive rule for choosing the gradient sample size is proposed, along with an automatic criterion for the forcing term related to a variance estimation of the Hessian accuracy. The Hessian sample size is a fixed fraction of the used gradient sample size.

Finally, we would like to mention that in Berahas *et al.* (2018) the authors perform a local complexity analysis of subsampled IN methods and also show that methods that incorporate stochastic second-order information can be far more efficient on badly scaled or ill-conditioned problems than first-order methods.

2. Inexact subsampled Newton method

We first introduce the notation and give some preliminary results. Throughout the paper $\mathcal{N}_k \subset \{1, 2, \ldots, N\}$ denotes the sample used to approximate the objective function and its gradient, N_k denotes its cardinality and the subsampled function and gradient are defined as

$$f_{\mathcal{N}_k}(x) = \frac{1}{N_k} \sum_{i \in \mathcal{N}_k} f_i(x), \quad \nabla f_{\mathcal{N}_k} = \frac{1}{N_k} \sum_{i \in \mathcal{N}_k} \nabla f_i(x).$$

Moreover, $\mathcal{D}_k \subset \{1, 2, \ldots, N\}$ is the sample used for Hessian approximation with cardinality D_k, and the subsampled Hessian is given by

$$\nabla^2 f_{\mathcal{D}_k}(x) = \frac{1}{D_k} \sum_{i \in \mathcal{D}_k} \nabla^2 f_i(x). \tag{2.1}$$

Here we will consider subsampled IN methods, that is, iterative processes where at iteration k, given the current iterate x^k, the step s^k used to update the iterate satisfies

$$\nabla^2 f_{\mathcal{D}_k}(x^k) s^k = -\nabla f_{\mathcal{N}_k}(x^k) + r^k, \quad \|r^k\| \leq \eta_k \|\nabla f_{\mathcal{N}_k}(x^k)\|. \tag{2.2}$$

The term η_k belongs to $(0, 1)$ and it is called the forcing term (Dembo *et al.*, 1982; Nash, 2000). Here and in the rest of the paper $\|\cdot\|$ denotes the 2-norm.

Throughout the paper we will restrict our attention to convex functions; more precisely, we will first consider strongly convex functions and then in Section 3.2 relax the strong convexity to convexity. Let us state this formally.

ASSUMPTION 2.1 The functions f_i, $i = 1, \ldots, N$ are twice continuously differentiable and strongly convex, i.e., for some $\lambda_n \geq \lambda_1 > 0$ there holds

$$\lambda_1 I \preceq \nabla^2 f_i(x) \preceq \lambda_n I \quad \forall\, x \in \mathbb{R}^n, \quad i = 1, \ldots, N. \tag{2.3}$$

Assumption 2.1 implies a couple of inequalities that will be used further on. First of all, for all $x \in \mathbb{R}^n$ we have

$$\lambda_1 \|x - x^*\| \leq \|\nabla f_{\mathcal{N}}(x)\| \leq \lambda_n \|x - x^*\|, \tag{2.4}$$

where x^* is the unique minimizer of the function $f_{\mathcal{N}}$. Furthermore, according to Nesterov (2013, Theorem 2.10), for every $x \in \mathbb{R}^n$,

$$\frac{\lambda_1}{2}\|x - x^*\|^2 \leq f_{\mathcal{N}}(x) - f_{\mathcal{N}}(x^*) \leq \frac{1}{\lambda_1}\|\nabla f_{\mathcal{N}}(x)\|^2. \tag{2.5}$$

Since $\nabla^2 f_{\mathcal{D}}(x)$ is positive definite we choose CG as the linear solver for computing s^k in (2.2). Thus, we assume that CG initialized with the zero vector is employed at each IN iteration. We will use the following technical lemma from Fountoulakis & Godzio (2016).

LEMMA 2.2 (Fountoulakis & Godzio, 2016). Let $Ax = b$, where A is a symmetric and positive definite matrix. Furthermore, let us assume that CG is applied to this system and it is terminated at the ith iteration. Then if CG is initialized with the null vector the approximate solution x_i satisfies $x_i^{\mathrm{T}} A x_i = x_i^{\mathrm{T}} b$.

Now, the above lemma and Assumption 2.1 clearly imply the following result.

LEMMA 2.3 Assume that s^k satisfying (2.2) is obtained through the CG method initialized with the null vector applied to the linear system

$$\nabla^2 f_{\mathcal{D}_k}(x^k) s = -\nabla f_{\mathcal{N}_k}(x^k).$$

Then $\|s^k\| \leq \lambda_1^{-1} \|\nabla f_{\mathcal{N}_k}(x^k)\|$.

2.1 *Global convergence*

In this section we analyze the behavior of the subsampled IN method and CG as the inner solver wrapped in the nonmonotone line-search strategy given in Li & Fukushima (2000). Iteration k of this procedure, denoted Algorithm GIN, is sketched in Algorithm 2.1. The nonmonotone line search has been chosen because in the first stage of the procedure subsampled functions and gradients are used and thus a strict Armijo-type decrease might yield unnecessarily small steps without a real decrease in the original objective function. An additional freedom in the step-size selection is therefore introduced, adding to the Armijo condition a positive term ν_k usually denoted as the error term (Li & Fukushima, 2000). The error sequence $\{\nu_k\}$ has to satisfy the following properties:

$$\nu_k > 0, \quad \sum_{k}^{\infty} \nu_k < \infty. \tag{2.6}$$

Algorithm 2.1 kth iteration of Algorithm GIN

Given $x^k \in \mathbb{R}^n$ $c \in (0,1)$, $\bar{\eta} :\in (0,1)$ and $\{\nu_k\}$ such that (2.6) holds.

Step 1. Choose $\mathcal{N}_k$, $\mathcal{D}_k$,$\eta_k \in (0, \bar{\eta})$.

Step 2. Apply the CG method initialized by the null vector to $\nabla^2 f_{\mathcal{D}_k}(x^k)s^k = -\nabla f_{\mathcal{N}_k}(x^k)$ and compute s^k satisfying (2.2).

Step 3. Find the smallest non-negative integer j such that for $t_k = 2^{-j}$ there holds

$$f_{\mathcal{N}_k}(x^k + t_k s^k) \leq f_{\mathcal{N}_k}(x^k) + ct_k(s^k)^{\mathrm{T}}\nabla f_{\mathcal{N}_k}(x^k) + \nu_k \tag{2.7}$$

and set $x^{k+1} = x^k + t_k s^k$, $k = k + 1$.

Algorithm GIN is stated with arbitrary scheduling sequences $\{N_k\}$ and $\{D_k\}$. The line-search rule is defined with $f_{\mathcal{N}_k}$ implying that inexact function values, as well as approximated gradient values, are allowed in Algorithm GIN. Naturally, one expects to save computational effort while working with smaller samples, before reaching the full sample at some iteration. Complexity analysis and global convergence of this algorithm are presented. First we give the iteration complexity result for an arbitrary schedule for the gradient approximation considering the decrease in (possibly inexact) gradient. Then we will prove R-linear convergence and show that the classical complexity result of $\mathcal{O}(\log(\epsilon^{-1}))$ (Grapiglia & Sachs, 2017) is obtained for a schedule that eventually ends up with the full sample.

As the search direction s^k is generated by the CG method and $f_{\mathcal{N}_k}$ is strictly convex we know by Lemma 2.2 that s^k is the descent direction for $f_{\mathcal{N}_k}$ at x^k and thus, under Assumption 2.1, the step size t_k is strictly positive even for the standard Armijo rule in Step 3. The lower bound for t_k is obtained in Lemma 2.4, similarly to Krejić & Krklec Jerinkić (2015).

LEMMA 2.4 Let s^k be the step generated in Step 2 of Algorithm GIN. Then

$$-(\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}}s^k \geq \frac{\lambda_1}{\lambda_n^2}(1-\eta_k)^2\|\nabla f_{\mathcal{N}_k}(x^k)\|^2.$$

Proof. The inexact condition implies

$$\|\nabla f_{\mathcal{N}_k}(x^k) - r^k\| \geq \|\nabla f_{\mathcal{N}_k}(x^k)\| - \|r^k\| \geq (1-\eta_k)\|\nabla f_{\mathcal{N}_k}(x^k)\|.$$

On the other hand, we have

$$\|\nabla f_{\mathcal{N}_k}(x^k) - r^k\| = \|\nabla^2 f_{\mathcal{D}_k}(x^k)s^k\| \leq \|\nabla^2 f_{\mathcal{D}_k}(x^k)\|\|s^k\| \leq \lambda_n\|s^k\|$$

due to (2.3). Therefore,

$$\|s^k\| \geq \frac{\|\nabla f_{\mathcal{N}_k}(x^k) - r^k\|}{\lambda_n} \geq \frac{1-\eta_k}{\lambda_n}\|\nabla f_{\mathcal{N}_k}(x^k)\|. \tag{2.8}$$

Then Lemma 2.2 and (2.3) yield

$$-(\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}} s^k = (s^k)^{\mathrm{T}} \nabla^2 f_{\mathcal{D}_k}(x^k) s^k \geq \lambda_1 \|s^k\|^2 \tag{2.9}$$

$$\geq \frac{\lambda_1}{\lambda_n^2}(1-\eta_k)^2 \|\nabla f_{\mathcal{N}_k}(x^k)\|^2. \tag{2.10}$$

□

LEMMA 2.5 Suppose that Assumption 2.1 holds and let s^k be generated in Step 2 of Algorithm GIN. Then (2.7) holds for $t_k \geq \bar{t} = (1-c)\lambda_1/\lambda_n$.

Proof. Let k be an arbitrary iteration. If $t_k = 1$ satisfies (2.7), that is, in Step 3 we have $j = 0$, then t_k is greater than $\bar{t}$, as $\bar{t} \in (0,1)$. So let us consider the case $t_k < 1$. Then there exists $t_k' = 2t_k$ such that

$$\begin{aligned} f_{\mathcal{N}_k}(x^k + t_k' s^k) &> f_{\mathcal{N}_k}(x^k) + c t_k' (s^k)^{\mathrm{T}} \nabla f_{\mathcal{N}_k}(x^k) + \nu_k \\ &\geq f_{\mathcal{N}_k}(x^k) + c t_k' (s^k)^{\mathrm{T}} \nabla f_{\mathcal{N}_k}(x^k). \end{aligned}$$

On the other hand, Assumption 2.1 implies, using the standard arguments for functions with bounded Hessians,

$$\begin{aligned} f_{\mathcal{N}_k}(x^k + t_k' s^k) &= f_{\mathcal{N}_k}(x^k) + \int_0^1 (\nabla f_{\mathcal{N}_k}(x^k + y t_k' s^k))^{\mathrm{T}} (t_k' s^k) \mathrm{d}y \\ &\leq \frac{\lambda_n}{2}(t_k')^2 \|s^k\|^2 + f_{\mathcal{N}_k}(x^k) + t_k' (\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}} s^k. \end{aligned}$$

Combining the previous two inequalities we obtain

$$c t_k' (s^k)^{\mathrm{T}} \nabla f_{\mathcal{N}_k}(x^k) \leq \frac{\lambda_n}{2}(t_k')^2 \|s^k\|^2 + t_k' (\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}} s^k.$$

Dividing by t_k' and using $t_k = t_k'/2$, by rearranging the previous inequality we get

$$t_k \geq \frac{-(1-c)(\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}} s^k}{\lambda_n \|s^k\|^2}. \tag{2.11}$$

Now the result follows from (2.9) and $\min\{1, (1-c)\lambda_1/\lambda_n\} = (1-c)\lambda_1/\lambda_n$. □

To prove the main results we need the following lemma from Krejić & Krklec Jerinkić (2015).

LEMMA 2.6 Assume that $\zeta_k \to 0$ R-linearly. Then for every $\rho \in (0,1)$ the sequence $\{a_k\}$ such that $a_k = \sum_{j=1}^{k} \rho^{j-1} \zeta_{k-j}$ converges to zero R-linearly.

Let us denote by ξ_k^g and ξ_k^f the inaccuracy in the function and gradient:

$$\max_{x\in\{x^k,x^{k+1}\}} |f_{\mathcal{N}_k}(x) - f_{\mathcal{N}}(x)| \le \xi_k^f, \quad \left|\|\nabla f_{\mathcal{N}_k}(x^k)\|^2 - \|\nabla f_{\mathcal{N}}(x^k)\|^2\right| \le \xi_k^g. \tag{2.12}$$

Following Grapiglia & Sachs (2017) we now prove that, despite the inaccuracy in function and gradient, Algorithm GIN meets the complexity results of nonmonotone line-search methods with exact function and gradients.

THEOREM 2.7 Suppose that Assumption 2.1 holds and

$$\sum_{k=0}^{\infty} \xi_k^f < \infty. \tag{2.13}$$

Then for a given $\epsilon \in (0,1)$ Algorithm GIN takes at most

$$\bar{k} = \left\lceil \frac{f_{\mathcal{N}}(x^0) - f_{\mathcal{N}}(x^*) + \sum_{k=0}^{\infty}(2\xi_k^f + \nu_k)}{\kappa_c} \epsilon^{-2} \right\rceil,$$

iterations to ensure $\|\nabla f_{\mathcal{N}_{\bar{k}}}(x^{\bar{k}})\| \le \epsilon$, where

$$\kappa_c = c(1-c)\frac{\lambda_1^2}{\lambda_n{}^3}(1-\bar{\eta})^2. \tag{2.14}$$

Proof. Note that by Lemma 2.4 there follows

$$(\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}} s^k \le -\frac{\lambda_1}{\lambda_n^2}(1-\bar{\eta})^2 \|\nabla f_{\mathcal{N}_k}(x^k)\|^2. \tag{2.15}$$

Then we can proceed as in Grapiglia & Sachs (2017, proof of Theorem 1). Let $\bar{k}$ be the first iteration such that $\|\nabla f_{\mathcal{N}_{\bar{k}}}(x^{\bar{k}})\| \le \epsilon$. By (2.7) we obtain

$$\nu_k + f_{\mathcal{N}_k}(x^k) - f_{\mathcal{N}_k}(x^{k+1}) \ge -ct_k(\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}} s^k, \quad k = 0,1,\ldots,\bar{k}-1.$$

Moreover, by (2.15) and Lemma 2.5, there follows

$$-ct_k(\nabla f_{\mathcal{N}_k}(x^k))^{\mathrm{T}} s^k \ge c(1-c)\frac{\lambda_1^2}{\lambda_n{}^3}(1-\bar{\eta})^2 \|\nabla f_{\mathcal{N}_k}(x^k)\|^2.$$

Then, for $k = 0,1,\ldots,\bar{k}-1$, there holds

$$\nu_k + f_{\mathcal{N}_k}(x^k) - f_{\mathcal{N}_k}(x^{k+1}) \ge \kappa_c \epsilon^2$$

with κ_c given in (2.14). Therefore, by (2.12),

$$\nu_k + 2\xi_k^f + f_{\mathcal{N}}(x^k) - f_{\mathcal{N}}(x^{k+1}) \geq \kappa_c \epsilon^2, \qquad k = 0, \ldots, \bar{k} - 1.$$

Summing for $k = 0, \ldots, \bar{k} - 1$ we get

$$\sum_{k=0}^{\infty} (2\xi_k^f + \nu_k) + f_{\mathcal{N}}(x^0) - f_{\mathcal{N}}(x^{\bar{k}}) \geq \bar{k} \kappa_c \epsilon^2$$

and the thesis follows. □

Note that in the previous theorem, if $N_{\bar{k}} < N$ then

$$\|\nabla f_{\mathcal{N}}(x^k)\| \leq \epsilon_{\bar{k}} + \xi_k^g. \tag{2.16}$$

Strengthening the assumption on the accuracy in function and gradients we can prove R-linear convergence and complexity bounds that are far better than $\mathcal{O}(\epsilon^{-2})$.

THEOREM 2.8 Assume that Assumption 2.1 holds and let $\{x^k\}$ be generated by Algorithm GIN. If the error sequences $\{\nu_k\}$, $\{\xi_k^f\}$ and $\{\xi_k^g\}$ converge to zero R-linearly then $\{x^k\}$ converges R-linearly to the solution of (1.1).

Proof. Inequalities (2.7), (2.12) and (2.5) and Lemma 2.4 imply

$$\begin{aligned} f_{\mathcal{N}_k}(x^{k+1}) - f_{\mathcal{N}}(x^*) &\leq f_{\mathcal{N}_k}(x^k) - f_{\mathcal{N}}(x^*) + c t_k \nabla f_{\mathcal{N}_k}(x^k)^{\mathrm{T}} s^k + \nu_k \\ &\leq f_{\mathcal{N}_k}(x^k) - f_{\mathcal{N}}(x^*) - c t_k \frac{\lambda_1}{\lambda_n^2} (1 - \eta_k)^2 \|\nabla f_{\mathcal{N}_k}(x^k)\|^2 + \nu_k \\ &\leq f_{\mathcal{N}_k}(x^k) - f_{\mathcal{N}}(x^*) \\ &\quad - c t_k \frac{\lambda_1}{\lambda_n^2} (1 - \eta_k)^2 (\lambda_1 (f_{\mathcal{N}}(x^k) - f_{\mathcal{N}}(x^*)) - \xi_k^g) + \nu_k. \end{aligned}$$

Then, using Lemma 2.5 and (2.12) again, we obtain

$$f_{\mathcal{N}}(x^{k+1}) - f_{\mathcal{N}}(x^*) \leq \rho (f_{\mathcal{N}}(x^k) - f_{\mathcal{N}}(x^*)) + \bar{\xi}_k, \tag{2.17}$$

where $\rho = 1 - c\bar{t} \frac{\lambda_1^2}{\lambda_n^2} (1 - \bar{\eta})^2 \in (0, 1)$ and $\bar{\xi}_k = \nu_k + 2\xi_k^f + \xi_k^g$. Furthermore, we obtain

$$f_{\mathcal{N}}(x^{k+1}) - f_{\mathcal{N}}(x^*) \leq \rho^{k+1} (f_{\mathcal{N}}(x^0) - f_{\mathcal{N}}(x^*)) + \rho \sum_{j=1}^{k} \rho^{j-1} \bar{\xi}_{k-j} + \bar{\xi}_k. \tag{2.18}$$

Thus, Lemma 2.6 yields the statement. □

Notice that the R-linear convergence result obtained in Theorem 2.8 also holds for $\nu_k = 0$, that is, for the Armijo line search. The theorem above also allows us to prove the complexity result below.

THEOREM 2.9 Assume that Assumption 2.1 holds. If the error sequences $\{\nu_k\}$, $\{\xi_k^f\}$ and $\{\xi_k^g\}$ converge to zero R-linearly then for any $\epsilon \in (0, e^{-1})$ there exist $\hat{\rho} \in (0,1)$ and $Q > 0$ such that Algorithm GIN takes at most

$$\bar{k} = \left\lceil \frac{\log((f_{\mathcal{N}}(x^0) - f_{\mathcal{N}}(x^*) + Q))}{|\log(\hat{\rho})|} \right\rceil \log(\epsilon^{-1})$$

iterations to ensure $f_{\mathcal{N}}(x^{\bar{k}}) - f_{\mathcal{N}}(x^*) < \epsilon$.

Proof. The assumptions of Theorem 2.8 are satisfied and

$$f_{\mathcal{N}}(x^k) - f_{\mathcal{N}}(x^*) \leq \rho^k (f_{\mathcal{N}}(x^0) - f_{\mathcal{N}}(x^*)) + \sum_{j=0}^{k-1} \rho^j \bar{\xi}_{k-j}. \tag{2.19}$$

Given that the sequence $\{\sum_{j=0}^{k} \rho^j \bar{\xi}_{k-j}\}$ converges to zero R-linearly by Lemma 2.6, there exist $\bar{\rho} \in (0,1)$ and $Q > 0$ such that

$$\sum_{j=0}^{k-1} \rho^j \bar{\xi}_{k-j} \leq Q \bar{\rho}^k.$$

Therefore, for $\hat{\rho} = \max\{\rho, \bar{\rho}\}$, we have from (2.19),

$$f_{\mathcal{N}}(x^k) - f_{\mathcal{N}}(x^*) \leq \hat{\rho}^k (f_{\mathcal{N}}(x^0) - f_{\mathcal{N}}(x^*) + Q)$$

and the statement follows as in Grapiglia & Sachs (2017, Theorem 6). □

A couple of comments are due here. First of all, Theorems 2.7–2.9 deal with the possibility of an infinite sequence of errors in the objective function and the gradient and thus provide a framework for considering unbounded N as well. However, our focus here is on problems with fixed N. So the statements of the above theorems apply to any kind of scheduling that ensures reaching the full sample for the objective function. Theorem 2.9 proves the complexity bound for $\log(\epsilon^{-1})$, although we work with a cheaper objective function and the gradient whenever $N_k < N$ and thus provides theoretical justification for working with smaller samples. Moreover, the analysis carried out so far does not involve either the forcing term or the accuracy in the Hessian approximation. Then the results we provided hold even if $D_k = 1$ and only one CG iteration is performed at each iteration of Algorithm GIN.

2.2 *Local convergence*

In this section we assume that the scheduling of the sample sizes $\mathcal{N}_k$ is given and that eventually we reach $f_{\mathcal{N}}$ and $\nabla f_{\mathcal{N}}$ at some iteration $\tilde{k}$ and continue with $\mathcal{N}_k = \mathcal{N}$ for $k \geq \tilde{k}$. Then we may restrict all asymptotic theoretical considerations to the method defined with $\mathcal{N}_k = \mathcal{N}$, that is, the step s^k satisfies

$$\nabla^2 f_{\mathcal{D}_k}(x^k) s^k = -\nabla f_{\mathcal{N}}(x^k) + r^k, \quad \|r^k\| \leq \eta_k \|\nabla f_{\mathcal{N}}(x^k)\|. \tag{2.20}$$

Following the analysis in Eisenstat & Walker (1996), in this subsection we focus on the local q-linear and q-superlinear convergence of method GIN. Then we assume that the generated sequence converges to the solution of (1.1), relying on the global convergence results proved in the previous subsection. The analysis presented here differs from that in Eisenstat & Walker (1996) as we have to take into account the approximation in the Hessian and shows why the accuracy in the Hessian's approximation must be related to the adopted forcing term. This theoretical study enables us to devise an adaptive and computable rule for the Hessian sample size yielding q-linear and q-superlinear convergence. In fact, this is what we should obtain in order to justify the use of a second-order method. The analysis is carried out under the following assumption on the subsampled Hessian matrices.

ASSUMPTION 2.10 There exists a constant L such that for any $\mathcal{D} \subseteq \{1, 2, \ldots, N\}$ and any x sufficiently close to x^* we have

$$\|\nabla^2 f_{\mathcal{D}}(x) - \nabla^2 f_{\mathcal{D}}(x^*)\| \leq L\|x - x^*\|. \tag{2.21}$$

The above assumption holds if there exists a neighborhood of x^* where each Hessian $\nabla^2 f_i$ is Lipschitz continuous. Note also that Assumption 2.1 implies that all functions ∇f_i are Lipschitz continuous with the constant λ_n.

Moreover, we assume that the error in the Hessian approximation is determined only by the subsample size, independently of the sample taken, and denote by $h(D, x)$ the norm of the error in the Hessian approximation for a given subsample size D at point x, i.e.,

$$h(D, x) := \|\nabla^2 f_{\mathcal{N}}(x) - \nabla^2 f_{\mathcal{D}}(x)\|.$$

We make the following assumption on the subsample size D_k used at each iteration k.

ASSUMPTION 2.11 At each iteration k of Algorithm GIN the subsample size D_k is chosen such that

$$h(D_k, x^k) \leq C\eta_k, \tag{2.22}$$

with $0 < C < (1/\bar{\eta} - 1)\lambda_1/2$.

In the subsequent analysis, for any $\delta > 0$ the ball with center x^* and radius δ will be denoted by $N_\delta(x^*)$, i.e., $N_\delta(x^*) = \{x \in \mathbb{R}^n : \|x - x^*\| \leq \delta\}$. Moreover, we let δ^* be sufficiently small such that (2.21) holds for any x in $N_{\delta^*}(x^*)$ and $\delta^* < 2\min\{\lambda_1, 1/\lambda_n\}/L$.

The next theorem states that the full step is taken in Algorithm GIN eventually.

THEOREM 2.12 Assume that the sequence $\{x^k\}$ generated by Algorithm GIN converges to x^*. Let Assumptions 2.1 and 2.11 hold and $c \in (0, 1/4)$ in (2.7). Then there exists k_0 such that for all $k \geq k_0$ the full step s^k is accepted in Step 3 of Algorithm GIN.

Proof. Take $\bar{\varepsilon} \in (0, \bar{\eta}\lambda_1/2)$ and $\bar{\delta} > 0$ such that

$$\|\nabla^2 f_{\mathcal{N}}(x^k + \xi s^k) - \nabla^2 f_{\mathcal{N}}(x^k)\| \leq \bar{\varepsilon} \quad \forall \|s^k\| \leq \bar{\delta}, \xi \in (0, 1). \tag{2.23}$$

According to Lemma 2.3, $\lim_{k\to\infty}\|s^k\| = 0$. Then there exists k_0 such that (2.23) holds for $k \geq k_0$. Moreover, the Taylor expansion yields

$$\begin{aligned} f_{\mathcal{N}}(x^k + s^k) &= f_{\mathcal{N}}(x^k) + (\nabla f_{\mathcal{N}}(x^k))^T s^k + \frac{1}{2}(s^k)^T \nabla^2 f_{\mathcal{N}}(\theta^k) s^k \\ &= f_{\mathcal{N}}(x^k) + (\nabla f_{\mathcal{N}}(x^k))^T s^k + \frac{1}{2}(s^k)^T \nabla^2 f_{\mathcal{D}_k}(x^k) s^k \\ &\quad + \frac{1}{2}(s^k)^T (\nabla^2 f_{\mathcal{N}}(\theta^k) \pm \nabla^2 f_{\mathcal{N}}(x^k) - \nabla^2 f_{\mathcal{D}_k}(x^k)) s^k, \end{aligned} \tag{2.24}$$

with $\theta^k = x^k + \xi s^k,\ \xi \in (0,1)$. From Assumption 2.11 we obtain

$$(s^k)^T (\nabla^2 f_{\mathcal{N}}(x^k) - \nabla^2 f_{\mathcal{D}_k}(x^k)) s^k \leq h(D_k, x^k)\|s^k\|^2 \leq C\eta_k \|s^k\|^2 \leq C\bar{\eta}\|s^k\|^2. \tag{2.25}$$

Recall that Lemma 2.2 implies

$$(s^k)^T \nabla^2 f_{\mathcal{D}_k}(x^k) s^k = -\nabla f_{\mathcal{N}}(x^k)^T s^k. \tag{2.26}$$

Putting (2.23), (2.25) and (2.26) into (2.24) we get

$$f_{\mathcal{N}}(x^k + s^k) \leq f_{\mathcal{N}}(x^k) + \frac{1}{2}(\nabla f_{\mathcal{N}}(x^k))^T s^k + \frac{1}{2}(\varepsilon + C\bar{\eta})\|s^k\|^2. \tag{2.27}$$

From (2.9) we conclude that

$$f_{\mathcal{N}}(x^k + s^k) \leq f_{\mathcal{N}}(x^k) + \frac{1}{2}\left(1 - \frac{\varepsilon + C\bar{\eta}}{\lambda_1}\right)(\nabla f_{\mathcal{N}}(x^k))^T s^k.$$

Note that from $C < (1/\bar{\eta} - 1)\lambda_1/2$ there follows $\frac{C\bar{\eta}}{\lambda_1} < (1-\bar{\eta})/2$. Therefore, the choice of ε yields $\varepsilon/\lambda_1 + C\bar{\eta}/\lambda_1 < 1/2$. Then

$$f_{\mathcal{N}}(x^k + s^k) \leq f_{\mathcal{N}}(x^k) + \frac{1}{4}\nabla f_{\mathcal{N}}^T(x^k) s^k$$

and condition (2.7) is satisfied with $t_k = 1$ for any $k > k_0$ as $c \in (0, 1/4)$. □

The following lemma, whose proofs can be found in the appendix, is needed in the subsequent convergence analysis.

LEMMA 2.13 Let Assumptions 2.1–2.10 hold. If $x^k \in \mathcal{N}_{\delta^*}(x^*)$, $s^k \in \mathbb{R}^n$ and $\eta \in (0,1)$ are such that $\|\nabla f_{\mathcal{N}}(x^k) + \nabla^2 f_{\mathcal{D}_k}(x^k) s^k\| \leq \eta \|\nabla f_{\mathcal{N}}(x^k)\|$ and $x^k + s^k \in \mathcal{N}_{\delta^*}(x^*)$ then

$$\|\nabla f_{\mathcal{N}}(x^k + s^k)\| \leq (\eta + B(x^k))\|\nabla f_{\mathcal{N}}(x^k)\|,$$

with $B(x^k) = \frac{1}{\lambda_1}(\frac{1}{2\lambda_1} L\|\nabla f_{\mathcal{N}}(x^k)\| + h(D, x^k))$.

LEMMA 2.14 Let Assumption 2.1 hold and $\delta \in (0, \delta^*/(1+\lambda_1^{-1}\lambda_n))$. If $x^k \in \mathcal{N}_\delta(x^*)$ and $\|\nabla f_{\mathcal{N}}(x^k) + \nabla^2 f_{\mathcal{D}_k}(x^k)s^k\| \le \eta\|\nabla f_{\mathcal{N}}(x^k)\|$ then $x^k + s^k \in N_{\delta^*}(x^*)$.

The convergence of $\{x^k\}$ together with (2.4) implies the following result.

THEOREM 2.15 Assume that the sequence $\{x^k\}$ generated by Algorithm GIN converges to x^*. Let Assumptions 2.1–2.11 hold and $c \in (0, 1/4)$ in (2.7). If $\eta_k = \bar{\eta}$ at each iteration of Algorithm GIN, then the sequence $\{x^k\}$ converges to x^* q-linearly for $\bar{\eta}$ small enough. Moreover, if $\lim_{k\to\infty} \eta_k = 0$, the convergence is q-superlinear.

Proof. Note that, by the choice of C in Assumption 2.11, $\bar{\eta}(1+\lambda_1^{-1}C) < (1+\bar{\eta})/2$. Let $\delta \in (0, \delta^*/(1+\lambda_1^{-1}\lambda_n))$. Take $\bar{\varepsilon} \in (0, \delta\lambda_1]$ sufficiently small such that $\bar{\eta}(1+\lambda_1^{-1}C) + \lambda_1^{-2}L\bar{\varepsilon}/2 < \tau < 1$ for some $\tau \in (0,1)$. Let k_0 be defined as in Theorem 2.12 and $\bar{k} \ge k_0$, such that $x^{\bar{k}} \in \mathcal{N}_\delta(x^*)$ sufficiently near to x^* to guarantee $\|\nabla f_{\mathcal{N}}(x^{\bar{k}})\| \le \bar{\varepsilon}$.

Lemma 2.14 yields $x^{\bar{k}+1} \in \mathcal{N}_{\delta^*}(x^*)$ and by Lemma 2.13 we obtain

$$\|\nabla f_{\mathcal{N}}(x^{\bar{k}+1})\| \le \tau\|\nabla f_{\mathcal{N}}(x^{\bar{k}})\| \le \|\nabla f_{\mathcal{N}}(x^{\bar{k}})\| \le \bar{\varepsilon}.$$

Therefore, using (2.4),

$$\|x^{\bar{k}+1} - x^*\| \le \frac{1}{\lambda_1}\|\nabla f_{\mathcal{N}}(x^{\bar{k}+1})\| \le \frac{1}{\lambda_1}\bar{\varepsilon} \le \delta,$$

so $x^{\bar{k}+1} \in \mathcal{N}_\delta(x^*)$ and $\|\nabla f_{\mathcal{N}}(x^{\bar{k}+1})\| \le \bar{\varepsilon}$.

As an inductive hypothesis suppose that for some $k > \bar{k}$ we have $x^k \in \mathcal{N}_\delta(x^*)$ and $\|\nabla f_{\mathcal{N}}(x^k)\| \le \bar{\varepsilon}$. Then $x^{k+1} = x^k + s^k \in \mathcal{N}_{\delta^*}(x^*)$ by Lemma 2.14, and Lemma 2.13 implies

$$\begin{aligned}\|\nabla f_{\mathcal{N}}(x^{k+1})\| &\le [(1+\lambda_1^{-1}C)\eta_k + \lambda_1^{-2}L\varepsilon/2]\|\nabla f_{\mathcal{N}}(x^k)\| \\ &\le \tau\|\nabla f_{\mathcal{N}}(x^k)\| \le \|\nabla f_{\mathcal{N}}(x^k)\| \le \varepsilon.\end{aligned}$$

Again, (2.4) yields $\|x^{k+1} - x^*\| \le \delta$ and $x^{k+1} \in \mathcal{N}_\delta(x^*)$. Therefore, proceeding by induction we conclude that $x^k \in \mathcal{N}_{\delta^*}(x^*)$ for any $k \ge \bar{k}$ and by Lemma 2.13,

$$\|\nabla f_{\mathcal{N}}(x^{k+1})\| \le [(1+\lambda_1^{-1}C)\eta_k + \lambda_1^{-2}L\|\nabla f_{\mathcal{N}}(x^k)\|/2]\|\nabla f_{\mathcal{N}}(x^k)\|, \quad k \ge \bar{k}. \tag{2.28}$$

Therefore, as $\|\nabla f_{\mathcal{N}}(x^k)\| \to 0$, using (2.4) we obtain that $\{x^k\}$ converges to x^* with a q-linear rate provided that

$$\bar{\eta} < \frac{\lambda_1}{\lambda_n}\frac{1}{1+\lambda_1^{-1}C} < \frac{\lambda_1}{\lambda_n}.$$

Moreover, the q-superlinear convergence follows if $\lim_{k\to\infty} \eta_k = 0$. □

The above results are in line with the classical convergence theory of IN methods (Dembo *et al.*, 1982) as the local linear convergence requires $\eta_k \le \bar{\eta} < 1$ and the upper bound on $\bar{\eta}$ depends on the inverse of the conditioning of the Hessian.

Let us now discuss one possible choice of η_k in order to obtain q-superlinear convergence of the procedure. Following the ideas in Eisenstat & Walker (1996) our choice of η_k depends on the agreement

between the function and the subsampled Newton model. If there is good agreement between these two quantities, even if the quality of the approximation in the Hessian is lower than that in the function, it is reasonable to use a small η in the subsequent iteration. Let us consider the following choice of η_k in Algorithm 2.1:

$$\eta_k = \min\left\{\bar{\eta}, \frac{|f_{\mathcal{N}_k}(x^k) - m_{k-1}(s^{k-1})|}{\|\nabla f_{\mathcal{N}_{k-1}}(x^{k-1})\|}\right\}, \quad \bar{\eta} < 1 \tag{2.29}$$

where

$$m_{k-1}(s) = f_{\mathcal{N}_{k-1}}(x^{k-1}) + \nabla f_{\mathcal{N}_{k-1}}(x^{k-1})^T s + \frac{1}{2} s^T \nabla^2 f_{\mathcal{D}_{k-1}}(x^{k-1}) s. \tag{2.30}$$

In the following theorem we show that the sequence $\{\eta_k\}$ generated by (2.29) converges to zero and ensures q-superlinear convergence provided that the full sample is used eventually.

THEOREM 2.16 Let the assumptions in Theorem 2.15 hold and η_k be given by (2.29). Then $\{x^k\}$ converges to x^* superlinearly.

Proof. Let the iteration index $k > \tilde{k} + 1$, i.e., such that $\mathcal{N}_{k-1} = \mathcal{N}_k = \mathcal{N}$. Using the Taylor expansion and (2.22) we obtain

$$|f_{\mathcal{N}}(x^k) - m_{k-1}(s^{k-1})| \leq \frac{1}{2}\|s^{k-1}\|^2 \left(\frac{L}{2}\|s^{k-1}\| + C\eta_{k-1}\right). \tag{2.31}$$

Now, by Lemma 2.3 there follows

$$\eta_k \leq \frac{1}{2}\lambda_1^{-2}\|\nabla f_{\mathcal{N}}(x^{k-1})\|^2 \left[\lambda_1^{-1}\frac{L}{2}\|\nabla f_{\mathcal{N}}(x^{k-1})\| + C\bar{\eta}\right]. \tag{2.32}$$

Then as $\lim_{k\to\infty} \|\nabla f_{\mathcal{N}}(x^k)\| = 0$ we have $\lim_{k\to\infty} \eta_k = 0$ and the superlinear convergence follows by Theorem 2.15. □

The above result can be proved also by choosing η_k as

$$\eta_k = \min\left\{\bar{\eta}, \frac{|f_{\mathcal{N}_k}(x^k) - m_{k-1}(s^{k-1})|}{\omega_k}\right\}, \quad \bar{\eta} < 1 \tag{2.33}$$

with

$$\omega_k = \mathcal{O}(\|\nabla f_{\mathcal{N}_{k-1}}(x^{k-1})\|).$$

We also underline that Theorem 2.16 in the case of full Hessian, i.e., $\mathcal{N}_k = \mathcal{D}_k$, shows that the IN method with the choice of forcing terms given by (2.33) is superlinearly convergent. Such a result also follows from the analysis in Lee *et al.* (2014, Theorem 3.10).

3. Mean square convergence

A large part of the previous analysis strongly relies on the Hessian error bound $h(D, x)$, which is not easily accessible. In this section we consider randomly chosen D_k and we ask for a good enough Hessian approximation with some probability smaller than 1. Thus, less conservative estimates are feasible. We

use a bound similar to that derived in Xu *et al.* (2018) to carry out the analysis and we obtain stochastic convergence results—convergence in an m.s. The main result is that the q-linear convergence in an m.s. can be achieved with a small enough but fixed forcing term η and with large enough but fixed Hessian sample size D. On the other hand, to achieve q-superlinear convergence in an m.s. with η_k defined by (2.33), the Hessian sample size is required to increase as η_k goes to zero. This analysis paves the way to devise adaptive rules for selecting the Hessian sample size such that a small Hessian sample size is used in the early stage of the procedure, when η_k is close to 1 and linear systems are solved only to a low accuracy, and it is automatically increased when the solution is approached.

In this section we assume that the subsample $\mathcal{D}$ is chosen randomly and uniformly—every $\nabla^2 f_i(x)$ has the same chance to be chosen. Let $\mathcal{D}$ be any subset of $\mathcal{N}$ such that $|\mathcal{D}| = D$. Then one can derive a bound on D such that, given $\gamma > 0$ and $\alpha \in (0, 1)$,

$$P(\|\nabla^2 f_{\mathcal{D}}(x) - \nabla^2 f_{\mathcal{N}}(x)\| \leq \gamma) \geq 1 - \alpha. \tag{3.1}$$

The corresponding bound is stated in Lemma 3.1 (see the proof in the appendix), while a similar bound is provided in Xu *et al.* (2018, Lemma 4). The result is obtained by using the Bernstein inequality; see Tropp (2012) and Minsker (2017) for further references.

LEMMA 3.1 Assume that 2.1 holds and that the subsample $\mathcal{D}$ is chosen randomly and uniformly from $\mathcal{N}$. Let $\gamma > 0$ and $\alpha \in (0, 1)$ be given. Then (3.1) holds at any point x if the subsample size D satisfies

$$D \geq \frac{2(\ln(2n/\alpha))(\lambda_n^2 + \lambda_n \gamma/3)}{\gamma^2} := \tilde{l}. \tag{3.2}$$

We use the above results and the analysis of the previous section, to design a globally convergent inexact subsampled Newton method with adaptive choice of the Hessian sample size. In particular, we will choose D_k such that the inequality

$$h(D_k, x^k) \leq C \max\{\eta_k, \|\nabla f_{\mathcal{N}_k}(x^k)\|\} \tag{3.3}$$

holds with probability $1 - \alpha_k$, with $\alpha_k \in (0, 1)$ and $0 < C < (1/\bar{\eta} - 1)\lambda_1/2$. This corresponds to $\gamma = C \max\{\eta_k, \|\nabla f_{\mathcal{N}_k}(x^k)\|\}$ in (3.1).

3.1 *Bounded sample—GIN-R method*

In this subsection we are interested in the case of finite N. Let us assume that the full sample is eventually reached for the objective function and the gradient, i.e., $\mathcal{N}_k = \mathcal{N}$, for k sufficiently large, say for all $k \geq \bar{k}$. The procedure we obtain is based on the GIN method but with a specific choice of the Hessian subsample $\mathcal{D}_k$; namely its cardinality is set according to (3.3) and the sample is randomly chosen. This procedure is denoted GIN-R to emphasize the specific random choice of the Hessian subsample. We list its generic iteration k in Algorithm 3.1, where we denote the first steps as Step 1.a–1.c to make clear that they correspond to specific choices in Step 1 of Algorithm GIN.

We will analyze the convergence in the m.s. considering two possibilities. If the forcing terms converge to zero, then γ_k given by (3.4) converges to zero. The other case we consider is $\eta_k = \bar{\eta} < 1$. In this latter case Algorithm GIN-R yields γ_k bounded away from zero and we have $\mathcal{D}_k \subset \mathcal{N}$ with $D_k < N$ during the whole iterative process. We remark that Theorem 2.8 implies the R-linear convergence of the sequence generated by Algorithm GIN for any Hessian subsampling under the

condition $\mathcal{N}_k = \mathcal{N}, k \geq \bar{k}$. The key issue in the global convergence analysis is that s^k is a descent search direction with an arbitrary good or poor Hessian approximation, i.e., regardless of the subsample used in the GIN algorithm. The line-search globalization strategy makes the algorithm R-linearly convergent for any choice of the subsample and the following statement holds.

Algorithm 3.1 kth iteration of method GIN-R

Given: $x^k \in \mathbb{R}^n,\ \bar{\eta} \in (0,1),\ c \in (0,1),\ C > 0,\ \nu_k,\ \alpha_k \in (0,1)$.

Step 1.a Choose $\mathcal{N}_k, \eta_k \in (0, \bar{\eta})$.

Step 1.b Compute

$$\gamma_k = C \max\{\eta_k, \|\nabla f_{\mathcal{N}_k}(x^k)\|\}. \tag{3.4}$$

Step 1.c Set $\alpha = \alpha_k$ and $\gamma = \gamma_k$ and compute D such that (3.2) holds. If $D \geq N_k$ set $\mathcal{D}_k = \mathcal{N}_k$. Else, choose the sample $\mathcal{D}_k$ randomly and uniformly from $\mathcal{N}_k$ such that $D_k \geq D$.

Step 2. Apply the CG method initialized by the null vector to $\nabla^2 f_{\mathcal{D}_k}(x^k)s^k = -\nabla f_{\mathcal{N}_k}(x^k)$ and compute s^k satisfying (2.2).

Step 3. Find the smallest non-negative integer j such that (2.7) holds for $t_k = 2^{-j}$ and set $x^{k+1} = x^k + t_k s^k$.

THEOREM 3.2 Assume that 2.1 holds and let $\{x^k\}$ be generated by algorithm GIN-R. If $\{\nu_k\}$ converges to zero R-linearly then $\{x^k\}$ converges R-linearly to x^*.

Next we show another important intermediate result.

LEMMA 3.3 Suppose that the assumptions of Theorem 3.2 are satisfied and let $\{x^k\}$ be a sequence generated by algorithm GIN-R. If η_k is defined by (2.29) then there exist positive constants B_1 and B_2 and $\tau \in (0,1)$ such that

$$\eta_k \leq B_1 \tau^k \quad \text{and} \quad \gamma_k \leq B_2 \tau^k$$

for all k large enough. If $\eta_k = \bar{\eta}$ then $\gamma_k = C\bar{\eta}$ for all k large enough.

Proof. First, Theorem 3.2 implies that the sequence of iterates x^k converges to the unique solutions R-linearly; for all k large enough we have

$$\|x^k - x^*\| \leq B\tau^k \tag{3.5}$$

where $B > 0$ and $\tau \in (0,1)$. Now using the Taylor expansion, Lemma 2.3 and (2.3) we obtain for some $\theta_k \in [x^{k-1}, x^k]$,

$$\begin{aligned}
&|f_{\mathcal{N}}(x^k) - m_{k-1}(s^{k-1})| \\
&\quad = \big| f_{\mathcal{N}}(x^{k-1}) + t_k \nabla f_{\mathcal{N}}(x^{k-1})^T s^{k-1} + \frac{1}{2} t_k^2 (s^{k-1})^T \nabla^2 f_{\mathcal{N}}(\theta_k) s^{k-1} \\
&\qquad - f_{\mathcal{N}}(x^{k-1}) - \nabla f_{\mathcal{N}}(x^{k-1})^T s^{k-1} - \frac{1}{2} (s^{k-1})^T \nabla^2 f_{\mathcal{D}_{k-1}}(x^{k-1}) s^{k-1} \big| \\
&\quad \le \frac{1 - t_k}{\lambda_1} \|\nabla f_{\mathcal{N}}(x^{k-1})\|^2 + \lambda_n \|s^{k-1}\|^2 \le \frac{1}{\lambda_1} \|\nabla f_{\mathcal{N}}(x^{k-1})\|^2 + \frac{\lambda_n}{\lambda_1^2} \|\nabla f_{\mathcal{N}}(x^{k-1})\|^2.
\end{aligned}$$

Therefore, using (2.4), for η_k given by (2.29) we obtain

$$\eta_k \le \frac{|f_N(x^k) - m_{k-1}(s^{k-1})|}{\|\nabla f_N(x^{k-1})\|} \le \left(\frac{1}{\lambda_1} + \frac{\lambda_n}{\lambda_1^2} \right) \lambda_n \|x^{k-1} - x^*\| \le B_1 \tau^k \tag{3.6}$$

where $B_1 = (\frac{1}{\lambda_1} + \frac{\lambda_n}{\lambda_1^2}) \lambda_n B/\tau$. Moreover, for γ_k given by (3.4) we get

$$\gamma_k \le C \max\{B_1 \tau^k, \lambda_n B \tau^k\} = C \max\{B_1, \lambda_n B\} \tau^k := B_2 \tau^k.$$

Considering the case with $\eta_k = \bar{\eta}$, since the gradient converges to zero, for all k large enough we have $\|\nabla f_{\mathcal{N}}(x^k)\| < \bar{\eta}$ and therefore $\gamma_k = C\bar{\eta}$. □

Now we are ready to show the main result of this subsection.

THEOREM 3.4 Suppose that the assumptions of Theorem 3.2 and Assumption 2.10 are satisfied. Moreover, assume $c \in (0, 1/4)$ in (2.7) and $0 < C < (1/\bar{\eta} - 1)\lambda_1/2$ in (3.3). Let $\{x^k\}$ be a sequence generated by Algorithm GIN-R. Then there are positive constants V_1, V_2, C_1, C_2 and $\tau \in (0,1)$ such that for all k sufficiently large

(a) if η_k is defined by (2.29) then

$$E(\|x^{k+1} - x^*\|^2) \le \left(V_1 \tau^{2k} + V_2 \alpha_k \right) E(\|x^k - x^*\|^2);$$

(b) if $\eta_k = \bar{\eta}$ is sufficiently small then

$$E(\|x^{k+1} - x^*\|^2) \le \left((C_1 \tau^k + C_2 \bar{\eta})^2 + V_2 \alpha_k \right) E(\|x^k - x^*\|^2).$$

Proof. Since the assumptions of Theorem 3.2 are satisfied, the sequence $\{x^k\}$ converges to the solution R-linearly, independently of D_k and for $\eta_k \in (0,1)$. Thus, there exist constants $B > 0$ and $\tau \in (0,1)$ such that $\|x^k - x^*\| \le B\tau^k$. Moreover, $N_k = N$ for all k sufficiently large. So, without loss of generality, we assume that the full sample is used for the gradient and the function and $\|\nabla f_{\mathcal{N}}(x^k)\| < \bar{\eta}$.

Employing (2.5) and Lemma 2.3, we have the following estimate:

$$\begin{aligned}\|x^k + t_k s^k - x^*\|^2 &\leq \frac{2}{\lambda_1}(f_{\mathcal{N}}(x^k + t_k s^k) - f_{\mathcal{N}}(x^*)) \\ &\leq \frac{2}{\lambda_1}\Big(f_{\mathcal{N}}(x^k) - f_{\mathcal{N}}(x^*) + t_k(\nabla f_{\mathcal{N}}(x^k))^T s^k + \frac{\lambda_n}{2}\|t_k s^k\|^2\Big) \\ &\leq \frac{2}{\lambda_1}\Big(\frac{1}{\lambda_1}\|\nabla f_{\mathcal{N}}(x^k)\|^2 + \frac{\lambda_n}{2}\frac{1}{\lambda_1^2}\|\nabla f_{\mathcal{N}_k}(x^k)\|^2\Big) \\ &\leq \frac{2}{\lambda_1}\Big(\frac{1}{\lambda_1} + \frac{\lambda_n}{2\lambda_1^2}\Big)\lambda_n^2\|x^k - x^*\|^2 := V_2\|x^k - x^*\|^2. \end{aligned} \tag{3.7}$$

Let us denote by A_k the event $\|\nabla^2 f_{\mathcal{D}_k}(x^k) - \nabla^2 f_{\mathcal{N}}(x^k)\| \leq \gamma_k$. Due to Step 1.c of Algorithm GIN-R it follows that $P(A_k) \geq 1 - \alpha_k$, i.e., $P(\bar{A}_k) \leq \alpha_k$. Notice that (3.7) holds in both cases but in the case of A_k we can derive a better estimate.

Assume that A_k happens. Then

$$\begin{aligned}&\|\nabla^2 f_{\mathcal{N}}(x^k)s^k + \nabla f_{\mathcal{N}}(x^k)\| \\ &\quad\leq \|(\nabla^2 f_{\mathcal{N}}(x^k) - \nabla^2 f_{\mathcal{D}_k}(x^k))s^k\| + \|\nabla^2 f_{\mathcal{D}_k}(x^k)s^k + \nabla f_{\mathcal{N}}(x^k)\| \\ &\quad\leq \gamma_k\|s^k\| + \eta_k\|\nabla f_{\mathcal{N}}(x^k)\| \leq (\gamma_k/\lambda_1 + \eta_k)\|\nabla f_{\mathcal{N}}(x^k)\|.\end{aligned}$$

Moreover, as $\gamma_k < C\bar{\eta}$ we can repeat the reasoning used in the proof of Theorem 2.12 to conclude that the full step is accepted for k sufficiently large. As the standard assumptions for the (inexact) Newton method are satisfied one can prove

$$\|x^k + s^k - x^*\| \leq c_1(\|x^k - x^*\| + \gamma_k/\lambda_1 + \eta_k)\|x^k - x^*\| \tag{3.8}$$

for some positive constant c_1 and for k sufficiently large. Indeed, denoting $\tilde{r}_k = \nabla^2 f_{\mathcal{N}}(x^k)s^k + \nabla f_{\mathcal{N}}(x^k)$, it holds $\|\tilde{r}_k\| \leq (\gamma_k/\lambda_1 + \eta_k)\|\nabla f_{\mathcal{N}}(x^k)\|$ and

$$\|x^k + s^k - x^*\| = \|x^k + (\nabla^2 f_{\mathcal{N}}(x^k))^{-1}\tilde{r}_k - (\nabla^2 f_{\mathcal{N}}(x^k))^{-1}\nabla f_{\mathcal{N}}(x^k) - x^*\|.$$

Since Newton's method converges quadratically there exists $\kappa > 0$ such that

$$\|x^k - (\nabla^2 f_{\mathcal{N}}(x^k))^{-1}\nabla f_{\mathcal{N}}(x^k) - x^*\| \leq \kappa\|x^k - x^*\|^2.$$

Now using $\|(\nabla^2 f_{\mathcal{N}}(x^k))^{-1}\| \leq 1/\lambda_1$, $\|\nabla f_{\mathcal{N}}(x^k)\| \leq \lambda_n\|x^k - x^*\|$ and defining $c_1 = \max\{\kappa, \lambda_n/\lambda_1\}$ we obtain (3.8). Using inequality (3.5) and squaring inequality (3.7) we obtain

$$\|x^k + s^k - x^*\|^2 \leq 2c_1^2(B^2\tau^{2k} + (\gamma_k/\lambda_1 + \eta_k)^2)\|x^k - x^*\|^2. \tag{3.9}$$

Now we distinguish two cases depending on η_k. Using the result of Lemma 3.3 and assuming that k is sufficiently large we obtain the following:

(a) If η_k is defined by (2.29), for $V_1 = 2c_1^2(B^2 + 2B_1^2 + 2(B_2/\lambda_1)^2)$ we get

$$\begin{aligned} &E(\|x^{k+1} - x^*\|^2) \\ &\quad = P(A_k)E(\|x^{k+1} - x^*\|^2|A_k) + P(\bar{A}_k)E(\|x^{k+1} - x^*\|^2|\bar{A}_k) \\ &\quad \leq (V_1\tau^{2k} + \alpha_k V_2)E(\|x^k - x^*\|^2). \end{aligned} \tag{3.10}$$

(b) Considering $\eta_k = \bar{\eta}$, for $C_1 = 2c_1^2B^2$ and $C_2 = 2c_1^2(C/\lambda_1 + 1)^2$ we get

$$E(\|x^{k+1} - x^*\|^2) \leq \left(C_1\tau^{2k} + C_2\bar{\eta}^2 + V_2\alpha_k\right) E(\|x^k - x^*\|^2). \tag{3.11}$$

□

We conclude the analysis with the following corollary.

COROLLARY 3.5 Assume that the conditions of Theorem 3.4 hold and let $\{x^k\}$ be a sequence generated with Algorithm GIN-R. Then the sequence $\{x^k\}$ converges to x^* in the m.s.,

(a) linearly if $\eta_k = \bar{\eta}$ is sufficiently small and $\alpha_k < \frac{1-C_2^2\bar{\eta}^2}{V_2}$;

(b) superlinearly if η_k is defined by (2.29) and $\lim_{k\to\infty} \alpha_k = 0$.

Note that the requirement on the probability α_k in order to get q-linear convergence influences only the logarithmic factor in (3.2).

3.2 *Unbounded sample—GIN method*

In many applications, the number of training points is enlarged over time so the cardinality of the sample set N is actually unbounded. This motivated us to consider the following problem as well:

$$\min_{x\in\mathbb{R}^n} f(x) = E(F(x, \xi)), \tag{3.12}$$

where ξ is a random variable defined on a probability space $(\mathcal{A}, \mathcal{F}, P)$ and F is a twice-differentiable function with respect to x. Let us denote $f_i(x) := F(x, \xi_i)$ where ξ_i, $i = 1, 2, \ldots$ is an i.i.d. sequence of variables following the same distribution as ξ. For example, ξ_i can represent the pair of input–output variables in machine-learning problems. Then we can use the same notation as in the previous sections to define the SAA approximation of the objective function and its derivatives: $f_{\mathcal{N}}$, $\nabla f_{\mathcal{N}}$, $\nabla^2 f_{\mathcal{D}}$. We will prove that, under appropriate assumptions, GIN converges in an m.s. towards the solution of problem (3.12).

ASSUMPTION 3.6 The sequence $\{\xi_i\}$, $i = 1, 2, \ldots$ is an i.i.d. sequence of variables.

Next we assume that the sequence of iterates $\{x^k\}$ belongs to a bounded set. This assumption is stronger than the assumption of bounded moments of iterates used in Bollapragada *et al.* (2018). However, Bollapragada *et al.* (2018) employ a fixed step length, assuming knowledge of the maximum eigenvalue λ_n, while here we employ a line search with approximate function and derivative values and the assumption is needed to cope with such inexactness in the line search. The analysis of properties of $F(x, \xi)$ that guarantee this assumption is beyond the scope of this paper.

ASSUMPTION 3.7 There exists a compact set Ω such that $\{x^k\}_{k\in\mathbb{N}} \subseteq \Omega$.

ASSUMPTION 3.8 Let $F(\cdot,\xi) \in C^2(\mathbb{R}^n)$ for every ξ. Let F and ∇F be dominated by integrable functions $M_f(\xi)$ and $M_g(\xi)$, respectively, on an open set containing Ω.

Assumption 3.6 implies that $E(f_{\mathcal{N}}(x)) = f(x)$. Moreover, 3.6 and 3.8 imply that $\nabla f(x) = E(\nabla F(x,\xi))$ and therefore $E(\nabla f_{\mathcal{N}}(x)) = \nabla f(x)$. Furthermore, the uniform law of large numbers implies that $f_{\mathcal{N}}$ and $\nabla f_{\mathcal{N}}$ almost surely (a.s.) converge to $f(x)$ and $\nabla f(x)$, respectively, uniformly on Ω when N tends to infinity. Denote

$$e_{\mathcal{N}} = \max_{x\in\Omega} |f_{\mathcal{N}}(x) - f(x)|, \qquad \tilde{e}_{\mathcal{N}} = \max_{x\in\Omega} \|\nabla f_{\mathcal{N}}(x) - \nabla f(x)\|. \tag{3.13}$$

Then $\lim_{N\to\infty} e_{\mathcal{N}} = 0$ and $\lim_{N\to\infty} \tilde{e}_{\mathcal{N}} = 0$ a.s. and using the Lebesgue dominated convergence theorem (see of Shapiro *et al.*, 2009, Theorem 7.31) we obtain

$$\lim_{N\to\infty} E(e_{\mathcal{N}}) = 0, \quad \lim_{N\to\infty} E(\tilde{e}_{\mathcal{N}}) = 0. \tag{3.14}$$

Assuming the strong convexity of f_i as in Assumption 2.1, it is easy to show that $f_{\mathcal{N}}$ is also strongly convex with the same constants λ_1 and λ_n for any $\mathcal{N}$. Moreover, assuming 3.8 , f also remains strongly convex with the constant λ_1. Indeed, for an arbitrary i and x, y there holds

$$f_i(y) \geq f_i(x) + \nabla^T f_i(x)(y-x) + \frac{\lambda_1}{2}\|x-y\|^2.$$

Taking the expectation and using that $E(\nabla f_i(x)) = \nabla f(x)$ we obtain the strong convexity of f. Therefore, problem (3.12) has a unique solution x^*.

THEOREM 3.9 Suppose that Assumptions 2.1 and 3.6– 3.8 hold and that $N_k \to \infty$. Then any sequence $\{x^k\}_{k\in\mathbb{N}}$ generated by GIN converges towards the solution of problem (3.12) in the m.s.

Proof. First notice that Assumptions 3.6– 3.8 imply that $|f_{\mathcal{N}}(x^k) - f(x^k)| \leq e_{\mathcal{N}}$ for every k, so following the reasoning as in the proof of Theorem 2.8 we obtain

$$f(x^{k+1}) - f(x^*) \leq f_{\mathcal{N}_k}(x^k) - c\bar{t}q\|\nabla f_{\mathcal{N}_k}(x^k)\|^2 + e_{\mathcal{N}_k} - f(x^*) + \nu_k, \tag{3.15}$$

where $\bar{t} = (1-c)\lambda_1/\lambda_n$ and $q = (\lambda_1(1-\bar{\eta})^2)/(\lambda_n^2)$. Now

$$\begin{aligned}
\|\nabla f_{\mathcal{N}_k}(x^k)\|^2 &= \|\nabla f_{\mathcal{N}_k}(x^k) - \nabla f(x^k) + \nabla f(x^k)\|^2 \\
&= \|\nabla f(x^k)\|^2 + 2(\nabla f(x^k))^T(\nabla f_{\mathcal{N}_k}(x^k) - \nabla f(x^k)) \\
&\quad + \|\nabla f_{\mathcal{N}_k}(x^k) - \nabla f(x^k)\|^2 \\
&\geq \|\nabla f(x^k)\|^2 - 2\|\nabla f(x^k)\|\|\nabla f_{\mathcal{N}_k}(x^k) - \nabla f(x^k)\| \\
&\geq \|\nabla f(x^k)\|^2 - 2M_g\tilde{e}_{\mathcal{N}_k},
\end{aligned} \tag{3.16}$$

where the last inequality follows from (3.13), continuity of ∇f, Assumption 3.7 and $M_g = \max_{x\in\Omega}\|\nabla f(x)\|$. On the other hand, strong convexity of f implies that $-\|\nabla f(x^k)\|^2 \leq -\lambda_1(f(x^k) -$

$f(x^*)$). Putting all together into (3.15) we obtain

$$f(x^{k+1}) - f(x^*) \leq (f(x^k) - f(x^*))(1-\omega) + 2e_{\mathcal{N}_k} + 2c\bar{t}qM_g\tilde{e}_{\mathcal{N}_k} + \nu_k,$$

where $\omega = c\bar{t}q\lambda_1 \in (0,1)$. Applying expectation we get

$$E(f(x^{k+1}) - f(x^*)) \leq E(f(x^k) - f(x^*))(1-\omega) + a_k,$$

where $a_k = 2E(e_{\mathcal{N}_k}) + 2c\bar{t}qM_gE(\tilde{e}_{\mathcal{N}_k}) + \nu_k$. Now (3.14), (2.6) and the assumption that $N_k \to \infty$ together imply that $\lim_{k\to\infty} a_k = 0$. Therefore, it follows (Jakovetic *et al.*, 2014) that

$$\lim_{k\to\infty} E(f(x^k) - f(x^*)) = 0.$$

Finally, strong convexity implies $\|x^k - x^*\|^2 \leq (f(x^k) - f(x^*))2/\lambda_1$; thus,

$$\lim_{k\to\infty} E(\|x^k - x^*\|^2) = 0.$$

□

3.3 *Relaxing the strong convexity—method GIN-M*

Let us now consider a relaxation of the strong convexity assumption (2.3) by letting $\nabla^2 f_i(x)$ be only positive semidefinite while the final objective function remains strongly convex. Similar assumptions are stated in Roosta-Khorasani & Mahoney (2019). Notice that Theorem 3.9 does not impose any assumption on the size of the Hessian subsample. On the contrary, when the strong convexity assumption is relaxed, a sufficiently large sample is needed to ensure a positive definite Hessian with some probability fixed beforehand. We use the bound on the Hessian sample size provided in Roosta-Khorasani & Mahoney (2019) but using a different approach. First we prove mean square convergence to the solution of (3.12) under appropriate conditions given in the sequel, while in Roosta-Khorasani & Mahoney (2019) convergence with some (high) probability for finite sum problems like (1.1) is considered. Second, we continue to use CG as the inner solver—although modified in this case to cope with a possibly singular matrix.

Assumption 3.10 The functions f and $F(\cdot,\xi)$ are twice continuously differentiable and there exist $0 < \lambda_1 < 1$ and $\lambda_n > 0$ such that for every x, ξ,

$$0 \preceq \nabla_x^2 F(x,\xi) \preceq \lambda_n I \quad \text{and} \quad \lambda_1 I \preceq \nabla^2 f(x) \preceq \lambda_n I.$$

This assumption ensures that the unique solution of the original problem still exists. Moreover, we assume that the Hessian approximations are unbiased.

Assumption 3.11 For every $x \in \mathbb{R}^n$ and every $i \in \mathbb{N}$ there holds $E(\nabla^2 f_i(x)) = \nabla^2 f(x)$.

This assumption allows us to use the matrix Chernoff result (Tropp, 2012) and to obtain the bound presented in Roosta-Khorasani & Mahoney (2019, Lemma 1). Although we observe unbounded sample the same result holds. More precisely, under Assumptions 3.10 and 3.11 we obtain that given

$\mu \in (0, 1-\lambda_1)$ the following holds:

$$P\left(\lambda_{\min}(\nabla^2 f_{\mathcal{D}}(x)) \geq \mu\right) \geq 1-\alpha \quad \text{if } D \geq \frac{2\lambda_n(1-\lambda_1)^2 \ln(n/\alpha)}{\mu^2 \lambda_1} := \bar{D}(\alpha). \tag{3.17}$$

Given that the subsampled Hessian $\nabla^2 f_{\mathcal{D}_k}$ might be singular, for the computation of the step s^k at Step 2 of Algorithm GIN-R, we proceed as follows. If at iteration j of CG we have $(s_j^k)^T \nabla^2 f_{\mathcal{D}_k}(x^k) s_j^k = 0$, CG is stopped and $s^k = s_{j-1}^k$ is set. Notice that s^k is still a descent direction as $(s_{j-1}^k)^T \nabla f(x^k) = -(s_{j-1}^k)^T \nabla^2 f_{\mathcal{D}_k}(x^k) s_{j-1}^k < 0$. The algorithm we use is again GIN-R where the sample size is selected such that the subsampled Hessian is positive definite with probability $1-\alpha$. The modified CG, as explained above, is used. We refer to the obtained procedure as Algorithm GINR-M and its iteration k is detailed in Algorithm 3.2.

Algorithm 3.2 kth iteration of method GINR-M

Given: $x^k \in \mathbb{R}^n$, $c \in (0,1)$, $\bar{\eta} \in (0,1)$, $C > 0$, $\{\nu_k\}$, $\alpha \in (0,1)$ $\mu \in (0, 1-\lambda_1)$.

Step 1.a Choose $\mathcal{N}_k$, $\eta_k \in (0, \bar{\eta})$.

Step 1.b If $N_k \leq \bar{D}(\alpha)$ given in (3.17), set $\mathcal{D}_k = \mathcal{N}_k$. Else, choose $D_k \geq \bar{D}(\alpha)$ and the subsample $\mathcal{D}_k$ randomly and uniformly from $\mathcal{N}_k$.

Step 2. Determine s^k with modified CG: if $(s_j^k)^T \nabla^2 f_{\mathcal{D}_k}(x^k) s_j^k = 0$ for some inner iteration j, set $s^k = s_{j-1}^k$. Otherwise, find the step s^k such that (2.2) holds.

Step 3. Find the smallest non-negative integer j such that (2.7) holds for $t_k = 2^{-j}$ and set $x^{k+1} = x^k + t_k s^k$.

Relaxing the strong convexity results in loss of the usual relation between the step and the gradient stated in Lemma 2.3. Therefore, we need the following assumption.

ASSUMPTION 3.12 There exists a constant $M_s > 0$ such that the step generated by GINR-M satisfies $\|s^k\| \leq M_s$ for every k.

A comment is due with respect to the above assumption. Assume that the step s^k computed at Step 4 of GINR-M has been generated at iteration j of CG. Then it belongs to the Krylov subspace:

$$\mathcal{K}_j = \text{span}\{\nabla f_{\mathcal{N}_k}(x^k), (\nabla^2 f_{\mathcal{D}_k}(x^k))\nabla f_{\mathcal{N}_k}(x^k), \ldots, (\nabla^2 f_{\mathcal{D}_k}(x^k))^{j-1}\nabla f_{\mathcal{N}_k}(x^k)\}.$$

Let V be an orthonormal basis of $\mathcal{K}_j$; then $s^k = Vy$, where $y \in \mathbb{R}^j$. Therefore,

$$(s^k)^T \nabla^2 f_{\mathcal{D}_k}(x^k) s^k = y^T V^T \nabla^2 f_{\mathcal{D}_k}(x^k) V y \geq \lambda_{\min}(V^T \nabla^2 f_{\mathcal{D}_k}(x^k) V)\|y\|^2.$$

Therefore, noting that $\|s\| = \|y\|$ as V is orthornormal, from

$$(s^k)^T \nabla^2 f_{\mathcal{D}_k}(x^k) s^k = -(s^k)^T \nabla f_{\mathcal{N}_k}(x^k)$$

and the Cauchy–Schwartz inequality, there follows

$$\|s^k\| \leq \frac{1}{\lambda_{\min}(V^T\nabla^2 f_{\mathcal{D}_k}(x^k)V)}\|\nabla f_{\mathcal{N}_k}(x^k)\|.$$

Then Assumption 3.12 is satisfied if the minimal eigenvalue of the projected subsampled Hessian $V^T\nabla^2 f_{\mathcal{D}_k}(x^k)V$ is bounded away from zero and Assumption 3.7 holds.

THEOREM 3.13 Suppose that Assumptions 3.10– 3.12 3.6, and 3.8 hold and that N_k tends to infinity. Then there exists α small enough such that any sequence $\{x^k\}_{k\in\mathbb{N}}$ generated by GINR-M converges towards the solution of (3.12) in an m.s.

Proof. Let us denote by A_k the event $\lambda_{\min}(\nabla^2 f_{\mathcal{D}_k}(x^k)) \geq \mu$ and note that since N_k tends to infinity, $N_k \geq \bar{D}(\alpha)$ will be satisfied for all k large enough ($k \geq k(\alpha)$). Since we are interested in an asymptotic result, without loss of generality we assume that $k \geq k(\alpha)$. Then in Step 1.b, D_k is chosen such that (3.17) holds and this implies $P(\bar{A}_k) \leq \alpha$.

Assume that A_k happens. Then we can proceed as in the proof of Theorem 3.9 as f is strongly convex and $\lambda_{\min}(\nabla^2 f_{\mathcal{D}}(x^k)) \geq \mu$ yields $(\nabla f_{\mathcal{N}_k}(x^k))^T s^k \leq -\mu\|s^k\|^2$. We obtain

$$f(x^{k+1}) - f(x^*) \leq (f(x^k) - f(x^*))(1-\omega) + 2e_{\mathcal{N}_k} + \theta\tilde{e}_{\mathcal{N}_k} + \nu_k,$$

where $\omega = c\bar{\iota}q\lambda_1 \in (0,1)$, $\bar{\iota} = (1-c)\mu/\lambda_n$, $q = \mu(\frac{1-\bar{\eta}}{\lambda_n})^2$, $\theta = 2c\bar{\iota}qM_g$ and $M_g = \max_{x\in\Omega}\|\nabla f(x)\|$.

On the other hand, assume that $\bar{A}_k$ happens. Then by the Taylor expansion and Assumption 3.10 we obtain

$$f(x^{k+1}) - f(x^*) \leq f(x^k) - f(x^*) + t_k(\nabla f(x^k))^T s^k + \frac{1}{2}\lambda_n\|t_k s^k\|^2. \tag{3.18}$$

Again, using the strong convexity of f we get

$$\begin{aligned}\|t_k s^k\|^2 &= \|x^{k+1} - x^k\|^2 \leq 2(\|x^{k+1} - x^*\|^2 + \|x^k - x^*\|^2)\\ &\leq \frac{4}{\lambda_1}(f(x^{k+1}) - f(x^*) + f(x^k) - f(x^*)).\end{aligned} \tag{3.19}$$

Moreover,

$$\begin{aligned}(s^k)^T\nabla f(x^k) &= (s^k)^T(\nabla f(x^k) \pm \nabla f_{\mathcal{N}_k}(x^k))\\ &\leq \|s^k\|\|\nabla f(x^k) - \nabla f_{\mathcal{N}_k}(x^k)\| + (\nabla f_{\mathcal{N}_k}(x^k))^T s^k\\ &= \|s^k\|\|\nabla f(x^k) - \nabla f_{\mathcal{N}_k}(x^k)\| - (s^k)^T\nabla^2 f_{\mathcal{D}_k}(x^k)s^k\\ &\leq M_s\tilde{e}_{\mathcal{N}_k},\end{aligned} \tag{3.20}$$

where the last inequality follows from the fact that $\nabla^2 f_{\mathcal{D}_k}(x^k)$ is positive semidefinite. Putting (3.20) into (3.18) together with (3.19) we obtain

$$f(x^{k+1}) - f(x^*) \leq (f(x^k) - f(x^*))(1 + 2\lambda_n/\lambda_1) + 2\lambda_n/\lambda_1(f(x^{k+1}) - f(x^*)) + M_s\tilde{e}_{\mathcal{N}_k}.$$

Combining all together we get

$$\begin{aligned}
&E(f(x^{k+1}) - f(x^*)) \\
&\quad = P(A_k)E(f(x^{k+1}) - f(x^*)|A_k) + P(\bar{A}_k)E(f(x^{k+1}) - f(x^*)|\bar{A}_k) \\
&\quad \leq E(f(x^k) - f(x^*))(1-\omega) + 2E(e_{\mathcal{N}_k}) + \theta E(\tilde{e}_{\mathcal{N}_k}) + \nu_k \\
&\qquad + \alpha \left(E(f(x^k) - f(x^*))(1 + 2\lambda_n/\lambda_1) + 2\lambda_n/\lambda_1 E(f(x^{k+1}) - f(x^*)) \right) \\
&\qquad + \alpha M_s E(\tilde{e}_{\mathcal{N}_k}).
\end{aligned}$$

Rearranging the previous inequality and assuming $\alpha < \frac{\lambda_1}{2\lambda_n}$ we obtain

$$E(f(x^{k+1}) - f(x^*)) \leq \tau E(f(x^k) - f(x^*)) + a_k,$$

where

$$\tau = \frac{1 - \omega + \alpha(1 + 2\lambda_n/\lambda_1)}{u}, \quad u = 1 - \alpha 2\lambda_n/\lambda_1$$

and

$$a_k = \frac{1}{u}(\nu_k + 2E(e_{\mathcal{N}_k}) + (\theta + \alpha M_s)E(\tilde{e}_{\mathcal{N}_k})).$$

Notice that $\tau \in (0, 1)$ provided that α is small enough. Moreover, as discussed in the previous proof, a_k tends to zero and the result follows. □

4. Numerical results

In this section we report on our numerical experience with subsampled IN approaches. The experiments were performed in MATLAB R2017a, on an Intel Core i5-6600K CPU 3.50 GHz x 4 16GB RAM. For the approximate solution of the linear systems we used the CG method implemented in the MATLAB function `pcg`. No preconditioner is employed and the CG is used in a matrix-free manner. Then only products of $\nabla f^2_{\mathcal{D}_k}$ with vectors are needed. The aim of this section is to provide numerical evidence of the benefits deriving from the employment of adaptive rules, streaming out from the presented theory, for choosing forcing terms and Hessian sample size. Full gradients and functions, i.e., $N_k = N$ for $k > 0$, are used and we compare the full IN (`FIN`) method with $\eta = 10^{-4}$, the subsampled Hessian (`SIN`) method with $\eta_k = 10^{-4}$ and $D_k = 0.3N$ for all k, the subsampled inexact method with adaptive choices of η_k and $D_k = 0.3N$ for all k (`SINA_FT`) and the subsampled inexact method with adaptive choices of η_k and D_k (`SINA_FT_Dk`). We also consider a subsampled method with constant $D_k = 0.3N$ and a maximum number of five iterations allowed in `pcg` (`SIN_cg5`).

In `SINA_FT` and `SINA_FT_Dk`, η_k is chosen as follows:

$$\begin{cases} \eta_k = \min\{0.1, \max\{|f(x_k) - m_{k-1}(s^k)|/\|\nabla f(x_{k-1})\|, 10^{-3}\}\}, \\ \eta_0 = 0.1, \end{cases}$$

with $m_{k-1}(s^k)$ given in (2.30). This choice is made according to (2.29). Finally, we choose the sample size D_k in `SINA_FT_Dk` as

$$D_k = \left\lceil \max\left\{c_0 D_0, \min\left\{c_1 \min\left\{\frac{1}{\eta_k^2}, \frac{1}{\|\nabla f(x_k)\|^2}\right\}, N\right\}\right\}\right\rceil \quad c_0, c_1 > 0.$$

The above rule is based on inequalities (3.2) and (3.4), so D_k is chosen inversely proportional to γ_k^2 given in (3.4) as suggested by the bounds in (3.2). The choice of constants c_0 and c_1 depends on the convergence behavior of the CG method. In fact, if CG converges fast, large values of D_k should be used, as the loss in the convergence rate due to less accurate second-order information is not compensated by the reduced cost of Hessian–vector products. On the other hand, when CG is slower, smaller values of D_k must be used as a large number of matrix–vector products are needed. Then c_0 and c_1 are chosen according to the following strategy. We set $c_0 = 1$ and $c_1 = 0.05$ in the case that at the previous iteration CG needed more than 20 iterations. Otherwise we set $c_0 = 2$ and $c_1 = 1$. Moreover, D_0 is set to $0.1N$. This choice is motivated by the fact that we allow D_k to change and increase; then we can start with a small D_k leaving the method free to adaptively modify it.

In all the subsampled methods the set $\mathcal{D}_k$ is chosen randomly, using the MATLAB function `randperm`. All the methods under comparison are in the framework of Algorithm GIN-R, i.e., the nonmonotone line search (2.7), with $c = 10^{-4}$ and $\nu_k = \max(1, f(x^0))/k^{1.1}$ is applied. This latter choice allows even larger step sizes than specified by the global convergence conditions (see also Diniz-Ehrhardt *et al.*, 2008 and Krejić & Krklec Jerinkić, 2015).

The problem we consider is the binary classification problem. We suppose we have at our disposal a training set composed of pairs $\{(a_i, b_i)\}$ with $a_i \in \mathbb{R}^n$, $b_i \in \{-1, +1\}$ and $i = 1, \ldots, N$, where b_i denotes the correct sample classification. We perform a logistic regression, then we consider as a training objective function the logistic loss with ℓ_2 regularization (Bollapragada *et al.*, 2018), i.e., in problem (1.1) we have

$$f_i(x) = \log c(x, \xi_i) + \lambda \|x\|^2, \qquad c(x, \xi_i) = 1 + e^{-b_i a_i^T x} \tag{4.1}$$

where $\xi_i = (a_i, b_i)$. Furthermore, the gradients and the Hessians have special forms

$$\nabla f_i(x) = \frac{(1 - c(x, \xi_i))}{c(x, \xi_i)} b_i a_i + 2\lambda x, \quad \nabla^2 f_i(x) = -\frac{1 - c(x, \xi_i)}{c^2(x, \xi_i)} a_i a_i^T + 2\lambda I. \tag{4.2}$$

Note that the evaluation of the full function f requires the evaluation of the quantities $(c(x, \xi_i) - 1) b_i a_i$ for $i = 1, \ldots, n$, and once these quantities have been computed they can be used for evaluating $\nabla f_i(x)$ and $\nabla^2 f_i(x)$, $i = 1, \ldots, n$. Then due to the form of the gradient and Hessian of each f_i given in (4.2), the evaluation of $\nabla f_i(x)$ comes for free and the evaluation of $\nabla^2 f_i$ times a vector is as expensive as evaluating $f_i(x)$. Then we evaluate the performance of the methods under comparison in terms of full function evaluations (FEV). We underline that one `pcg` iteration costs $\frac{D_k}{N}$ FEV. Finally, in (4.1) we set $\lambda = 1/N$.

We used the following three data sets:

- `CINA0` Causality Workbench Team (2008), $N = 16033$, $n = 132$;
- `Mushrooms` Lichman (2013), $N = 5000$, $n = 112$;

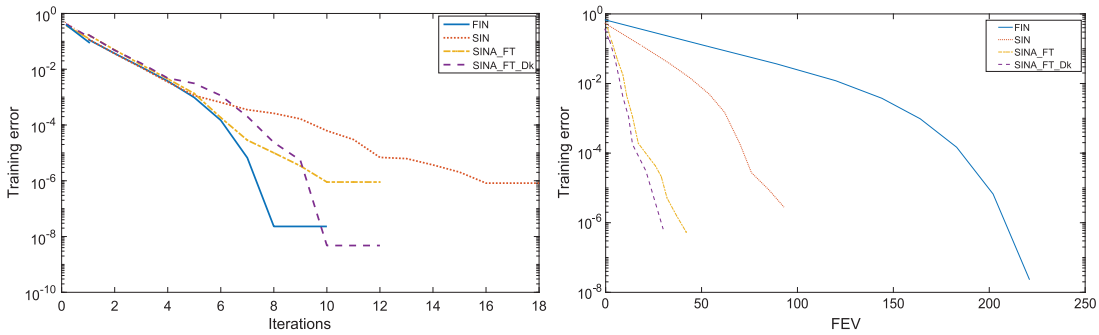

FIG. 1. `Mushrooms`, training error versus iteration (left) and versus FEV (right).

- `Gisette` Lichman (2013), $N = 6000$, $n = 5000$.

We stopped the methods under comparison when $\|\nabla f(x_k)\| \leq 10^{-4}$ or when a maximum number of 50 nonlinear iterations is reached. In order to compute the training error we compute the minimizer x^* with the full Newton method to a tight accuracy, i.e., we run it until $||\nabla f(x_k)||$ is less than 10^{-8}.

We begin by reporting our result with the `mushrooms` data set. We first underline that the linear algebra phase for this test is not demanding and the average number of CG iterations required is small; as an example it is around 10 when the adaptive choice of the forcing term is used. In Fig. 1 we plot $f_k - f(x^*)$ (training error) versus iterations (left) and versus function evaluations FEV (right). We observe that as expected the full IN (FIN) method is the fastest procedure. Moreover, the adaptive choice of the forcing terms seems to speed up the subsampled inexact procedure. Finally, remarkably, the procedure employing both the adaptive choice of η_k and D_k seems to work quite well. Indeed it is slower than `SINA_FT` in the first stage of the convergence history as it uses a smaller Hessian sample set. In the last stage of the procedure it becomes faster as the sample size increases (Fig. 2). On the other hand, if we look to the computational cost of the procedures we observe that `SINA_FT_Dk` outperforms all the procedures under comparison. We also note that `SINA_FT` outperforms `SIN`. Overall these results show the efficiency of the proposed adaptive strategies.

To give more insight into our adaptive choices we plot in Fig. 2 the values of the forcing terms η_k (left) and the value of D_k (right) versus iterations using the `SINA_FT_Dk`. We observe that whenever η_k becomes smaller D_k, increases and the approximation of the Hessian improves. Moreover, we note that the adaptive procedure allows the Hessian sample size to decrease when the model does not approximate the function sufficiently well and correspondingly the forcing term is increased.

We also compare, in Fig. 3, our adaptive procedure `SINA_FT_Dk` with `SIN_cg5`. It is interesting to note that the adaptive procedure is faster than `SIN_cg5` and greater accuracy can be reached. In terms of FEV, `SIN_cg5` is slightly better than `SINA_FT_Dk` until a training error value of the order 10^{-4}. If more accuracy is needed the adaptive procedure is clearly preferable.

Finally, in Fig. 4 we compare the behavior of testing errors versus iterations (left) and FEVs (right) along the sequences generated the by full Newton method `FIN`, the adaptive procedure `SINA_FT_Dk` and the `SIN_cg5` method. We evaluated the testing error as follows. Let x^k be the approximation computed by a method using the data in the training set. Then x^k is used to classify the samples in the

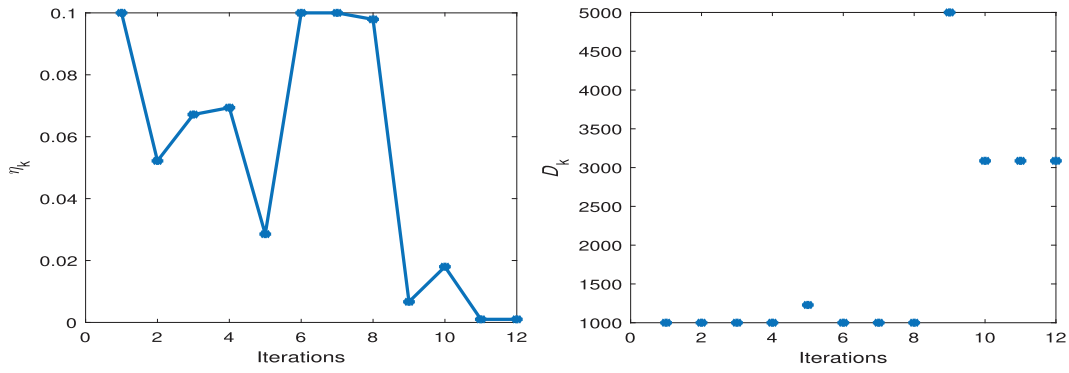

FIG. 2. `Mushrooms`, `SINA_FT_Dk`, η_k values (right) and D_k (left) versus iterations.

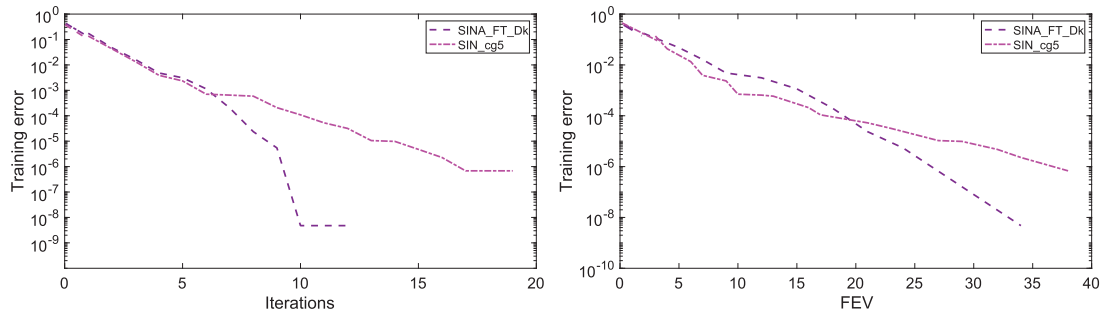

FIG. 3. `Mushrooms`, training error versus iteration (left) and versus FEV (right).

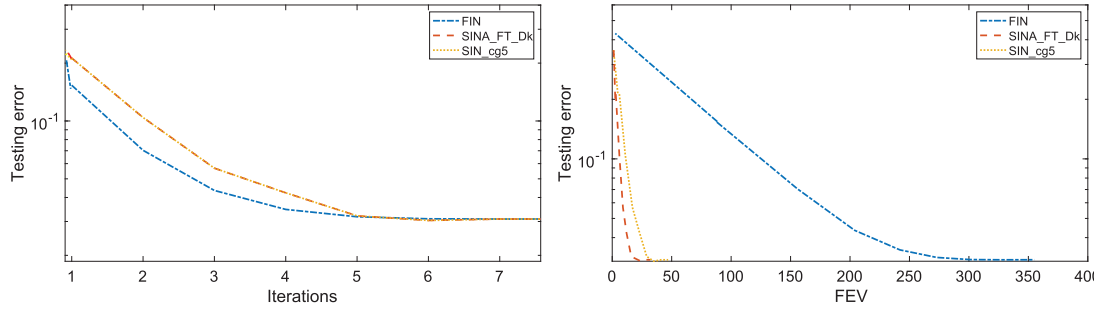

FIG. 4. `Mushrooms`, testing error versus iteration (left) and versus FEV (right).

testing set made up of $\bar{N} = 3124$ instances z_i, $i = 1, \ldots, \bar{N}$ and corresponding b_i. The classification error at iteration k is defined as $\frac{1}{\bar{N}} \sum_{i=1}^{\bar{N}} \log(1 + \exp(-b_i z_i^T x^k))$. We observe that the three methods reached almost the same testing errors, the subsampled approaches require a lower computational cost and again the adaptive procedure outperforms `SIN_cg5`. We also observe that the testing error quickly steadies.

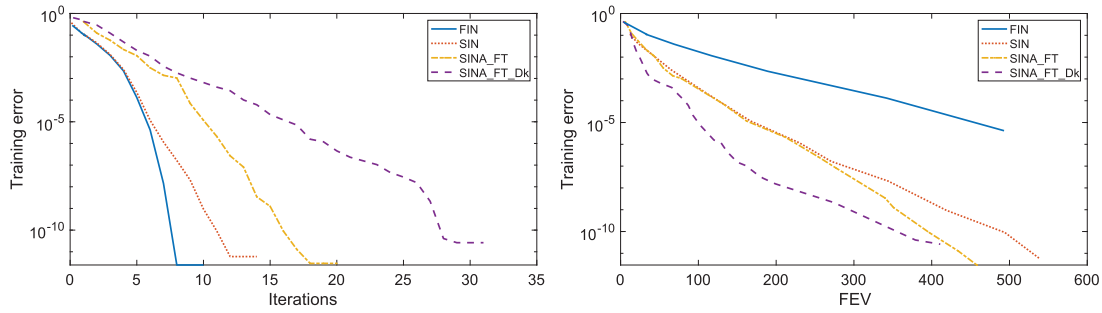

FIG. 5. CINA0, training error versus iteration (left) and versus FEV (right).

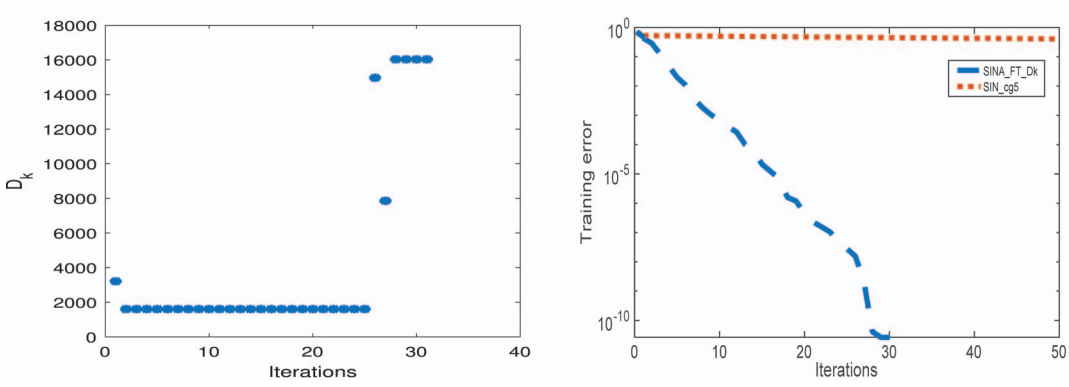

FIG. 6. CINA0 and SINA_FT_DK, D_k (left) and training error (right) versus iterations.

This shows that the stopping tolerance we used is tight enough and going on with the iterations would not decrease the testing error further.

Let us consider the CINA0 data set. In Fig. 5 we plot the training error versus iterations (left) and versus FEV (right). The FIN method is the fastest one, as expected. On the other hand, also in this case, it is the most expensive. The behavior of the subsampled procedures is the desiderable one and the adaptive procedures outperform SIN in terms of FEV. Note that SINA_FT_Dk is slower than the other two subsampled procedures, but less costly, as expected. In fact this is a problem where the linear algebra phase is more demanding, the average number of CG iterations using SINA_FT is about 70 and small values of D_k are used to limit the overall computational cost (Fig. 6, left). Since the convergence of CG is slow, SIN_cg5 is not able to converge with a reasonable rate and it can only provide a very rough accuracy (Fig. 6, right)

We finally show the results obtained with the Gisette data set, where n is larger than in the previous tests. In Fig. 7, we compare SINA_FT_DK, SIN and SIN_FT in terms of FEV (left) and we also report the behavior of the sample size along the iterations (right). We observe that SIN_FT and SINA_FT_DK are less expensive and more accurate than SIN; however, within 50 nonlinear iterations SINA_FT_DK is not able to produce an approximation as accurate as that provided by SIN_FT. In fact, the convergence is slow due to the small size of the Hessian subsample set. However, note that if a training error of the order of 10^{-2} is enough, SINA_FT_DK is the method of choice as it is less

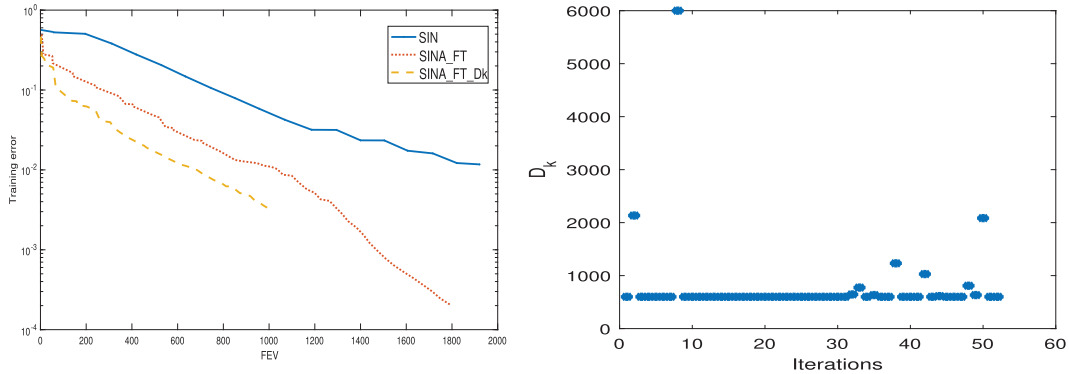

FIG. 7. `Gisette`, training error versus FEV (left) and D_k versus iterations (right).

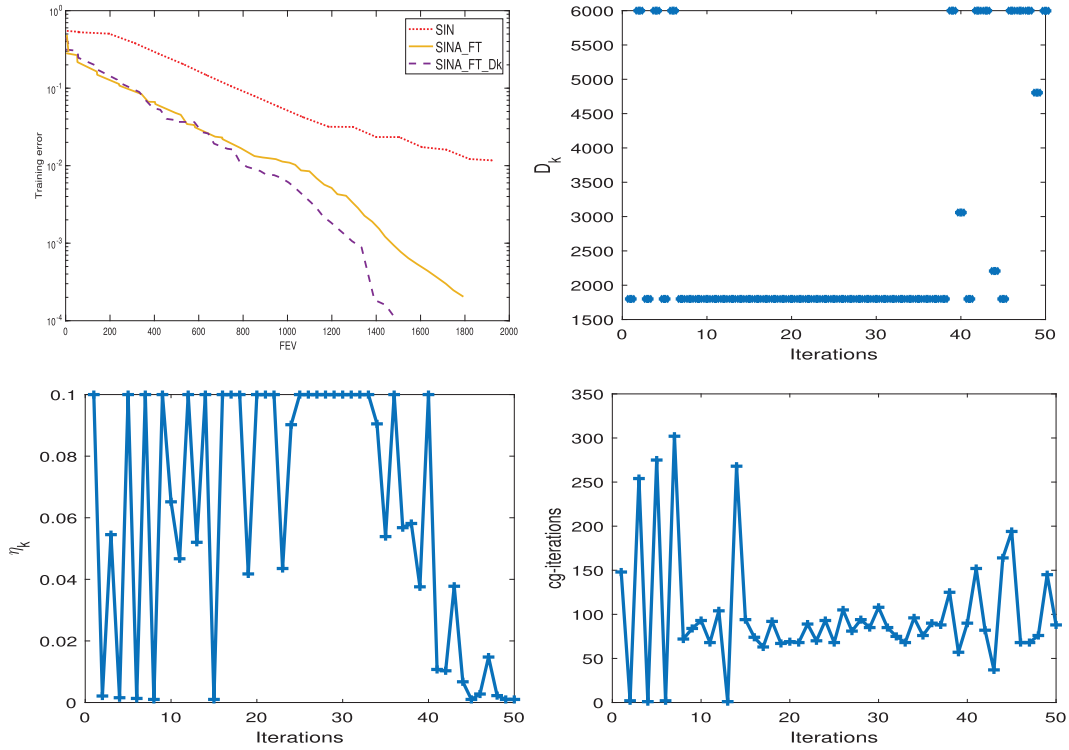

FIG. 8. `Gisette` (`SINA_FT_DK` with $D_0 = 0.3N$), training error versus FEV (top-left), D_k versus iterations (top-right), η_k versus iterations (bottom left) and CG iterations versus Newton iterations (bottom right).

expensive. In case a greater accuracy is needed, $D_0 = 0.3N$ should be used in `SINA_FT_DK`. Figure 8 refers to this choice of D_0. In this case `SINA_FT_DK` outperforms both `SIN` and `SIN_FT`. We also report the values of D_k, η_k and the number of CG iterations at each Newton iteration. Plots show that we use small values of D_k and large values of η_k until the last stage of convergence. Since we are

solving linear systems with a rough accuracy, the number of CG iterations is reasonable (except for a few occurrences) considering that $n = 5000$ and we are not employing a preconditioner.

As a final comment we observe that in SINA_FT and SIN different choices of the Hessian sample size can be adopted and those corresponding to $D_k < 0.3N$ are of particular interest. Then we performed runs also with $D_k = 0.1N$ for all k. The SIN method was not competitive, showing that adaptivity of the forcing term is crucial. The results obtained with SINA_FT are strongly dependent on the test problem; below we provide some statistics referring to the average number of FEV employed over 50 runs. In the solution of Gisette test, SINA_FT with $D_k = 0.1N$ is not able to reach the same level of accuracy obtained with $D_k = 0.3N$ as the Hessian subsample is too small. Solution of the Mushrooms test is 23% more expensive than SINA_FT with $D_k = 0.3N$ and 30% more expensive than SINA_FT_DK, showing that a fixed sample size larger than $D_k = 0.1N$ is preferable and the adaptivity makes the method able to recover from the non ideal choice $D_0 = 0.1N$ providing an overall faster solution. When applied to the CINA0 test, it is 40% less expensive than SINA_FT with $D_k = 0.3N$ and 11% less expensive than SINA_FT_DK. Then in this latter case the smaller fixed sample size $D_k = 0.1N$ is the best choice, but the adaptive rule provides reasonable values of the parameters, avoiding tuning for each data set.

Acknowledgements

The authors are grateful to the anonymous referees for their suggestions, which lead to significant improvement of the manuscript.

Funding

S.B. is a member of the INdAM Research Group-Gruppo Nazionale per il Calcolo Scientifico (GNCS). GNCS-Istituto Nazionale di Alta Matematica of Italy (to S.B). Serbian Ministry of Education Science and Technological Development grant no. 174030 (to N.K. and N.K.J.).

References

Bastin, F. (2004) Trust-region algorithms for nonlinear stochastic programming and mixed logit models. *Ph.D. Thesis*, University of Namur, Belgium.

Bastin F., Cirillo, C. & Toint, P. L. (2006a) An adaptive Monte Carlo algorithm for computing mixed logit estimators. *Comput. Manag. Sci.*, **3**, 55–79.

Bastin, F., Cirillo, C. & Toint, P. L. (2006b) Convergence theory for nonconvex stochastic programming with an application to mixed logit. *Math. Program.*, **108**, 207–234.

Bellavia, S., Gurioli, G. & Morini, B. (2018) Theoretical study of an adaptive cubic regularization method with dynamic inexact Hessian information. preprint arXiv:1808.06239.

Berahas, A. S., Bollapragada, R. & Nocedal, J. (2018) An investigation of Newton–sketch and subsampled Newton methods. preprint arXiv:1705.06211v3.

Birgin, E. G., Krejić, N. & Martínez, J. M. (2018) On the employment of inexact restoration for the minimization of functions whose evaluation is subject to programming errors. *Math. Comp.*, **87**, 1307–1326.

Bollapragada, R., Byrd, R. & Nocedal, J. (2018) Exact and inexact subsampled Newton methods for optimization. *IMA J. Numer. Anal.*, **39**, 545–578.

Bottou, L., Curtis, F. C. & Nocedal, J. (2017) Optimization methods for large-scale machine learning. preprint arXiv:1606.04838v2 [stat.ML].

Byrd, R. H., Chin, G. M., Neveitt, W. & Nocedal, J. (2011) On the use of stochastic Hessian information in optimization methods for machine learning. *SIAM J. Optim.*, **21**, 977–995.

Byrd, R. H., Chin, G. M., Nocedal, J. & Wu, Y. (2012) Sample size selection in optimization methods for machine learning. *Math. Program.*, **134**, 127–155.

Byrd, R. H., Hansen, S. L., Nocedal, J. & Singer, Y. (2016) A stochastic quasi-Newton method for large-scale optimization. *SIAM J. Optim.*, **26**, 1008–1021.

Causality Workbench Team (2008) *A Marketing Dataset*. Available at http://www.causality.inf.ethz.ch/data/CINA.html.

Chernoff, H. (1952) A measure of the asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Ann. Math. Stat.*, **23**, 493–507.

Dembo, R. S., Eisenstat, S. C. & Steinhaug, T. (1982) Inexact Newton method. *SIAM J. Numer. Anal.*, **19**, 400–409.

Deng, G. & Ferris, M. C. (2009) Variable-number sample path optimization. *Math. Program.*, **117**, 81–109.

Diniz-Ehrhardt, M. A., Martínez, J. M. & Raydan, M. (2008) A derivative-free nonmonotone line-search technique for unconstrained optimization. *J. Comput. Appl. Math.*, **219**, 383–397.

Eisenstat, S. C. & Walker, H. F. (1996) Choosing the forcing terms in an inexact Newton method. *SIAM J. Sci. Comput.*, **17**, 16–32.

Erdogdu, M. A. & Montanari, A. (2015) Convergence rates of sub-sampled Newton methods. *NIPS'15 Proceedings of the 28th International Conference on Neural Information Processing Systems*, vol. 2, Montreal, Canada, pp. 3052–3060.

Fountoulakis, K. & Godzio, J. (2016) A second order method for strongly convex ℓ_1-regularization problems. *Math. Program.*, **156**, 189–219.

Friedlander, M. P. & Schmidt, M. (2012) Hybrid deterministic–stochastic methods for data fitting. *SIAM J. Sci. Comput.*, **34**, 1380–1405.

Grapiglia, G. N. & Sachs, E. W. (2017) On the worst-case evaluation complexity of non-monotone line search algorithms. *Comput. Optim. Appl.*, **68**, 555–577.

Jakovetic, D., Xavier, J., Moura, J. M. F. (2014) Fast distributed gradient methods. *IEEE Trans. Automat. Control*, **59**, 1131–1146.

Kelley, C. T. (1995) *Iterative Methods for Linear and Nonlinear Equations*. Philadelphia: SIAM.

Krejić, N. & Krklec, N. (2013) Line search methods with variable sample size for unconstrained optimization. *J. Comput. Appl. Math.*, **245**, 213–231.

Krejić, N. & Krklec Jerinkić, N. (2015) Nonmonotone line search methods with variable sample size. *Numer. Algorithms*, **68**, 711–739.

Krejić, N. & Martínez, J. M. (2016) Inexact restoration approach for minimization with inexact evaluation of the objective function. *Math. Comp.*, **85**, 1775–1791.

Lee, J. D., Sun, Y. & Saunders, M. A. (2014) Proximal Newton type methods for minimizing composite functions. *SIAM J. Optim.*, **24**, 1420–1443.

Li, D. H. & Fukushima, M. (2000) A derivative-free line search and global convergence of Broyden-like method for nonlinear equations. *Optim. Methods Softw.*, **13**, 181–201.

Lichman, M. (2013) *UCI Machine Learning Repository*. Available at https://archive.ics.uci.edu/ml/index.php.

Minsker, S. (2017) On some extensions of Bernstein's inequality for self-adjoint operators. arXiv:1112.5448 [math.PR].

Nash, S. G. (2000) A survey of truncated-Newton methods. *J. Comput. Appl. Math.*, **124**, 45–59.

Nesterov, Y. (2013) *Introductory Lectures on Convex Optimization: A Basic Course, Applied Optimization* vol. 87. New York: Springer Science and Media.

Pasupathy, R. (2010) On choosing parameters in retrospective-approximation algorithms for stochastic root finding and simulation optimization. *Oper. Res.*, **58**, 889–901.

Pilanci, M. & Wainwright, M. J. (2017) Newton sketch: a near linear-time optimization algorithm with linear-quadratic convergence. *SIAM J. Optim.*, **27**, 205–245.

Polak, E., Royset, J. O. (2008) Efficient sample sizes in stochastic nonlinear programing. *J. Comput. Appl. Math.*, **217**, 301–310.

Roosta-Khorasani, F. & Mahoney, M. W. (2019) Sub-sampled Newton methods. *Math. Program.*, **174**, 293–326.

Shapiro, A., Dentcheva, D. & Ruszczynski, A. (2009) Lectures on stochastic programming: modeling and theory. *Optimization*, **9**.

Tropp, J. A. (2012) User-friendly tail bounds for sums of random matrices. *Found. Comput. Math.*, **12**, 389–434.

Xu, P., Yang, J., Roosta-Khorasani, F., Ré, C. & Mahoney, M. W. (2016) Sub-sampled Newton methods with non-uniform sampling. *Advances in Neural Information Processing Systems (NIPS)*, vol. 30, Los Angeles: Neural Information Processing Systems Foundation, pp. 2530–2538.

Xu, P., Roosta-Khorasani, F. & Mahoney, M. W. (2018) Newton-type methods for non-convex optimization under inexact Hessian information. preprint arxiv:1708.07164v3 [math.OC].

Appendix

Proofs of Lemmas 2.13, 2.14 and 3.1

Proof of Lemma 2.13. Using the mean value theorem we get the following inequality: $\|\nabla f_{\mathcal{N}}(x^k+s^k) - \nabla f_{\mathcal{N}}(x^k) - \nabla^2 f_{\mathcal{N}}(x^k)s^k\| \leq \frac{L}{2}\|s^k\|^2$. Then

$$\begin{aligned}\|\nabla f_{\mathcal{N}}(x^k+s^k) - \nabla f_{\mathcal{N}}(x^k) - \nabla^2 f_{\mathcal{D}}(x^k)s^k\| &\leq \|\nabla f_{\mathcal{N}}(x^k+s^k) - \nabla f_{\mathcal{N}}(x^k) - \nabla^2 f_{\mathcal{N}}(x^k)s^k\| \\ &\quad + \|\nabla^2 f_{\mathcal{D}_k}(x^k) - \nabla^2 f_{\mathcal{N}}(x^k)\|\|s^k\| \\ &\leq \|s^k\|(h(D_k,x^k) + \tfrac{1}{2}L\|s^k\|).\end{aligned}$$

Therefore,

$$\begin{aligned}\|\nabla f_{\mathcal{N}}(x^k+s^k)\| &\leq \|\nabla f_{\mathcal{N}}(x^k) + \nabla^2 f_{\mathcal{D}_k}(x^k)s^k\| + \|\nabla f_{\mathcal{N}}(x^k+s^k) - \nabla f_{\mathcal{N}}(x^k) - \nabla^2 f_{\mathcal{D}_k}(x^k)s^k\| \\ &\leq \eta\|\nabla f_{\mathcal{N}}(x^k)\| + \|s^k\|(\tfrac{1}{2}L\|s^k\| + h(D_k,x^k))\end{aligned}$$

and, by Lemma 2.3,

$$\|\nabla f_{\mathcal{N}}(x^k+s^k)\| \leq \eta\|\nabla f_{\mathcal{N}}(x^k)\| + \frac{1}{\lambda_1}\|\nabla f_{\mathcal{N}}(x^k)\|\Big(\frac{1}{2\lambda_1}L\|\nabla f_{\mathcal{N}}(x^k)\| + h(D_k,x^k)\Big).$$

□

Proof of Lemma 2.14. By Lemma 2.3 and (2.6) we have

$$\begin{aligned}\|x^k+s^k-x^*\| &\leq \|x^k-x^*\| + \|s^k\| \leq \|x^k-x^*\| + \lambda_1^{-1}\|\nabla f_{\mathcal{N}}(x^k)\| \\ &\leq \|x^k-x^*\| + \lambda_1^{-1}\lambda_n\|x^k-x^*\| \leq (1+\lambda_1^{-1}\lambda_n)\|x^k-x^*\| \leq \delta^*.\end{aligned}$$

□

Proof of Lemma 3.1. Let us define $Y_i(x) = (H_i(x) - \nabla^2 f_{\mathcal{N}}(x))/D$ where $H_i(x)$ is a randomly chosen Hessian. Then

$$\sum_{i=1}^{D} Y_i(x) = \nabla^2 f_{\mathcal{D}}(x) - \nabla^2 f_{\mathcal{N}}(x).$$

Also, notice that $E(Y_i(x)) = 0$ and that the Weyl inequality yields $\lambda_{\max}(Y_i(x)) \leq \lambda_n/D$ and $\lambda_{\min}(Y_i(x)) \geq -\lambda_n/D$. Then $\|Y_i(x)\| \leq \lambda_n/D$ and

$$\tilde{\sigma}^2(x) := \left\| \sum_{i=1}^{D} E(Y_i^2(x)) \right\| \leq \sum_{i=1}^{D} E(\|Y_i(x)^2\|) \leq \frac{\lambda_n^2}{D}.$$

Then using the Bernstein's inequality (Tropp, 2012, Theorem 1.6) we derive

$$P(\|\nabla^2 f_{\mathcal{D}}(x) - \nabla^2 f_{\mathcal{N}}(x)\| \geq \gamma) \leq 2n \ \exp\left(-\frac{\gamma^2/2}{\lambda_n^2/D + (\lambda_n/D)\gamma/3}\right).$$

This yields the bound. □