Mathematical Programming (2019) 174:293–326
https://doi.org/10.1007/s10107-018-1346-5

FULL LENGTH PAPER

Series B

Sub-sampled Newton methods

Farbod Roosta-Khorasani[1] · **Michael W. Mahoney**[2]

Received: 14 March 2017 / Accepted: 27 October 2018 / Published online: 16 November 2018

Abstract
For large-scale finite-sum minimization problems, we study non-asymptotic and high-probability global as well as local convergence properties of variants of Newton's method where the Hessian and/or gradients are randomly sub-sampled. For Hessian sub-sampling, using random matrix concentration inequalities, one can sub-sample in a way that second-order information, i.e., curvature, is suitably preserved. For gradient sub-sampling, approximate matrix multiplication results from randomized numerical linear algebra provide a way to construct the sub-sampled gradient which contains as much of the first-order information as possible. While sample sizes all depend on problem specific constants, e.g., condition number, we demonstrate that local convergence rates are *problem-independent*.

Keywords Newton-type methods · Local and global convergence · Sub-sampling

Mathematics Subject Classification 49M15 · 65K05 · 90C25 · 90C06

1 Introduction

Consider the convex optimization problem

$$\min_{\mathbf{x}\in\mathscr{D}\cap\mathscr{C}} F(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}), \tag{1}$$

where $\mathscr{C} \subseteq \mathbb{R}^p$ is a convex constraint set and $\mathscr{D} = \bigcap_{i=1}^{n} \mathrm{dom}(f_i)$ is a convex and open domain of the strongly convex objective F. Many data fitting applications can

✉ Farbod Roosta-Khorasani
fred.roosta@uq.edu.au

Michael W. Mahoney
mmahoney@stat.berkeley.edu

[1] School of Mathematics and Physics, University of Queensland, St Lucia, QLD, Australia

[2] ICSI and Department of Statistics, UC Berkeley, Berkeley, CA, USA

be expressed as (1) where each f_i corresponds to an observation (or a measurement) which models the loss (or misfit) given a particular choice of the underlying parameter $\mathbf{x}$, e.g., empirical risk minimization in machine learning including softmax classification, support vector machines, and graphical models, among many others. Many optimization algorithms have been developed to solve (1), [3,6,32]. Here, we consider the regime where $n, p \gg 1$. In such high dimensional settings, the mere evaluation of the gradient or the Hessian of F can be computationally prohibitive. As a result, many of the classical *deterministic* optimization algorithms might prove to be inefficient, if applicable at all. In this light, faced with modern "big data" problems, there has been a great deal of effort to design *stochastic* variants which are efficient and inherit much of the "nice" convergence behavior of the original deterministic counterparts.

Many of these stochastic algorithms employ *sub-sampling* to speed up the computations. Within the class of first order methods, i.e., those which only use gradient information, there are many such algorithms with various kinds of theoretical guarantees. However, for second order methods, i.e., those that employ both the gradient and the Hessian information, studying the theoretical properties of such sub-sampled algorithms lags behind. In this paper, we provide a detailed analysis of *sub-sampling* as a way to leverage the "magic of randomness" in the classical Newton's method and study variants that are more suited for modern large-scale problems.

The common occurring theme in our approach is the *high-probability* and *non-asymptotic* convergence analysis. This is so since high-probability analysis allows for a small, yet non-zero, probability of occurrence of "bad events" in each iteration. Hence, the accumulative probability of occurrence of "good events" at all times becomes smaller with increasing iterations, and in fact is asymptotically zero for an infinite sequence of iterates. As a result, although the term "convergence" typically implies the asymptotic limit of an infinite sequence, here we consider non-asymptotic behavior of a finite number of *random* iterates and provide high-probability results on their properties. For example, we study whether, with high-probability, a finite set of random iterates generated by an algorithm approaches the solution of (1) and, if so, at what rate. Henceforth, we use the term "convergence" loosely in this sense.

Under such an analytic framework, our theoretical results are delivered in two stages. At first, we take a "coarse-grained" approach and provide a variety of conditions under which the convergence is *global*, i.e., random iterations are guaranteed to converge, with high-probability, to the solution of (1) starting from any initial point. In the second stage, we zoom-in and employ a "finer-grained" approach to study non-asymptotic *local convergence* of these sub-sampled methods, i.e., when the initial iterate is chosen in a neighborhood "close enough" to the solution of (1). This is so since the theoretical appeal of many second-order methods mainly lies in their local convergence behaviors, e.g., local quadratic convergence of the classical Newton's method. Indeed, any method claiming to be second-order must be accompanied by similar superior local convergence properties. In this light, the "big-picture" contribution of this paper is to map the lay of the land for the non-asymptotic local and global convergence properties of variants of Newton's method in which the Hessian and/or gradient are randomly sub-sampled.

The rest of the paper is organized as follows. In Sect. 1.1, we first give a brief overview of the general framework for the iterative schemes considered in our anal-

ysis. In Sect. 1.2, we briefly survey the related works, and in their light, discuss our contributions in Sect. 1.3. The notation and the assumptions used in this paper are given in Sects. 1.4 and 1.5, respectively. Section 2 addresses sampling strategies for approximating the Hessian and gradient. The non-asymptotic and high-probability convergence results for sub-sampled Newton's methods are given in Sect. 3. In particular, global and local convergence properties are treated in Sects. 3.1and 3.2, respectively, followed by some unifying results in Sect. 3.3. Worst-case computational complexities involving various parts of these algorithms are gathered in Sect. 3.4. Conclusions and further thoughts are gathered in Sect. 4.

1.1 Framework

The iterative framework under which we study the sub-sampled Newton-type variants is what is best known as *scaled gradient projection* formulation [3]. More specifically, given the current iterate, $\mathbf{x}^{(k)} \in \mathscr{D} \cap \mathscr{C}$, consider the following iterative scheme,

$$\widehat{\mathbf{x}}^{(k)} = \arg\min_{\mathbf{x}\in\mathscr{D}\cap\mathscr{C}} \left\{ (\mathbf{x}-\mathbf{x}^{(k)})^T \mathbf{g}(\mathbf{x}^{(k)}) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^{(k)})^T H(\mathbf{x}^{(k)})(\mathbf{x}-\mathbf{x}^{(k)}) \right\}, \quad (2a)$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \left(\widehat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k)} \right), \quad (2b)$$

where $\mathbf{g}(\mathbf{x}^{(k)})$ and $H(\mathbf{x}^{(k)})$ are some approximations to (in our case, sub-samples of) the actual gradient and the Hessian at the k^{th} iteration, respectively, and α_k is the step-size. A variety of first and second order methods can be written in this form. For example, classical Newton's method is obtained by setting $\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)})$ and $H(\mathbf{x}^{(k)}) = \nabla^2 F(\mathbf{x}^{(k)})$, the (projected) gradient descent is with $\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)})$ and $H(\mathbf{x}^{(k)}) = \mathbb{I}$, and the pair

$$\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)}),\ H(\mathbf{x}^{(k)}) = \frac{1}{|\mathscr{S}_H|} \sum_{j\in\mathscr{S}_H} \nabla^2 f_j(\mathbf{x}^{(k)}), \quad \text{or}$$

$$\mathbf{g}(\mathbf{x}^{(k)}) = \frac{1}{|\mathscr{S}_\mathbf{g}|} \sum_{j\in\mathscr{S}_\mathbf{g}} \nabla f_j(\mathbf{x}^{(k)}),\ H(\mathbf{x}^{(k)}) = \frac{1}{|\mathscr{S}_H|} \sum_{j\in\mathscr{S}_H} \nabla^2 f_j(\mathbf{x}^{(k)}),$$

for some index sets $\mathscr{S}_\mathbf{g}, \mathscr{S}_H \subseteq [n]$ gives rise to *sub-sampled Newton methods* [5,7,8, 18,42], henceforth referred to as SSN, which are the focus of this paper. Depending on the method, the step-size α_k is sometimes set to a predefined value, or alternatively is chosen adaptively, e.g., using line-search techniques.

Our primary objective in this paper is to study conditions under which variants of SSN are not only *efficient* for large-scale problems, but also preserve, at least *locally*, as much of the *superior* convergence properties of the classical Newton's method as possible while maintaining a reasonable *global* convergence behavior. In doing so, we need to *simultaneously* ensure the following requirements.

(R.1) Our sampling strategy needs to provide a sample size which is *independent* of n, or at least *smaller*, e.g, grows slowly with n.

(R.2) To re-scale the gradient direction appropriately, the sub-sampled matrix must *preserve the spectrum* of the full Hessian as much as possible. At the very least, it should be able to generate *descent directions*, e.g., when $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$, if the original matrix is uniformly positive definite (PD), so should be the sub-sampled approximation (preferably without any additional regularization to the Hessian). Approximation to the gradient should also contain as much of the first order information as possible.

(R.3) Such algorithms need to be globally convergent and approach the optimum starting from any initial guess. More importantly, for any variant of SSN to be considered "Newton-like", it must enjoy a reasonably *fast convergence rate* which is, at least locally, similar to that of the classical Newton's method.

(R.4) In high-dimensional regimes where $p \gg 1$, solving (2a) exactly at each iteration can pose a significant computational challenge. In such settings, allowing for (2a) to be solved *inexactly* is indispensable.

In this paper, we strive to, at least partially, address challenges (R.1)–(R.4). More precisely, by using random matrix concentration inequalities and results from approximate matrix multiplication of randomized numerical linear algebra (RandNLA) [28], we aim to ensure (R.1) and (R.2). For (R.3), we give variants of SSN which are globally convergent and whose local convergence rates can be made close to that of the classical Newton's method. Finally, to satisfy (R.4), for both global and local convergence, we give inexactness requirements, which are less strict than prior works.

1.2 Related work

Randomized approximation of the full Hessian matrix has been previously considered in [1,5,7,8,18,29,31,35,40,42].Within the context of deep learning, [29] is the first to suggest a heuristic sub-sampled Newton-type algorithm and study its empirical performance. The pioneering work in [7,8] establishes, for the first time, the convergence of variants of Newton's method with the sub-sampled Hessian. However, the results are asymptotic and no quantitative convergence rate is given. In addition, convergence is established for the case where each f_i is assumed to be strongly convex. Under the same setting, some modifications and improvements are given in [40]. The work in [35] is the first to use "sketching" within the context of Newton-like methods, specialized to the cases where some square root of the Hessian matrix is readily available. Under the same setting, non-uniform sampling strategies are proposed in [42]. Non-asymptotic local convergence rates for the uniform sub-sampling of the Hessian is first established in [18]. The authors suggest an algorithm where, at each iteration, the spectrum of the uniformly sub-sampled Hessian is modified as a form of regularization. In [1] a Hessian sub-sampling algorithm is proposed that employs unbiased estimator of the inverse of the Hessian. This is followed by an improved and simplified convergence analysis in [31]. Arguably, the closest results to those of the present paper are given in [5]. However, the convergence results in [5] are given in expectation whereas, here, we give high probability results. In addition, [5] assumes that each f_i in (1) is strongly convex, while here, we only make such an assumption for the objective function, F, and each f_i need only be (weakly) convex.

Within the context of second order methods, gradient sub-sampling has been successfully applied in large scale machine learning, e.g., [5,7,8], as well as nonlinear inverse problems, e.g., [22,36]. By carefully increasing the sample size across iterations, the hybrid methods of [19] combine the inexpensive iterations of incremental gradient algorithms and the convergence of full gradient methods. Such careful increase of the sample size is one of the main ingredients of our analysis.

Finally, inexact updates have been used in many second-order optimization algorithms; see [5,9,14,17,26] and references therein.

1.3 Contributions

In light of the related work, the main contributions of this paper are summarized as follows; see Table 1. We first consider the unconstrained version of (1) where $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$, and under certain assumptions, study non-asymptotic and high-probability global convergence of SSN with Armijo line search. Specifically, we consider the variants of SSN depicted in Algorithms 1 and 2. Theorems 1 and 3 give global convergence results for the case where the linear system arising from (2a) is solved exactly, Theorems 2 and 4 give similar results for inexact updates. For all these results, we only require that the objective F in (1) is strongly convex, while each component function, f_i, is allowed to be only (weakly) convex. This is a relaxation over all previous work where strong convexity is assumed for *all* f_i's.

We then zoom in, and in full generality, i.e., omitting the assumption $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$, study non-asymptotic and high-probability local convergence behavior of SSN using the natural step size of the classical Newton's method, i.e., $\alpha_k = 1$ in (2b). Specifically, for Algorithms 3, 4, and 5, we establish non-asymptotic equivalents of Q-linear, Q-superlinear, and R-linear convergence results in Theorem 6, 7, 8, and 11. Though the sample size depends on the condition number of the problem, we show that the local rates for the (super)linear convergence phase are, in fact, problem-independent, a property common to most Newton-type methods. Fast and problem-independent local convergence rates using the inexact solution of (2a) (in unconstrained case) are studied in Theorems 9 and 12.

Compared to similar results, e.g., [5,42], our inexactness tolerance is a significant improvement. More specifically, in prior works, the inexactness tolerance depends

Table 1 Summary of the main results. "Lin" and "SupLin", respectively, are short for "Linear" and "Superlinear". "Exact" and "Inexact", refer to the exact and approximate solution of (2a), respectively

Summary of Results					
$\mathbf{g}(\mathbf{x})$ and $H(\mathbf{x})$ in (2a)	Global		Local		Unifying Results
	Algorithms	Theorems	Algorithms	Theorems	
$\mathbf{g}(\mathbf{x})$ = Full $H(\mathbf{x})$ = Sampled	1	1 (Exact) 2 (Inexact)	3, 4	6 (Q-Lin, Exact) 7, 8 (Q-SupLin, Exact) 9 (Q-Lin, Inexact)	13
$\mathbf{g}(\mathbf{x})$ = Sampled $H(\mathbf{x})$ = Sampled	2	3 (Exact) 4 (Inexact)	5	11 (R-Lin, Exact) 12 (R-Lin, Inexact)	14

on the inverse of the condition number, where as here, our tolerance depends on the *square root* of this inverse (Theorems 2, 4, 9, and 12).

We finally combine these global and local analysis and obtain unifying results (Theorems 13 and 14), which ensure that SSN with Armijo line search is globally convergent with problem-independent local rate. Further, after certain number of iterations, the line-search automatically adopts the natural step size of the classical Newton's method, i.e., $\alpha_k = 1$, for *all subsequent iterations*.

1.4 Notation

Throughout this paper, vectors are denoted by bold lowercase letters, e.g., $\mathbf{v}$, and matrices or random variables are denoted by regular upper case letters, e.g., V, which is clear from the context. For a vector $\mathbf{v}$, and a matrix V, $\|\mathbf{v}\|$, $\|V\|$ and $\|V\|_F$, respectively, denote the vector ℓ_2 norm, the matrix spectral norm, and the matrix Frobenius norm. $\nabla f(\mathbf{x})$ and $\nabla^2 f(\mathbf{x})$ are the gradient and the Hessian of f at $\mathbf{x}$, respectively. For a set $\mathscr{X}$ and two symmetric matrices A and B, $A \preceq_{\mathscr{X}} B$ indicates that $\mathbf{x}^T(B-A)\mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathscr{X}$. The superscript, e.g., $\mathbf{x}^{(k)}$, denotes iteration counter and $\ln(x)$ denotes the natural logarithm of x. Throughout the paper, $\mathscr{S}$ denotes a collection of indices from $[n] := \{1, 2, \ldots, n\}$, with potentially repeated items and $|\mathscr{S}|$ denote its cardinality. The cone of feasible directions at the optimum $\mathbf{x}^\star$ is denoted by

$$\mathscr{K} := \left\{ \mathbf{p} \in \mathbb{R}^p;\ \exists t > 0 \text{ s.t. } \mathbf{x}^\star + t\mathbf{p} \in \mathscr{D} \cap \mathscr{C} \right\}. \tag{3}$$

If $\mathbf{x}^\star$ lies in the relative interior of $\mathscr{D} \cap \mathscr{C}$, then, as a consequence of Prolongation Lemma [4, Lemma 1.3.3], it is easy to show that $\mathscr{K}$ is a subspace. For a vector $\mathbf{v}$ and a matrix A, using $\mathscr{K}$, we can define their $\mathscr{K}$-restricted seminorms, respectively, as

$$\|\mathbf{v}\|_{\mathscr{K}} := \sup_{\mathbf{p} \in \mathscr{K} \setminus \{0\}} \frac{|\mathbf{p}^T \mathbf{v}|}{\|\mathbf{p}\|}, \quad \text{and} \quad \|A\|_{\mathscr{K}} := \sup_{\mathbf{p},\mathbf{q} \in \mathscr{K} \setminus \{0\}} \frac{|\mathbf{p}^T A \mathbf{q}|}{\|\mathbf{p}\|\|\mathbf{q}\|}. \tag{4}$$

Similarly, one can define the $\mathscr{K}$-restricted maximum and the minimum eigenvalues of a symmetric matrix A as

$$\lambda^{\mathscr{K}}_{\min}(A) := \inf_{\mathbf{p} \in \mathscr{K} \setminus \{0\}} \frac{\mathbf{p}^T A \mathbf{p}}{\|\mathbf{p}\|^2}, \quad \text{and} \quad \lambda^{\mathscr{K}}_{\max}(A) := \sup_{\mathbf{p} \in \mathscr{K} \setminus \{0\}} \frac{\mathbf{p}^T A \mathbf{p}}{\|\mathbf{p}\|^2}. \tag{5}$$

Alternatively, let U be an orthonormal basis for the subspace $\mathscr{K}$. The definitions above are equivalent to $\|\mathbf{v}\|_{\mathscr{K}} = \|U^T \mathbf{v}\|$, $\|A\|_{\mathscr{K}} = \|U^T A U\|$. Also, this representation allows us to define any $\mathscr{K}$-restricted eigenvalue of A as $\lambda_i^{\mathscr{K}}(A) = \lambda_i(U^T A U)$, where $\lambda_i(A)$ is the usual i^{th} eigenvalue of A, i.e., computed with respect to all vectors in $\mathbb{R}^p$.

For the non-asymptotic high-probability analysis in this paper, we use the following adaptations of the classical notions of convergence, which are typically applied asymptotically. Recall that we use the term "convergence" loosely for a finite sequence. To evaluate convergence, one typically uses an appropriate distance

measure, $d : \mathbb{R}^p \times \mathbb{R}^p \to [0, \infty)$, such that $d(\mathbf{z}, \mathbf{z}) = 0$. Here, we consider $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$ and $d(\mathbf{x}, \mathbf{y}) = |F(\mathbf{x}) - F(\mathbf{y})|$. Let $\mathbf{z}^{(k)}$ denote the iterate generated by an iterative algorithm at the k^{th} iteration. An algorithm's iterations are said to converge Q-linearly to a limiting value $\mathbf{z}^\star$ if $\exists \rho \in [0, 1)$ such that $\forall k_0 \in \mathbb{N} = \{1, 2, \ldots\}$, the iterates generated by k_0 iterations of the algorithm starting from $\mathbf{z}^{(0)}$ satisfy $d(\mathbf{z}^{(k+1)}, \mathbf{z}^\star) \leq \rho d(\mathbf{z}^{(k)}, \mathbf{z}^\star),\ k = 0, 1, \ldots, k_0 - 1$. Q-superlinear convergence is defined similarly by requiring that $\exists \rho : \mathbb{N} \to [0, 1),\ \rho(k) \downarrow 0$, s.t. $\forall k_0 \in \mathbb{N}$, $d(\mathbf{z}^{(k+1)}, \mathbf{z}^\star) \leq \rho(k) d(\mathbf{z}^{(k)}, \mathbf{z}^\star),\ k = 0, 1, \ldots, k_0 - 1$. The notion of R-convergence rate is an extension which captures sequences which still converge reasonably fast, but whose "speed" is variable. An algorithm's iterations are said to converge R-linearly to $\mathbf{z}^\star$ if $\exists \rho \in [0, 1),\ \exists R > 0$ such that $\forall k_0 \in \mathbb{N}$, $d(\mathbf{z}^{(k)}, \mathbf{z}^\star) \leq R\rho^k,\ k = 1, 2, \ldots, k_0$. Clearly, for $k_0 = \infty$, these notions imply the typical asymptotic definitions. For randomized algorithms considered in this paper, we study conditions under which these algorithms, with high-probability, generate iterates that satisfy the corresponding convergence criteria.

1.5 Assumptions

We assume that each f_i is twice-differentiable, smooth and convex with respect to the cone $\mathscr{K}$, i.e., we have

$$0 \leq \inf_{\mathbf{x} \in \mathscr{D} \cap \mathscr{C}} \lambda_{\min}^{\mathscr{K}} \left(\nabla^2 f_i(\mathbf{x}) \right) \leq \sup_{\mathbf{x} \in \mathscr{D} \cap \mathscr{C}} \lambda_{\max}^{\mathscr{K}} \left(\nabla^2 f_i(\mathbf{x}) \right) \triangleq K_i < \infty, \quad i \in [n], \tag{6a}$$

where $\lambda_{\min}^{\mathscr{K}}(A)$ and $\lambda_{\max}^{\mathscr{K}}(A)$ are defined in (5). We further assume that F is smooth and strongly convex, i.e., we have

$$0 < \gamma \triangleq \inf_{\mathbf{x} \in \mathscr{D} \cap \mathscr{C}} \lambda_{\min}^{\mathscr{K}} \left(\nabla^2 F(\mathbf{x}) \right) \leq \sup_{\mathbf{x} \in \mathscr{D} \cap \mathscr{C}} \lambda_{\max}^{\mathscr{K}} \left(\nabla^2 F(\mathbf{x}) \right) \triangleq K < \infty, \tag{6b}$$

and it has a Lipschitz continuous Hessian with respect to $\mathscr{K}$, i.e.,

$$\sup_{\substack{\mathbf{x} - \mathbf{y} \in \mathscr{K} \\ \mathbf{x} \neq \mathbf{y}}} \frac{\|\nabla^2 F(\mathbf{x}) - \nabla^2 F(\mathbf{y})\|_{\mathscr{K}}}{\|\mathbf{x} - \mathbf{y}\|} \triangleq L < \infty, \tag{7}$$

where $\|A\|_{\mathscr{K}}$ is defined in (4). Since $\mathscr{D} \cap \mathscr{C}$ is convex, Assumption (6b) implies the *uniqueness* of the optimum, $\mathbf{x}^\star$. We further assume that $\mathbf{x}^\star$ lies in the relative interior of $\mathscr{D} \cap \mathscr{C}$. The quantity

$$\kappa \triangleq K/\gamma, \tag{8}$$

is known as the condition number of the problem, *restricted* to vectors in $\mathscr{K}$. Note that, depending on $\mathscr{K}$, (8) might be significantly smaller than the usual condition number, which is usually defined using all vectors in $\mathbb{R}^p$. For example, consider the case where $\mathscr{D} = \mathbb{R}^p$ and $\mathscr{C} = \{\mathbf{x} \in \mathbb{R}^p;\ A\mathbf{x} = \mathbf{b}\}$ for some matrix $A \in \mathbb{R}^{m \times p}$ with $m < p$ that

has full row-rank . Then $\mathscr{K}$ is the null space of A, i.e., $\mathscr{K} = \{\mathbf{p} \in \mathbb{R}^p;\ A\mathbf{p} = 0\}$, and $\text{rank}(\mathscr{K}) = p - m$. As a result, one can compute K and γ using Rayleigh quotients of $\nabla^2 f_i(\mathbf{x})$ and $\nabla^2 F(\mathbf{x})$ for all $\mathbf{x} \in \mathscr{D} \cap \mathscr{C}$, respectively, but restricted to vectors in this $p-m$ dimensional sub-space. Depending on A, these values can be much smaller than the minimum and maximum of such Rayleigh quotients over the entire $\mathbb{R}^p$. Of course, in an unconstrained problem, κ coincides with the usual condition number.

For an integer $1 \leq q \leq n$, let $\mathscr{Q}$ be the set of indices corresponding to q largest K_i's and define the "sub-sampling" condition number as

$$\kappa_q \triangleq \widehat{K}_q/\gamma, \tag{9a}$$

where

$$\widehat{K}_q \triangleq \frac{1}{q}\sum_{j\in\mathscr{Q}} K_j. \tag{9b}$$

It is easy to see that $K \leq \widehat{K}_n$ and for any two integers q and r such that $1 \leq q \leq r \leq n$, we have $\kappa \leq \kappa_r \leq \kappa_q$. Finally, define

$$\tilde{\kappa} \triangleq \begin{cases} \kappa_1, & \text{If sample } \mathscr{S} \text{is drawn } \textit{with} \text{ replacement} \\ \kappa_{|\mathscr{S}|}, & \text{If sample } \mathscr{S} \text{is drawn } \textit{without} \text{ replacement} \end{cases}, \tag{9c}$$

where κ_1 and $\kappa_{|\mathscr{S}|}$ are as in (9a).

2 Sub-sampling

We now study various sampling strategies for appropriately approximating the Hessian and the gradient. We note that all the following sampling results provide worst-case sample sizes which are useful in regimes where $n \gg 1$. More generally, however, the prescribed sample sizes should be mostly regarded as a qualitative guide to practice, as opposed to verbatim.

2.1 Sub-sampling Hessian

For the optimization problem (1), at each iteration, consider picking $\mathscr{S}$, uniformly at random *with* or *without* replacement. Let

$$H(\mathbf{x}) \triangleq \frac{1}{|\mathscr{S}|}\sum_{j\in\mathscr{S}} \nabla^2 f_j(\mathbf{x}), \tag{10}$$

be the sub-sampled Hessian. As mentioned before in Sect. 1.1, in order for such sub-sampling to be useful, we need the sample size $|\mathscr{S}|$ to satisfy (R.1). In addition, as mentioned in (R.2), we need to at least ensure that $H(\mathbf{x})$ is $\mathscr{K}$-restricted PD similar

to the original Hessian, e.g., for $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$, $H(\mathbf{x})$ should be uniformly PD. In this case, the direction given by $H(\mathbf{x})$, indeed, yields a descent direction with respect to $\mathscr{K}$. It is important to emphasize that, ideally, we'd like to ensure such $\mathscr{K}$-restricted PD property *without* any additional regularization to the Hessian, e.g., Levenberg–Marquardt type. This is because such added regularization perturbs the spectrum corresponding to small eigenvalues of the Hessian, which in turn destroys the curvature information. Recall that the most "informative" part of the curvature information is contained in the spectrum corresponding to small eigenvalues. Levenberg–Marquardt type regularization, for example, easily diminishes the contributions of the directions corresponding to small eigenvalues. This could make the algorithm behave more like gradient descent, which defeats the purpose of using second order information! Lemma 1 shows that we can indeed probabilistically guarantee such PD property.

Lemma 1 ($\mathscr{K}$-restricted positive definiteness) *Given any* $0 < \varepsilon, \delta < 1$, *and* $\mathbf{x} \in \mathscr{D} \cap \mathscr{C}$, *if* $|\mathscr{S}| \geq 2\kappa_1 \ln(p/\delta)/\varepsilon^2$, *then for* $H(\mathbf{x})$ *defined in (10), we have*

$$\mathbf{Pr}\big((1-\varepsilon)\gamma \leq \lambda_{\min}^{\mathscr{K}}(H(\mathbf{x}))\big) \geq 1-\delta,$$

where γ *and* κ_1 *are defined in* (6b) *and* (9a), *respectively.*

Proof Let U be an orthonormal basis for the subspace $\mathscr{K}$ in (3). For $\mathbf{x} \in \mathscr{D} \cap \mathscr{C}$, consider $|\mathscr{S}|$ random matrices $X_j(\mathbf{x})$, $j = 1, 2, \ldots, |\mathscr{S}|$ such that $\mathbf{Pr}(X_j(\mathbf{x}) = \nabla^2 f_i(\mathbf{x})) = 1/n$; $\forall i \in [n]$. Define $H(\mathbf{x}) \triangleq \sum_{j\in\mathscr{S}} X_j(\mathbf{x})/|\mathscr{S}|$ and also $X(\mathbf{x}) \triangleq \sum_{j\in\mathscr{S}} U^T X_j(\mathbf{x}) U = |\mathscr{S}| U^T H(\mathbf{x}) U$. Note that we have $\mathbb{E}(X_j(\mathbf{x})) = \nabla^2 F(\mathbf{x})$, $X_j(\mathbf{x}) \succeq_{\mathscr{K}} 0$, $\lambda_{\max}^{\mathscr{K}}(X_j(\mathbf{x})) \leq \widehat{K}_1$, and $\lambda_{\min}(\sum_{j\in\mathscr{S}} \mathbb{E}(U^T X_j(\mathbf{x}) U)) = |\mathscr{S}| \lambda_{\min}^{\mathscr{K}}(\nabla^2 F(\mathbf{x})) \geq |\mathscr{S}|\gamma$, where γ and $\widehat{K}_1$ are defined in (6b) and (9b), respectively. Noting that $\lambda_{\min}(X(\mathbf{x})) = |\mathscr{S}| \lambda_{\min}^{\mathscr{K}}(H(\mathbf{x}))$, Matrix Chernoff bound [39, Theorem 1.1] or [38, Theorem 2.2] for sampling with or without replacement, respectively, gives $\mathbf{Pr}(\lambda_{\min}(X(\mathbf{x})) \leq (1-\varepsilon)|\mathscr{S}| \lambda_{\min}^{\mathscr{K}}(\nabla^2 F(\mathbf{x}))) \leq p\big[e^{-\varepsilon}(1-\varepsilon)^{(\varepsilon-1)}\big]^{|\mathscr{S}|\gamma/\widehat{K}_1}$. The result follows by noticing that $e^{-\varepsilon}(1-\varepsilon)^{(\varepsilon-1)} \leq e^{-\varepsilon^2/2}$, and requiring that $\exp\{-\varepsilon^2|\mathscr{S}|/(2\kappa_1)\} \leq \delta$. □

If, instead, we require that the sub-sampled Hessian preserves the spectrum of the full Hessian, we will need larger sample than that of Lemma 1.

Lemma 2 ($\mathscr{K}$-restricted spectrum preserving) *Given any* $0 < \varepsilon, \delta < 1$ *and* $\mathbf{x} \in \mathscr{D} \cap \mathscr{C}$, *if* $|\mathscr{S}| \geq 16\kappa_1^2 \ln(2p/\delta)/\varepsilon^2$, *then for* $H(\mathbf{x})$ *defined in (10), we have*

$$\mathbf{Pr}\big(\|H(\mathbf{x}) - \nabla^2 F(\mathbf{x})\|_{\mathscr{K}} \leq \varepsilon\gamma\big) \geq 1-\delta,$$

where γ *and* κ_1 *are defined in* (6b) *and* (9a), *respectively.*

Proof Let U be an orthonormal basis for the subspace $\mathscr{K}$ in (3). Consider $|\mathscr{S}|$ random matrices $X_j(\mathbf{x})$, $j = 1, 2, \ldots, |\mathscr{S}|$ as in the proof of Lemma 1. Define $Y_j(\mathbf{x}) \triangleq U^T\big(X_j(\mathbf{x}) - \nabla^2 F(\mathbf{x})\big)U$ and $Y(\mathbf{x}) \triangleq \sum_{j\in\mathscr{S}} Y_j(\mathbf{x}) = |\mathscr{S}| U^T\big(H(\mathbf{x}) - \nabla^2 F(\mathbf{x})\big)U$, where $H(\mathbf{x}) \triangleq \sum_{j\in\mathscr{S}} X_j(\mathbf{x})/|\mathscr{S}|$. Note that $\mathbb{E}(Y_j(\mathbf{x})) = 0$ and for $X_j(\mathbf{x}) = \nabla^2 f_1(\mathbf{x})$,

$\|Y_j^2(\mathbf{x})\| = \|Y_j(\mathbf{x})\|^2 = \|U^T(\frac{n-1}{n}\nabla^2 f_1(\mathbf{x}) - \sum_{i=2}^n \frac{1}{n}\nabla^2 f_i(\mathbf{x}))U\|^2 \leq 4(\frac{n-1}{n})^2\widehat{K}_1^2 \leq 4\widehat{K}_1^2$, where $\widehat{K}_1$ is defined in (9b). Operator-Bernstein inequality [21, Theorem 1], for both sampling with and without replacement, gives $\mathbf{Pr}(\|H(\mathbf{x}) - \nabla^2 F(\mathbf{x})\|_{\mathscr{K}} \geq \varepsilon\gamma) = \mathbf{Pr}(\|Y(\mathbf{x})\| \geq \varepsilon|\mathscr{S}|\gamma) \leq 2p\exp\{-\varepsilon^2|\mathscr{S}|\gamma^2/(16K^2)\}$. The requirement on $|\mathscr{S}|$ gives the result. □

As a consequence of Lemma 2, we have the spectral approximation property of the sub-sampled matrix $H(\mathbf{x})$, with respect to the cone $\mathscr{K}$ i.e., $\mathbf{Pr}\big((1-\varepsilon)\nabla^2 F(\mathbf{x}) \preceq_{\mathscr{K}} H(\mathbf{x}) \preceq_{\mathscr{K}} (1+\varepsilon)\nabla^2 F(\mathbf{x})\big) \geq 1-\delta$, where $A \preceq_{\mathscr{K}} B$ is defined in Sect. 1.4. This follows from $\|H - \nabla^2 F(\mathbf{x})\|_{\mathscr{K}} \leq \varepsilon\gamma$, which gives $(1-\varepsilon)U^T\nabla^2 F(\mathbf{x})U \preceq U^T\nabla^2 F(\mathbf{x})U - \varepsilon\gamma\mathbb{I} \preceq U^T H(\mathbf{x})U \preceq U^T\nabla^2 F(\mathbf{x})U + \varepsilon\gamma\mathbb{I} \preceq (1+\varepsilon)U^T\nabla^2 F(\mathbf{x})U$. This ensures that the sub-sampled Hessian, to ε accuracy, preserves the spectrum of the full Hessian.

In Lemma 1, the sufficient sample size, $|\mathscr{S}|$, grows only *linearly* in κ_1, i.e., $\Omega(\kappa_1)$, as opposed to *quadratically*, i.e., $\Omega(\kappa_1^2)$, in Lemma 2. This difference, in fact, boils down to the difference between the requirements for global and local convergence in (R.2). For example, for $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$, in order to guarantee global convergence, we only require that the sub-sampled Hessian is uniformly PD. In contrast, to obtain fast local convergence, we need a much stronger guarantee to preserve the spectrum of the true Hessian. Consequently, Lemma 1 requires a smaller sample size, i.e., in the order of κ_1 versus κ_1^2 for Lemma 2, while delivering a much weaker guarantee.

Depending on κ_1 and for $n \gg 1$, in Lemmas 1 and 2, we can have $|\mathscr{S}| \ll n$. However, since both bounds on $|\mathscr{S}|$ in Lemmas 1 and 2 are too conservative, the required sample size can be unnecessarily large. Unfortunately, for uniform sampling, this is unavoidable as the prescribed sample size protects against the worst-case scenario of extreme non-uniformity among $\nabla^2 f_i(\mathbf{x})$'s [12,25,42]. In some extreme cases, uniform sampling might even require $\Omega(n)$ samples to capture the Hessian appropriately. In such cases, if possible, non-uniform sampling schemes result in sample sizes that are independent of n and are resilient to such non-uniformity [42].

2.2 Sub-sampling gradients

For sub-sampling the gradient, consider picking the indices in $\mathscr{S}$ uniformly at random with replacement, and let

$$\mathbf{g}(\mathbf{x}) \triangleq \frac{1}{|\mathscr{S}|}\sum_{j\in\mathscr{S}} \nabla f_j(\mathbf{x}), \tag{11}$$

be the sub-sampled gradient. By (R.2), we require that the sub-sampled gradient contains as much of the first order information from the full gradient as possible. For this, we write the gradient $\nabla F(\mathbf{x})$ in a *matrix-matrix product* from as $\nabla F(\mathbf{x}) = AB$ where

$$A \triangleq \big(\nabla f_1(\mathbf{x})\ \nabla f_2(\mathbf{x}) \cdots \nabla f_n(\mathbf{x})\big) \in \mathbb{R}^{p\times n}\ ,\ B \triangleq \big(1/n\ 1/n\ \ldots\ 1/n\big)^T \in \mathbb{R}^{n\times 1}. \tag{12}$$

Table 2 Uniform estimates for $G(\mathbf{x})$ in GLMs, $F(\mathbf{x}) = n^{-1}\sum_{i=1}^{n}\left(\Phi(\mathbf{a}_i^T\mathbf{x}) - b_i\mathbf{a}_i^T\mathbf{x}\right)$, over the sparsity inducing constraint set $\mathscr{C} = \{\mathbf{x} \in \mathbb{R}^p;\ \|\mathbf{x}\|_1 \le 1\}$. The infinity norm of a vector $\mathbf{a}$ is denoted by $\|\mathbf{a}\|_\infty$

	Least Squares	Logistic	Poisson
$\Phi(t)$	$t^2/2$	$\ln(1+\exp(t))$	$\exp(t)$
$\nabla f_i(\mathbf{x})$	$\left(\mathbf{a}_i^T\mathbf{x} - b_i\right)\mathbf{a}_i$	$\left(1/\left(1+e^{-\mathbf{a}_i^T\mathbf{x}}\right) - b_i\right)\mathbf{a}_i$	$\left(e^{\mathbf{a}_i^T\mathbf{x}} - b_i\right)\mathbf{a}_i$
$\sup_{\mathbf{x}\in\mathscr{C}} G(\mathbf{x})$	$\max_{i\in[n]}\left(\|\mathbf{a}_i\|_\infty + \|b_i\|\right)\|\mathbf{a}_i\|$	$\max_{i\in[n]}\left(1/\left(1+e^{-\|\mathbf{a}_i\|_\infty}\right) + \|b_i\|\right)\|\mathbf{a}_i\|$	$\max_{i\in[n]}\left(e^{\|\mathbf{a}_i\|_\infty} + \|b_i\|\right)\|\mathbf{a}_i\|$

We then use approximate matrix multiplication results from RandNLA [15,28], to probabilistically control the error in the approximation of $\nabla F(\mathbf{x})$ by $\mathbf{g}(\mathbf{x})$, through uniform sampling of the columns and rows A, B.

Lemma 3 (Gradient sub-sampling) *For a given* $\mathbf{x} \in \mathscr{D}\cap\mathscr{C}$, *let* $\|\nabla f_i(\mathbf{x})\|_{\mathscr{K}} \le G(\mathbf{x}) < \infty,\ i = 1, 2, \ldots, n$. *For any* $0 < \varepsilon, \delta < 1$, *if* $|\mathscr{S}| \ge G(\mathbf{x})^2\left(1 + \sqrt{8\ln(1/\delta)}\right)^2/\varepsilon^2$, *then for* $\mathbf{g}(\mathbf{x})$ *as in* (11), *we have*

$$\mathbf{Pr}\left(\|\nabla F(\mathbf{x}) - \mathbf{g}(\mathbf{x})\|_{\mathscr{K}} \le \varepsilon\right) \ge 1 - \delta,$$

where $\|.\|_{\mathscr{K}}$ *is defined as in* (4).

Proof Let U be an orthonormal basis for $\mathscr{K}$. By the assumption and the definition (4), we have that at a given $\mathbf{x} \in \mathscr{D}\cap\mathscr{C}$, $\|U^T\nabla f_i(\mathbf{x})\| \le G(\mathbf{x}),\quad i = 1, 2, \ldots, n$. Now, approximating the gradient using sub-sampling is equivalent to approximating the product AB in (12) by sampling columns and rows of A and B, respectively, and forming matrices $\widehat{A}$ and $\widehat{B}$ such $\widehat{A}\widehat{B} \approx AB$. More precisely, for a random sampling index set $\mathscr{S}$, we can represent the sub-sampled gradient (11), by the product $\widehat{A}\widehat{B}$ where $\widehat{A} \in \mathbb{R}^{p\times|\mathscr{S}|}$ and $\widehat{B} \in \mathbb{R}^{|\mathscr{S}|\times 1}$ are formed by selecting uniformly at random and with replacement, $|\mathscr{S}|$ columns and rows of A and B, respectively, rescaled by $\sqrt{n/|\mathscr{S}|}$. By the assumption on $G(\mathbf{x})$, we can use [15, Lemma 11] to get, with probability $1-\delta$, $\|U^TAB - U^T\widehat{A}\widehat{B}\|_F = \|\nabla F(\mathbf{x}) - \mathbf{g}(\mathbf{x})\|_{\mathscr{K}} \le G(\mathbf{x})(1 + \sqrt{8\ln(1/\delta)})/\sqrt{|\mathscr{S}|}$. The result follows by requiring $G(\mathbf{x})(1 + \sqrt{8\ln(1/\delta)}) \le \varepsilon\sqrt{|\mathscr{S}|}$. □

Note that the gradient sampling in Lemma 3 is done with replacement; for gradient sampling without replacement see [5,8,19]. Further, the sample size from Lemma 3 is given with respect to the current iterate, $\mathbf{x}^{(k)}$. As a result, we need to be able to efficiently estimate $G(\mathbf{x}^{(k)})$ at every iteration or, a priori, have a uniform upper bound for $G(\mathbf{x})$ for all $\mathbf{x} \in \mathscr{D}\cap\mathscr{C}$. Fortunately, in many different problems, it is possible to efficiently estimate $G(\mathbf{x})$. For example, consider GLM objective function $F(\mathbf{x}) = n^{-1}\sum_{i=1}^{n}\left(\Phi(\mathbf{a}_i^T\mathbf{x}) - b_i\mathbf{a}_i^T\mathbf{x}\right)$, over a sparsity inducing constraint set, e.g., $\mathscr{C} = \{\mathbf{x} \in \mathbb{R}^p;\ \|\mathbf{x}\|_1 \le 1\}$. Here, $(\mathbf{a}_i, b_i),\ i = 1, 2, \ldots, n$, form response and covariate pairs where $\mathbf{a}_i \in \mathbb{R}^p$, and the domain of b_i depends on the GLM. The cumulant generating function, Φ, determines the type of GLM; see the book [30] for further details and applications. For illustration purposes only, Table 2 gives some very rough uniform estimates of the constant $G(\mathbf{x})$ with respect to $\mathscr{C}$ for some popular GLMs.

3 Convergence results

We will now leverage the sampling strategies described in Sect. 2 to study the convergence properties of SSN. We consider the global convergence behaviour in Sect. 3.1, followed by the local convergence properties in Sect. 3.2. We then combine these to give unifying results in Sect. 3.3. Finally, worst-case computational complexities involving various parts of these algorithms are gathered in Sect. 3.4.

The following results rely on the high-probability occurrence of the events $\{(1-\varepsilon)\gamma \leq \lambda_{\min}^{\mathscr{K}}(H(\mathbf{x}))\}$, $\{\|H(\mathbf{x}) - \nabla^2 F(\mathbf{x})\|_{\mathscr{K}} \leq \varepsilon\gamma\}$ and/or $\{\|\nabla F(\mathbf{x}) - \mathbf{g}(\mathbf{x})\|_{\mathscr{K}} \leq \varepsilon\}$ from Lemmas 1, 2, and 3, for one or several iterations. In what follows, the convergence probabilities, which can be controlled a priori, are explicitly given in all theorem statements. However, for simplicity, the respective proofs are given by implicitly conditioning on the occurrence of the corresponding events.

3.1 Global convergence

In this section, we study the global convergence of SSN using Armijo line search and only in the unconstrained case where $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$. Our choice in considering unconstrained optimization lies in the unfortunate fact that defining "inexactness" for the solution of constrained variant of (2a) in a *computationally feasible* way is non-trivial, if possible at all. In unconstrained case, the solution to (2a) boils down to a linear system [cf. (13a) and (15a)], where the notion of inexactness naturally arises.

Similar algorithms with asymptotic convergence guarantees are given in the pioneering work of [7], while [5] gives quantitative convergence rates in expectation. However, for both sets of these results, it is assumed that each f_i in (1) is strongly convex. Using Lemma 1, we study such globally-convergent algorithms under a milder assumption (6b), where strong convexity is only assumed for F, while each f_i is allowed to be only (weakly) convex as in (6a). Many optimization problems can be of this form, e.g., $f_i(\mathbf{x}) = g(\mathbf{a}_i^T\mathbf{x})$, with $g : \mathbb{R} \to \mathbb{R}$, $\mathbf{a}_i \in \mathbb{R}^p$ and $\text{Range}(\{\mathbf{a}_i\}_{i=1}^n) = \mathbb{R}^p$, i.e., the matrix whose rows are formed by $\mathbf{a}_i$ is full column rank. If the real valued function $g(t)$ is strongly convex, then we have $\nabla^2 f_i(\mathbf{x}) = g''(\mathbf{a}_i^T\mathbf{x})\mathbf{a}_i\mathbf{a}_i^T$, which is clearly rank one and not positive definite, but $F(x)$ is indeed strongly convex. A simple example is when $g(t) = t^2$ which gives rise to ordinary linear least squares.

3.1.1 Global convergence: sub-sampled Hessian and full gradient

In this section, we consider iterations (2) using the sub-sampled Hessian, $H(\mathbf{x}^{(k)})$ and the full gradient, $\nabla F(\mathbf{x}^{(k)})$. We first present an iterative algorithm in which, at every iteration, the linear system in (2a) is solved exactly, followed by an inexact variant.

Exact update In the unconstrained case where $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$, the iterations (2) using Armijo-type line-search to select α_k, can be re-written as $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k\mathbf{p}_k$, where

$$\mathbf{p}_k = -[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)}), \tag{13a}$$

and α_k is the largest $\alpha \leq 1$ such that

$$F(\mathbf{x}^{(k)} + \alpha \mathbf{p}_k) \leq F(\mathbf{x}^{(k)}) + \alpha\beta \mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}), \tag{13b}$$

for some $\beta \in (0, 1)$. Recall that (13b) can be approximately solved using various methods such as backtracking line search [6].

Algorithm 1 Globally Convergent SSN with Hessian Sub-Sampling

1: **Input:** $\mathbf{x}^{(0)}$, $0 < \delta < 1$, $0 < \varepsilon < 1$, $0 < \beta < 1$
2: - Set the sample size, $|\mathscr{S}|$, with ε and δ as in Lemma 1
3: **for** $k = 0, 1, 2, \cdots$ until termination **do**
4: - Select a sample set, $\mathscr{S}$, of size $|\mathscr{S}|$ and form $H(\mathbf{x}^{(k)})$ as in (10)
5: - Update $\mathbf{x}^{(k+1)}$ as in (13) with $H(\mathbf{x}^{(k)})$
6: **end for**

Theorem 1 (Global convergence of Algorithm 1) *Let Assumptions (6) hold. Using Algorithm 1 with any* $\mathbf{x}^{(k)} \in \mathbb{R}^p$*, with probability* $1 - \delta$*, we have*

$$F(\mathbf{x}^{(k+1)}) - F(\mathbf{x}^\star) \leq (1 - \rho_k)\big(F(\mathbf{x}^{(k)}) - F(\mathbf{x}^\star)\big), \tag{14}$$

where $\rho_k = 2\alpha_k\beta/\tilde{\kappa}$*, and* $\tilde{\kappa}$ *is defined as in (9c). Moreover, the step size is at least* $\alpha_k \geq 2(1-\beta)(1-\varepsilon)/\kappa$*, where* κ *is defined as in (8).*

Proof Since $\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k \geq \lambda_{\min}(H(\mathbf{x}))\|\mathbf{p}\|^2$, by Lemma 1, we have $\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k \geq (1-\varepsilon)\gamma\|\mathbf{p}_k\|^2$. By $\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) = -\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k$, this implies that $\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) < 0$ and we can indeed obtain decrease in the objective function. Now, it suffices to show that there exists an *iteration-independent* $\widetilde{\alpha} > 0$, such that (13b) holds for any $0 \leq \alpha \leq \widetilde{\alpha}$. For any $0 \leq \alpha$, define $\mathbf{x}_\alpha = \mathbf{x}^{(k)} + \alpha\mathbf{p}_k$. Assumption (6b) implies $F(\mathbf{x}_\alpha) - F(\mathbf{x}^{(k)}) \leq (\mathbf{x}_\alpha - \mathbf{x}^{(k)})^T \nabla F(\mathbf{x}^{(k)}) + K\|\mathbf{x}_\alpha - \mathbf{x}^{(k)}\|^2/2 = \alpha\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) + \alpha^2 K\|\mathbf{p}_k\|^2/2$. Now in order to pass the Armijo rule, we search for α such that $2\alpha\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) + \alpha^2 K\|\mathbf{p}_k\|^2 \leq 2\alpha\beta\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)})$, which gives $\alpha K\|\mathbf{p}_k\|^2 \leq -2(1-\beta)\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)})$. This latter inequality is satisfied if we require that $\alpha K\|\mathbf{p}_k\|^2 \leq 2(1-\beta)\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k$. As a result, having $\alpha \leq 2(1-\beta)(1-\varepsilon)/\kappa$, satisfies the Armijo rule. So in particular, we can always find an iteration-independent $\widetilde{\alpha} = 2(1-\beta)(1-\varepsilon)/\kappa$ such that (13b) holds for all $\alpha \leq \widetilde{\alpha}$. Now, for sampling *without* replacement and by $H(\mathbf{x}^{(k)})\mathbf{p}_k = -\nabla F(\mathbf{x}^{(k)})$ we get $\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k = \nabla F(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \nabla F(\mathbf{x}^{(k)}) \geq \|\nabla F(\mathbf{x}^{(k)})\|^2/\widehat{K}_{|\mathscr{S}|}$. Similarly for sampling *with* replacement, we have $\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k \geq 1/\widehat{K}_1\|\nabla F(\mathbf{x}^{(k)})\|^2$. By $\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) = -\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k$, (13b) gives $F(\mathbf{x}^{(k+1)}) \leq F(\mathbf{x}^{(k)}) - \alpha_k\beta\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k$. The result follows by subtracting $F(\mathbf{x}^\star)$ from both sides and noting that Assumption (6b) gives $F(\mathbf{x}^{(k)}) - F(\mathbf{x}^\star) \leq \|\nabla F(\mathbf{x}^{(k)})\|^2/(2\gamma)$; see [32, Theorem 2.1.10]. □

The role of ε in Theorem 1, which appears explicitly in the worst-case step-size, is rather interesting. As it can be seen, the better we approximate the Hessian, the

larger our "bottom-line" step-size would be, which in turn implies a faster worst-case convergence speed. In fact, this is rather intuitive: inaccurate estimation of the curvature implies that the local quadratic approximation of F at $\mathbf{x}^{(k)}$ in (2a) is unreliable, i.e, the local approximation error between the true function and the second-order model increases. Hence, the resulting method would tend to take more conservative, i.e., shorter, steps, to account for such increased local inaccuracy.

Theorem 1 states that the iterates generated by Algorithm 1, with high-probability, approach $\mathbf{x}^\star$, starting from any $\mathbf{x}^{(0)}$. However, in the worst case, the global linear rate is with $\rho_k \in \Omega(1/(\kappa\tilde{\kappa}))$ which is seemingly unsatisfying, and indeed worse than that of simple gradient descent. This is due to the application of Armijo line search and appears as a by-product of the analysis. In fact, such unsatisfying global rate appears throughout the literature for similar Newton-type methods, e.g. Theorem 2.1 in [5], as well as in some classical and widely cited textbooks such as Sect. 9.3.5 in [6]. This has indeed been pointed out in [33] where a cubic regularization is employed to circumvent this issue. Examples have also been constructed for which, in the worst case, steepest-descent and Newton's method are equally slow for unconstrained optimization [10,11]. Despite these worst-case scenarios, efficient variants of Newton-type methods have been shown to outperform first order alternatives in many practical settings, e.g., see the numerical examples in [2,5,16,27,41,42].

Inexact update In high dimensional settings where $p \gg 1$, finding the exact update, $\mathbf{p}_k$, in (13a) is computationally expensive. Hence, it is imperative to be able to calculate the update direction only approximately. Such inexact updates have been used in many second-order optimization algorithms, e.g. [5,9,14,26].

Our results are inspired by [9]. More specifically, instead of (13a), we approximately solve the underlying linear system such that for some $0 \leq \theta_1, \theta_2 < 1$, we have

$$\|H(\mathbf{x}^{(k)})\mathbf{p}_k + \nabla F(\mathbf{x}^{(k)})\| \leq \theta_1 \|\nabla F(\mathbf{x}^{(k)})\|, \tag{15a}$$

$$\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) \leq -(1-\theta_2)\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k. \tag{15b}$$

The condition (15a) is the usual relative residual of the approximate solution, while condition (15b) ensures that such a $\mathbf{p}_k$ is always a descent direction. Note that given any $0 \leq \theta_1, \theta_2 < 1$, one can always find a $\mathbf{p}_k$ satisfying (15), e.g., the exact solution.

Assume that to find $\mathbf{p}_k$ in (15), the linear system $H(\mathbf{x}^{(k)})\mathbf{p}_k^\star = -\nabla F(\mathbf{x}^{(k)})$ is solved approximately using an iterative method, in which the iterates, $\mathbf{p}_k^{(t)}$, are generated by successive minimization of the function $g_k(\mathbf{p}) \triangleq \mathbf{p}^T \nabla F(\mathbf{x}^{(k)}) + \mathbf{p}^T H(\mathbf{x}^{(k)})\mathbf{p}/2$ over progressively expanding linear manifolds $\mathscr{M}_t$, i.e., $\mathbf{p}_k^{(t)} = \arg\min_{\mathbf{p}\in\mathscr{M}_t} g_k(\mathbf{p})$. One such method, which is suitable for our setting here, is the celebrated conjugate gradient (CG). Now since over any such space $\mathscr{M}_t$, $g_k(\mathbf{0}) = 0$, we always have $g_k(\mathbf{p}_k^{(t)}) \leq 0$, $\forall t$. As a result, for $\theta_2 = 1/2$, any such $\mathbf{p}_k^{(t)}$ always satisfies (15b). Although for $\theta_2 = 1/2$, (15b) can be simply dropped from (15), in the following results, we will treat θ_2 as a hyper-parameter to discuss its effect on convergence.

Recall that by Lemma 1, with high probability, we have $(1-\varepsilon)\gamma \leq \lambda_{\min}(H(\mathbf{x}^{(k)}))$; hence $\text{Cond}(H(\mathbf{x}^{(k)})) \leq \tilde{\kappa}/(1-\varepsilon)$ where $\tilde{\kappa}$ is as in (9c). If CG is used to obtain a solution for (15a), then by its worst case convergence [24], we get

$$\|\mathbf{p}_k^{(t)} - \mathbf{p}_k^\star\|_{H(\mathbf{x}^{(k)})} \leq 2\left(\frac{\sqrt{\tilde{\kappa}/(1-\varepsilon)}-1}{\sqrt{\tilde{\kappa}/(1-\varepsilon)}+1}\right)^t \|\mathbf{p}^{(0)} - \mathbf{p}_k^\star\|_{H(\mathbf{x}^{(k)})},$$

where $\|\mathbf{p}\|_A = \sqrt{\mathbf{p}^T A\mathbf{p}}$. If $\mathbf{p}^{(0)} = \mathbf{0}$, it follows that

$$\|H(\mathbf{x}^{(k)})\mathbf{p}_k^{(t)} + \nabla F(\mathbf{x}^{(k)})\| \leq 2\sqrt{\frac{\tilde{\kappa}}{(1-\varepsilon)}}\left(\frac{\sqrt{\tilde{\kappa}/(1-\varepsilon)}-1}{\sqrt{\tilde{\kappa}/(1-\varepsilon)}+1}\right)^t \|\nabla F(\mathbf{x}^{(k)})\|.$$

Hence, after

$$t \geq \ln\left(\frac{2}{\theta_1}\sqrt{\frac{\tilde{\kappa}}{(1-\varepsilon)}}\right) / \ln\left(\frac{\sqrt{\tilde{\kappa}/(1-\varepsilon)}+1}{\sqrt{\tilde{\kappa}/(1-\varepsilon)}-1}\right), \tag{16}$$

iterations, we get (15a). Theorem 2 prescribes a sufficient condition on θ_1, which is less strict than similar prior works, and yet ensures a desirable convergence property.

As shown above, for $\theta_2 = 1/2$ and any $\theta_1 < 1$, the complexity of checking (15) is directly related to the computational cost involved in solving the underlying linear system. For $\theta_2 < 1/2$, however, we are not aware of a way to quantify the cost required to ensure (15b). Overall computational complexities of Algorithms involving (15) are discussed in Sect. 3.4.

Theorem 2 (Global convergence of Algorithm 1*:* inexact update) *Let Assumptions* (6) *hold, and* $0 \leq \theta_1, \theta_2 < 1$ *be given. Using Algorithm* 1 *with any* $\mathbf{x}^{(k)} \in \mathbb{R}^p$*, and the "inexact" update* (15) *instead of* (13a)*, with probability* $1-\delta$*, we have* (14) *with* ρ_k *as follows: (i) if*

$$\theta_1 \leq \sqrt{(1-\varepsilon)/(4\tilde{\kappa})},$$

then $\rho_k = \alpha_k\beta/\tilde{\kappa}$*, (ii) otherwise,* $\rho_k = 2(1-\theta_2)(1-\theta_1)^2(1-\varepsilon)\alpha_k\beta/\tilde{\kappa}^2$*, with* $\tilde{\kappa}$*,* θ_1 *and* θ_2 *as in* (9c) *for* (15a)*, and* (15b)*, respectively. Moreover, for both cases, the step size is* $\alpha_k \geq 2(1-\theta_2)(1-\beta)(1-\varepsilon)/\kappa$*, where* κ *is as in* (8)*.*

Proof We give the proof for sampling without replacement. The proof for sampling with replacement is obtained similarly. First, we note that Lemma 1 and (15b) imply

$$\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) \leq -(1-\theta_2)(1-\varepsilon)\gamma\|\mathbf{p}_k\|^2. \tag{17}$$

So, $\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) < 0$ and we can indeed obtain decrease in the objective function. As in the proof of Theorem (1), we get $F(\mathbf{x}_\alpha) - F(\mathbf{x}^{(k)}) \leq \alpha\mathbf{p}_k^T\nabla F(\mathbf{x}^{(k)}) + \alpha^2 K\|\mathbf{p}_k\|^2/2$. In order to pass the Armijo rule, we search for α such that $\alpha K\|\mathbf{p}_k\|^2 \leq -2(1-\beta)\mathbf{p}_k^T\nabla F(\mathbf{x}^{(k)})$. Hence, $\alpha \leq 2(1-\theta_2)(1-\beta)(1-\varepsilon)/\kappa$, satisfies the Armijo rule.

For part (i), we notice that by self-duality of the vector ℓ_2 norm, i.e., $\|\mathbf{v}\|_2 = \sup\{\mathbf{w}^T\mathbf{v};\ \|\mathbf{w}\|_2 = 1\}$, (15a) implies $\mathbf{p}_k^T\nabla F(\mathbf{x}^{(k)}) + \nabla F(\mathbf{x}^{(k)})^T[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)})$

$\leq \theta_1 \|\nabla F(\mathbf{x}^{(k)})\| \|[H(\mathbf{x}^{(k)})]^{-1} \nabla F(\mathbf{x}^{(k)})\|$. From Lemma 1 we have that $[H(\mathbf{x}^{(k)})]^{-1} \preceq 1/((1-\varepsilon)\gamma)\mathbb{I}$, which, in turn, gives

$$\|[H(\mathbf{x}^{(k)})]^{-1} \nabla F(\mathbf{x}^{(k)})\| \leq \sqrt{\nabla F(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \nabla F(\mathbf{x}^{(k)})/((1-\varepsilon)\gamma)}.$$

Hence,

$$\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) \leq q\big(\theta_1 \|\nabla F(\mathbf{x}^{(k)})\|/\sqrt{(1-\varepsilon)\gamma} - q\big),$$

where $q \triangleq \sqrt{\nabla F(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \nabla F(\mathbf{x}^{(k)})}$. Since $q \geq \|\nabla F(\mathbf{x}^{(k)})\|/\sqrt{\widehat{K}_{|\mathscr{S}|}}$, if $\theta_1 \leq \sqrt{(1-\varepsilon)/(4\tilde{\kappa})}$, then we get $\theta_1 \|\nabla F(\mathbf{x}^{(k)})\| \leq q\sqrt{(1-\varepsilon)\gamma}/2$.

For part (ii), we have $\theta_1 \|\nabla F(\mathbf{x}^{(k)})\| \geq \|H(\mathbf{x}^{(k)})\mathbf{p}_k + \nabla F(\mathbf{x}^{(k)})\| \geq \|\nabla F(\mathbf{x}^{(k)})\| - \|H(\mathbf{x}^{(k)})\mathbf{p}_k\|$, which gives $(1-\theta_1)\|\nabla F(\mathbf{x}^{(k)})\| \leq \|H(\mathbf{x}^{(k)})\mathbf{p}_k\| \leq \|H(\mathbf{x}^{(k)})\|\|\mathbf{p}_k\| \leq \widehat{K}_{|\mathscr{S}|}\|\mathbf{p}_k\|$. (17) implies $\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) \leq -(1-\theta_2)(1-\varepsilon)\gamma(1-\theta_1)^2 \|\nabla F(\mathbf{x}^{(k)})\|^2/\widehat{K}_{|\mathscr{S}|}^2$. By Assumption (6b), the results follow as in the end of the proof of Theorem 1. □

By Theorem 2, in order to guarantee similar worst-case global convergence rate as that with exact update in Theorem 1, i.e., $\rho_k \in \Omega(1/(\kappa\tilde{\kappa}))$, it is sufficient to solve the linear system to a "small-enough" accuracy, i.e., $\Omega(\sqrt{1/\tilde{\kappa}})$. This requirement on relative accuracy is less strict than with what is found in similar literature. Clearly, this improvement merely manifests itself as a constant in the worst-case analysis of the number of CG iterations. However, in practice, the difference between the actual amount of work by CG to achieve a relative residual of $\mathscr{O}(1/\tilde{\kappa})$ versus $\mathscr{O}(\sqrt{1/\tilde{\kappa}})$, given the non-monotonic behavior of CG in terms of residuals, can be significant. By Theorem 2, large ill-conditioning, i.e., $\tilde{\kappa} \gg 1$, or inaccurate Hessian estimation, i.e., large ε, both can necessitate small θ_1, and if the linear system is not solved accurately enough, then the convergence rate can degrade, i.e., $\rho_k \in \Omega(1/(\kappa\tilde{\kappa}^2))$.

From Theorem 2, we see that the minimum amount of decrease in the objective function is mainly dependent on θ_1, i.e., the accuracy of the linear system solve. On the other hand, the dependence of the step size, α_k, on θ_2 indicates that the algorithm can take larger steps along a search direction, $\mathbf{p}_k$, that points more accurately towards the direction of the largest rate of decrease.

3.1.2 Global convergence: sub-sampled Hessian and gradient

We now consider SSN-type algorithms which are fully stochastic, in which, both the gradient and the Hessian are approximated. As before, we first study the iterations with exact solution of (2a), and then turn to inexact variants.

Exact update For the sub-sampled gradient and the Hessian, $\mathbf{g}(\mathbf{x}^{(k)})$, $H(\mathbf{x}^{(k)})$, respectively, consider rewriting the update (2) for as $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{p}_k$, where

$$\mathbf{p}_k = -[H(\mathbf{x}^{(k)})]^{-1} \mathbf{g}(\mathbf{x}^{(k)}), \tag{18a}$$

and α_k is the largest $\alpha \leq 1$ such that, for some $\beta \in (0,1)$, we have

$$F(\mathbf{x}^{(k)} + \alpha \mathbf{p}_k) \leq F(\mathbf{x}^{(k)}) + \alpha\beta \mathbf{p}_k^T \mathbf{g}(\mathbf{x}^{(k)}). \tag{18b}$$

Algorithm 2 Globally Convergent SSN with Hessian and Gradient Sub-Sampling

Input: $\mathbf{x}^{(0)}$, $0 < \delta < 1$, $0 < \varepsilon_1 < 1$, $0 < \varepsilon_2 < 1$, $0 < \beta < 1$, $\sigma \geq 0$
- Set the sample size, $|\mathcal{S}_H|$, with ε_1 and δ as in Lemma 1
for $k = 0, 1, 2, \cdots$ until termination **do**
  - Select a sample set, $\mathcal{S}_H$, of size $|\mathcal{S}_H|$ and form $H(\mathbf{x}^{(k)})$ as in (10)
  - Set the sample size, $|\mathcal{S}_\mathbf{g}|$, with ε_2, δ and $\mathbf{x}^{(k)}$ as in Lemma 3
  - Select a sample set, $\mathcal{S}_\mathbf{g}$ of size $|\mathcal{S}_\mathbf{g}|$ and form $\mathbf{g}(\mathbf{x}^{(k)})$ as in (11)
  if $\|\mathbf{g}(\mathbf{x}^{(k)})\| < \sigma\varepsilon_2$ **then**
    - STOP
  end if
  - Update $\mathbf{x}^{(k+1)}$ as in (18) with $H(\mathbf{x}^{(k)})$ and $\mathbf{g}(\mathbf{x}^{(k)})$
end for

Theorem 3 (Global convergence of Algorithm 2) *Let Assumptions (6) hold. For any $\mathbf{x}^{(k)} \in \mathbb{R}^p$, using Algorithm 2 with $\varepsilon_1 \leq 1/2$ and $\sigma \geq 4\tilde{\kappa}/(1-\beta)$, we have the following with probability $(1-\delta)^2$: if "STOP", then*

$$\|\nabla F(\mathbf{x}^{(k)})\| < (1+\sigma)\,\varepsilon_2,$$

otherwise, (14) holds with $\rho_k = 8\alpha_k\beta/(9\tilde{\kappa})$ and $\alpha_k \geq (1-\beta)(1-\varepsilon_1)/\kappa$, where κ and $\tilde{\kappa}$ are defined in (8) and (9c), respectively.

Proof We give the proof for the sampling without replacement as sampling with replacement is obtained similarly. By Lemma 1 and (13a), we have $\mathbf{p}_k^T\mathbf{g}(\mathbf{x}^{(k)}) = -\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k \geq -(1-\varepsilon_1)\gamma\|\mathbf{p}_k\|^2$. Hence, since $\mathbf{p}_k^T\mathbf{g}(\mathbf{x}^{(k)}) < 0$, we can obtain decrease in the objective function. As in the proof of Theorem 1, we first need to show that there exists an iteration-independent step-size, $\widetilde{\alpha} > 0$, such that (18b) holds for any $0 \leq \alpha \leq \widetilde{\alpha}$. For any $0 \leq \alpha$, define $\mathbf{x}_\alpha = \mathbf{x}^{(k)} + \alpha\mathbf{p}_k$. By Assumption (6b), we have

$$\begin{aligned} F(\mathbf{x}_\alpha) - F(\mathbf{x}^{(k)}) &\leq \alpha\mathbf{p}_k^T\nabla F(\mathbf{x}^{(k)}) + \alpha^2\frac{K}{2}\|\mathbf{p}_k\|^2 \\ &\leq \alpha\mathbf{p}_k^T\mathbf{g}(\mathbf{x}^{(k)}) + \alpha\|\nabla F(\mathbf{x}^{(k)}) - \mathbf{g}(\mathbf{x}^{(k)})\|\|\mathbf{p}_k\| + \alpha^2\frac{K}{2}\|\mathbf{p}_k\|^2 \\ &\leq -\alpha\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k + \alpha\varepsilon_2\|\mathbf{p}_k\| + \alpha^2\frac{K}{2}\|\mathbf{p}_k\|^2. \end{aligned}$$

As a result, we need to find α such that $-\alpha\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k + \varepsilon_2\alpha\|\mathbf{p}_k\| + \alpha^2 K\|\mathbf{p}_k\|^2/2 \leq -\alpha\beta\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k$, which follows if $\varepsilon_2 + \alpha K\|\mathbf{p}_k\|/2 \leq (1-\beta)(1-\varepsilon_1)\gamma\|\mathbf{p}_k\|$. This latter inequality holds if $\alpha = (1-\beta)(1-\varepsilon_1)\gamma/K$ and $\varepsilon_2 = (1-\beta)(1-\varepsilon_1)\gamma\|\mathbf{p}_k\|/2$. From $H(\mathbf{x}^{(k)})\mathbf{p}_k = -\mathbf{g}(\mathbf{x}^{(k)})$, it is implied that to guarantee an iteration-independent lower bound for α as above, we need $\varepsilon_2 \leq (1-\beta)(1-\varepsilon_1)\gamma\|\mathbf{g}(\mathbf{x}^{(k)})\|/(2\widehat{K}_{|\mathcal{S}|})$, which, by the choice of σ and ε_1, is imposed by the algorithm. If the stopping criterion succeeds, then by $\|\mathbf{g}(\mathbf{x}^{(k)})\| \geq \|\nabla F(\mathbf{x}^{(k)})\| - \varepsilon_2$, we have $\|\nabla F(\mathbf{x}^{(k)})\| < (1+\sigma)\,\varepsilon_2$. Otherwise, by $\|\mathbf{g}(\mathbf{x}^{(k)})\| \leq \|\nabla F(\mathbf{x}^{(k)})\| + \varepsilon_2$, we get $(\sigma-1)\varepsilon_2 \leq \|\nabla F(\mathbf{x}^{(k)})\|$. Since $\sigma \geq 4$, the latter inequality implies that

$$2\|\nabla F(\mathbf{x}^{(k)})\|/3 \leq (\sigma-2)\|\nabla F(\mathbf{x}^{(k)})\|/(\sigma-1) \leq \|\nabla F(\mathbf{x}^{(k)})\| - \varepsilon_2.$$

From $H(\mathbf{x}^{(k)})\mathbf{p}_k = -\mathbf{g}(\mathbf{x}^{(k)})$, it follows that

$$\begin{aligned}\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k &= \mathbf{g}(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \mathbf{g}(\mathbf{x}^{(k)}) \geq \|\mathbf{g}(\mathbf{x}^{(k)})\|^2 / \widehat{K}_{|\mathcal{S}|} \\ &\geq (\|\nabla F(\mathbf{x}^{(k)})\| - \varepsilon_2)^2 / \widehat{K}_{|\mathcal{S}|} \geq 4\|\nabla F(\mathbf{x}^{(k)})\|^2 / (9\widehat{K}_{|\mathcal{S}|}).\end{aligned}$$

By Assumption (6b), the result follows as in the end of the proof of Theorem 1. □

Theorem 3 only guarantees *approximate optimality*, where $\|\nabla F(\mathbf{x}^{(k)})\|$ is small, and no iterate from Algorithm 2 is ensured to be exactly optimal, where $\|\nabla F(\mathbf{x}^{(k)})\|$ vanishes. However, by (6b), in order to obtain sub-optimality in objective function as $F(\mathbf{x}^{(k)}) - F(\mathbf{x}^\star) \leq \varsigma$ for some $\varsigma \leq 1$, it is sufficient to require that upon termination $\|\nabla F(\mathbf{x}^{(k)})\|^2 < 2\varsigma\gamma$, which, in turn, requires $\varepsilon_2 \leq \sqrt{2\varsigma\gamma}/(1+\sigma)$; further related complexities are given in Table 3. It has been also well-established, e.g., [5,8,19], that without increasing sample sizes for gradient estimation, one can, at best, hope for convergence to a neighborhood of the solution. This is indeed inline with Theorem 3; see also Theorems 11, 12, and 14.

Inexact update For some $0 \leq \theta_1, \theta_2 < 1$, consider the inexact version of (18a), as a solution of

$$\|H(\mathbf{x}^{(k)})\mathbf{p}_k + \mathbf{g}(\mathbf{x}^{(k)})\| \leq \theta_1 \|\mathbf{g}(\mathbf{x}^{(k)})\|, \tag{19a}$$

$$\mathbf{p}_k^T \mathbf{g}(\mathbf{x}^{(k)}) \leq -(1-\theta_2)\mathbf{p}_k^T H(\mathbf{x}^{(k)})\mathbf{p}_k. \tag{19b}$$

Theorem 4 gives the global convergence of Algorithm 2 with update (19). The proof is given by combining the arguments used to prove Theorems 2 and 3, and hence, is omitted here.

Theorem 4 (Global convergence of Algorithm 2*:* inexact update) *Let Assumptions* (6) *hold, and* $0 < \theta_1, \theta_2 < 1$ *be given. For any* $\mathbf{x}^{(k)} \in \mathbb{R}^p$*, using Algorithm 2 with* $\varepsilon_1 \leq 1/2$*, the "inexact" update* (19) *instead of* (18a)*, and* $\sigma \geq 4\tilde{\kappa}/\big((1-\theta_1)(1-\theta_2)(1-\beta)\big)$*, the following holds with probability* $(1-\delta)^2$*. If "STOP", then* $\|\nabla F(\mathbf{x}^{(k)})\| < (1+\sigma)\,\varepsilon_2$*. Otherwise* (14) *holds in which case if* $\theta_1 \leq \sqrt{(1-\varepsilon_1)/(4\tilde{\kappa})}$*, then* $\rho_k = 4\alpha_k\beta/9\tilde{\kappa}$*, else* $\rho_k = 8\alpha_k\beta(1-\theta_2)(1-\theta_1)^2(1-\varepsilon_1)/9\tilde{\kappa}^2$*, with* $\tilde{\kappa}$ *defined as in* (9c)*. Moreover, for both cases,* $\alpha_k \geq (1-\theta_2)(1-\beta)(1-\varepsilon_1)/\kappa$*, with* κ *as in* (8)*.*

Although Algorithm 2 employs sub-sampled gradients, the step-size α_k, at each iteration, is chosen using exact evaluations of F in (18b). For most problems, the cost of evaluating a gradient is of the same order as that of the corresponding function. In this light, gradient sub-sampling in Algorithm 2 might, in fact, not contribute much in reducing the overall computational costs. However, in many problems, this can be remedied. Indeed, from the proof of Theorems 3, and 4, it is clear that we can simply, but conservatively, replace (18b) with $\alpha \leq 2((\beta-1)\mathbf{p}_k^T \mathbf{g}(\mathbf{x}^{(k)}) - \varepsilon_2\|\mathbf{p}_k\|)/(K\|\mathbf{p}_k\|^2)$, and Theorems 3, and 4 would stay the same. In this case, at each iteration, the step size can be readily chosen, although potentially smaller than what we could obtain from (18b), without having to resort to any functional evaluations. However, this requires the knowledge of K, which might not be available for all problems. Fortunately, there are many important instances, in particular in machine learning, in which an estimate of K can easily be obtained, e.g., most popular generalized linear models (GLMs) such as ridge regression, logistic regression, and Poisson regression.

3.2 Local convergence

While the classical Newton's method has seen great practical success in many application areas, the theoretical appeal mainly lies in its local convergence behavior. The folklore notion that "Newton method converges quadratically" is, in fact, a statement about its local theoretical guarantee. Indeed, any method striving to be Newton-type must enjoy similar superior local convergence properties. In this section, we set out to do that by showing that, locally, variants of SSN with Newton's method's "natural" step size, i.e., $\alpha_k = 1$, can achieve linear or superlinear convergence rates. Moreover, we show that such rates are *problem-independent*, i.e., the rates are prescribed by the user and are not affected by the problem under consideration.

3.2.1 Local convergence: sub-sampled Hessian and full gradient

We now consider the framework (2) with $\alpha_k = 1$, where, for a given $\mathbf{x}^{(k)} \in \mathscr{D} \cap \mathscr{C}$, we set $\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)})$ and $H(\mathbf{x}^{(k)})$ is sub-sampled as in (10).

Exact update We first study the case where, at every iteration, (2a) is solved exactly. Lemma 4 which will be the foundation of our main results for this Section.

Lemma 4 (Structural Lemma A) *Let Assumption* (7) *hold. Also, for* $H(\mathbf{x}^{(k)})$ *as in* (10)*, assume that*

$$\mathbf{p}^T H(\mathbf{x}^{(k)})\mathbf{p} > 0, \quad \forall\, \mathbf{p} \in \mathscr{K}\backslash\{0\}. \tag{20}$$

For the update (2) *with* $\alpha_k = 1$, $\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)})$ *and* $H(\mathbf{x}^{(k)})$*, we have* $\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| \leq \rho_0 \|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2$*, where* $\xi \triangleq 0.5L/\lambda^{\mathscr{K}}_{\min}(H(\mathbf{x}^{(k)}))$*, and* $\rho_0 \triangleq \|H(\mathbf{x}^{(k)}) - \nabla^2 F(\mathbf{x}^{(k)})\|_{\mathscr{K}}/\lambda^{\mathscr{K}}_{\min}(H(\mathbf{x}^{(k)}))$.

Proof Define $\Delta_k \triangleq \mathbf{x}^{(k)} - \mathbf{x}^\star$. From (2), since $\alpha_k = 1$, we get $\mathbf{x}^{(k+1)} = \widehat{\mathbf{x}}^{(k)}$. By optimality of $\mathbf{x}^{(k+1)}$ in (2a), we have for any $\mathbf{x} \in \mathscr{D} \cap \mathscr{C}$, $(\mathbf{x} - \mathbf{x}^{(k+1)})^T \nabla F(\mathbf{x}^{(k)}) + (\mathbf{x} - \mathbf{x}^{(k+1)})^T H(\mathbf{x}^{(k)})(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}) \geq 0$. In particular, setting $\mathbf{x} = \mathbf{x}^\star$, and noting that $\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} = \Delta_{k+1} - \Delta_k$, we get $\Delta_{k+1}^T H(\mathbf{x}^{(k)})\Delta_{k+1} \leq \Delta_{k+1}^T H(\mathbf{x}^{(k)})\Delta_k - \Delta_{k+1}^T \nabla F(\mathbf{x}^{(k)})$. Optimality of $\mathbf{x}^\star$ gives $\nabla F(\mathbf{x}^\star)^T(\mathbf{x}^{(k+1)} - \mathbf{x}^\star) \geq 0$, which implies $\Delta_{k+1}^T H(\mathbf{x}^{(k)})\Delta_{k+1} \leq \Delta_{k+1}^T H(\mathbf{x}^{(k)})\Delta_k - \Delta_{k+1}^T \nabla F(\mathbf{x}^{(k)}) + \Delta_{k+1}^T \nabla F(\mathbf{x}^\star)$. Now, by the mean value theorem

$$\nabla F(\mathbf{x}^{(k)}) - \nabla F(\mathbf{x}^\star) = \left(\int_0^1 \nabla^2 F\big(\mathbf{x}^\star + t(\mathbf{x}^{(k)} - \mathbf{x}^\star)\big) dt \right) (\mathbf{x}^{(k)} - \mathbf{x}^\star),$$

we have

$$\begin{aligned}
\Delta_{k+1}^T H(\mathbf{x}^{(k)})\Delta_{k+1} &\leq \Delta_{k+1}^T H(\mathbf{x}^{(k)})\Delta_k - \Delta_{k+1}^T \left(\int_0^1 \nabla^2 F\big(\mathbf{x}^\star + t(\mathbf{x}^{(k)} - \mathbf{x}^\star)\big) dt \right) \Delta_k \\
&= \Delta_{k+1}^T H(\mathbf{x}^{(k)})\Delta_k - \Delta_{k+1}^T \nabla^2 F(\mathbf{x}^{(k)})\Delta_k \\
&\quad + \Delta_{k+1}^T \nabla^2 F(\mathbf{x}^{(k)})\Delta_k - \Delta_{k+1}^T \left(\int_0^1 \nabla^2 F\big(\mathbf{x}^\star + t(\mathbf{x}^{(k)} - \mathbf{x}^\star)\big) dt \right) \Delta_k
\end{aligned}$$

$$\leq \left\| H(\mathbf{x}^{(k)}) - \nabla^2 F(\mathbf{x}^{(k)}) \right\|_{\mathscr{K}} \|\Delta_k\| \|\Delta_{k+1}\|$$
$$+ \int_0^1 \|\nabla^2 F(\mathbf{x}^{(k)}) - \nabla^2 F\big(\mathbf{x}^\star + t(\mathbf{x}^{(k)} - \mathbf{x}^\star)\big)\|_{\mathscr{K}} dt \|\Delta_k\| \|\Delta_{k+1}\|$$
$$\leq \left\| H(\mathbf{x}^{(k)}) - \nabla^2 F(\mathbf{x}^{(k)}) \right\|_{\mathscr{K}} \|\Delta_k\| \|\Delta_{k+1}\| + \frac{L}{2} \|\Delta_k\|^2 \|\Delta_{k+1}\|,$$

where $\|A\|_{\mathscr{K}}$ is as in (4). We also have $\Delta_{k+1}^T H(\mathbf{x}^{(k)}) \Delta_{k+1} \geq \lambda_{\min}^{\mathscr{K}}\big(H(\mathbf{x}^{(k)})\big) \|\Delta_{k+1}\|^2$. Assumption (20) implies $\lambda_{\min}^{\mathscr{K}}\big(H(\mathbf{x}^{(k)})\big) > 0$, and the result follows. □

Note that for the case of $H(\mathbf{x}^{(k)}) = \nabla^2 F(\mathbf{x}^{(k)})$, we exactly recover the convergence rate of the classical Newton's method [3, Proposition 1.4.1].

Using Sampling Lemma 2 and Structural Lemma 4, we are now in the position to present the main results of this section.

Algorithm 3 Locally Linearly Convergent SSN with Hessian Sub-Sampling

1: **Input:** $\mathbf{x}^{(0)}$, $0 < \delta < 1$, $0 < \varepsilon < 1$
2: - Set the sample size, $|\mathscr{S}|$, with ε and δ as described in Lemma 2
3: **for** $k = 0, 1, 2, \cdots$ until termination **do**
4: - Select a sample set, $\mathscr{S}$, of size $|\mathscr{S}|$ and $H(\mathbf{x}^{(k)})$ as in (10)
5: - Update $\mathbf{x}^{(k+1)}$ as in (2) with $\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)})$, $H(\mathbf{x}^{(k)})$, and $\alpha_k = 1$
6: **end for**

Theorem 5 (Error recursion of (2) with $\nabla F(\mathbf{x}^{(k)})$ and $H(\mathbf{x}^{(k)})$) *Let Assumptions (6) and (7) hold and let $0 < \delta < 1$ and $0 < \varepsilon < 1$ be given. Set $|\mathscr{S}|$ as in Lemma 2, and let $H(\mathbf{x}^{(k)})$ be as in (10). Then, for the update (2) with $\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)})$, $H(\mathbf{x}^{(k)})$, and $\alpha_k = 1$, with probability $1 - \delta$, we have*

$$\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| \leq \rho_0 \|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2, \tag{21a}$$

where

$$\rho_0 = \frac{\varepsilon}{(1-\varepsilon)}, \quad \textit{and} \quad \xi = \frac{L}{2(1-\varepsilon)\gamma}. \tag{21b}$$

Proof By Lemma 2, we get $\|H(\mathbf{x}) - \nabla^2 F(\mathbf{x})\|_{\mathscr{K}} / \lambda_{\min}^{\mathscr{K}}(H(\mathbf{x})) \leq \varepsilon/(1-\varepsilon)$. Now the results follow immediately by applying Lemma 4. □

Bounds given here exhibit a composite behavior where the error recursion, when far from the optimum, is first dominated by a quadratic term and then by a linear term near the optimum. Notably, the coefficient of the linear term, ρ_0, is indeed *independent* of any problem-related quantities, and only depends on the sub-sampling accuracy, ε. Of course, such problem dependent quantities indeed appear in the lower bound for the sample size, in the form of p and κ_1; see Lemma 2.

Now we establish sufficient conditions for Q-linear convergence of Algorithm 3.

Theorem 6 (Q-linear convergence of Algorithm 3) *Let Assumptions* (6) *and* (7) *hold and consider any* $0 < \rho_0 < \rho < 1$. *Using Algorithm 3 with* $\varepsilon \leq \rho_0/(1+\rho_0)$, *if*

$$\|\mathbf{x}^{(0)} - \mathbf{x}^\star\| \leq \frac{\rho - \rho_0}{\xi}, \tag{22}$$

with ξ *as in Theorem 5, we get Q-linear convergence*

$$\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq \rho \|\mathbf{x}^{(k-1)} - \mathbf{x}^\star\|, \quad k = 1, \ldots, k_0 \tag{23}$$

with probability $(1-\delta)^{k_0}$.

Proof Using this particular choice of ε, Theorem 5, for every k, yields $\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| \leq \rho_0 \|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2$. The result follows by requiring that $\rho_0 \|\mathbf{x}^{(0)} - \mathbf{x}^\star\| + \xi \|\mathbf{x}^{(0)} - \mathbf{x}^\star\|^2 \leq \rho \|\mathbf{x}^{(0)} - \mathbf{x}^\star\|$. Finally, let A_k denote the event that $\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq \rho \|\mathbf{x}^{(k-1)} - \mathbf{x}^\star\|$. The overall success probability is

$$\mathbf{Pr}\left(\bigcap_{k=1}^{k_0} A_k\right) = \mathbf{Pr}\left(A_{k_0} \mid \bigcap_{k=1}^{k_0-1} A_k\right) \mathbf{Pr}\left(\bigcap_{k=1}^{k_0-1} A_k\right) = \cdots = \prod_{k=1}^{k_0} \mathbf{Pr}\left(A_k \mid \bigcap_{i=1}^{k-1} A_i\right) = (1-\delta)^{k_0},$$

since for every k, the conditional probability of a successful update $\mathbf{x}^{(k+1)}$, given the past successful iterations $\{\mathbf{x}_i\}_{i=1}^k$, is $1-\delta$. □

If the Hessian approximation accuracy increases as the iterations progress, we can also obtain Q-superlinear rate. In fact, it seems reasonable to expect that the rate at which the Hessian estimation accuracy increases, determines the actual rate of Q-superlinear convergence. To verify this, Theorems 7 and 8 consider, respectively, geometric and logarithmic increase in the Hessian approximation accuracy. These, in turn, give rise to Q-superlinearly convergent iterates where the actual speed of convergence from one iteration to the next increases, respectively, at geometric and logarithmic rates; cf. $\rho(k)$ in the definition of Q-superlinear convergence in Sect. 1.4.

Algorithm 4 Locally Superlinearly Convergent SSN with Hessian Sub-Sampling

1: **Input:** $\mathbf{x}^{(0)}$, $0 < \delta < 1$, $0 < \varepsilon < 1$, $0 < \rho < 1$
2: **for** $k = 0, 1, 2, \cdots$ until termination **do**
3: - Set $\varepsilon^{(k)}$, for example as in Theorems 7 or 8
4: - Set the sample size, $|\mathscr{S}^{(k)}|$, with $\varepsilon^{(k)}$ and δ as in Lemma 2
5: - Select a sample set, $\mathscr{S}^{(k)}$, of size $|\mathscr{S}^{(k)}|$ and $H(\mathbf{x}^{(k)})$ as in (10)
6: - Update $\mathbf{x}^{(k+1)}$ as in (2) with $\mathbf{g}(\mathbf{x}^{(k)}) = \nabla F(\mathbf{x}^{(k)})$, $H(\mathbf{x}^{(k)})$, and $\alpha_k = 1$
7: **end for**

Theorem 7 (Q-superlinear convergence of Algorithm 4: geometric growth) *Let the assumptions of Theorem 6 hold. Using Algorithm 4, with* $\varepsilon^{(k)} = \rho^k \varepsilon$, $k = 0, 1, \ldots, k_0$, *if* $\mathbf{x}^{(0)}$ *satisfies* (22) *with* ρ, ρ_0, *and* $\xi^{(0)}$, *we get Q-superlinear convergence*

$$\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq \rho^k \|\mathbf{x}^{(k-1)} - \mathbf{x}^\star\|, \quad k = 1, \ldots, k_0, \tag{24}$$

with probability $(1-\delta)^{k_0}$*, where* $\xi^{(0)}$ *is as in* (21b) *with* $\varepsilon^{(0)}$.

Proof Theorem 5, for each k, gives $\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| \leq \rho_0^{(k)} \|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi^{(k)} \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2$, where $\rho_0^{(k)}$ and $\xi^{(k)}$ are as in (21b) using $\varepsilon^{(k)}$. Note that, by $\varepsilon^{(k)} = \rho^k \varepsilon$ and ε set as in Theorem 6, it follows that $\rho_0^{(0)} \leq \rho_0$, $\rho_0^{(k)} \leq \rho^k \rho_0$, and $\xi^{(k)} \leq \xi^{(k-1)}$. We prove the result by induction on k. Define $\Delta_k \triangleq \mathbf{x}^{(k)} - \mathbf{x}^\star$. For $k = 0$, by assumptions on ρ, ρ_0, and $\xi^{(0)}$, we have $\|\Delta_1\| \leq \rho_0^{(0)} \|\Delta_0\| + \xi^{(0)} \|\Delta_0\|^2 \leq \rho_0 \|\Delta_0\| + \xi^{(0)} \|\Delta_0\|^2 \leq \rho \|\Delta_0\|$. Now assume that (24) holds up to the iteration k. For $k+1$, we get $\|\Delta_{k+1}\| \leq \rho_0^{(k)} \|\Delta_k\| + \xi^{(k)} \|\Delta_k\|^2 \leq \rho^k \rho_0 \|\Delta_k\| + \xi^{(k)} \|\Delta_k\|^2 \leq \rho^k \rho_0 \|\Delta_k\| + \xi^{(0)} \|\Delta_k\|^2$. By induction hypothesis, we have $\|\Delta_{k-1}\| \leq \|\Delta_0\|$, and $\|\Delta_k\| \leq \rho^k \|\Delta_{k-1}\| < \rho^k (\rho - \rho_0)/\xi^{(0)}$, hence it follows that $\|\Delta_{k+1}\| \leq \rho^{k+1} \|\Delta_k\|$. □

Theorem 8 (Q-superlinear convergence of Algorithm 4: logarithmic growth) *Let Assumptions* (6) *and* (7) *hold. Using Algorithm 4 with* $\varepsilon^{(k)} = 1/(1 + 2\ln(4+k))$, $k = 0, 1, \ldots, k_0$, *if* $\mathbf{x}^{(0)}$ *satisfies* $\|\mathbf{x}^{(0)} - \mathbf{x}^\star\| \leq 2\gamma/((1 + 4\ln(2))\, L)$, *we get Q-superlinear convergence*

$$\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq \frac{1}{\ln(3+k)} \|\mathbf{x}^{(k-1)} - \mathbf{x}^\star\|, \quad k = 1, \ldots, k_0, \tag{25}$$

with probability $(1-\delta)^{k_0}$.

Proof By the choice of $\varepsilon^{(k)}$ in (21b), we have $\rho_0^{(k)} = 1/(2\ln(4+k))$, and as before $\rho_0^{(k)} < \rho_0^{(k-1)}$ and $\xi^{(k)} \leq \xi^{(k-1)}$. We again prove the result by induction on k. For $k = 0$, by assumptions on $\mathbf{x}^{(0)}$ and the choice of $\varepsilon^{(0)}$, we have $\|\Delta_1\| \leq \rho_0^{(0)} \|\Delta_0\| + \xi^{(0)} \|\Delta_0\|^2 = \|\Delta_0\|/(2\ln(4)) + \xi^{(0)} \|\Delta_0\|^2 \leq \|\Delta_0\|/\ln(4)$. Now assume that (25) holds up to the iteration k. For $k+1$, we get

$$\|\Delta_{k+1}\| \leq \rho_0^{(k)} \|\Delta_k\| + \xi^{(k)} \|\Delta_k\|^2 \leq \|\Delta_k\|/(2\ln(4+k)) + \xi^{(k)} \|\Delta_k\|^2. \tag{26}$$

Now consider $\phi(x) = \ln(4+x)/\ln(3+x)$. Since $\phi(0) < 2\ln(2)$ and $d\phi(x)/dx < 0$, $\forall x \geq 0$, i.e., $\phi(x)$ is decreasing, it follows that $\ln(4+k) \leq 2\ln(2)\ln(3+k)$, $\forall k \geq 0$. Since $\ln(3+k) \geq 1$, we get $(1 + 2\ln(4+k)) \leq \ln(3+k)\,(1 + 4\ln(2))$. By induction hypothesis, we have $\|\Delta_{k-1}\| \leq \|\Delta_0\|$, and so $\|\Delta_k\| \leq \|\Delta_{k-1}\|/\ln(3+k) < 2\gamma/(\ln(3+k)\,(1 + 2\ln(4))\, L)$, which by above implies that $\|\Delta_k\| \leq 2\gamma/((1 + 2\ln(4+k))\, L) = 1/(2\ln(4+k)\xi^{(k)})$. As a result, from (26), we get $\|\Delta_{k+1}\| \leq \|\Delta_k\|/(\ln(4+k))$. □

Theorems 7 and 8 require that the sample-sizes increase across iterations at some prescribed rates. Consequently, sooner or later, one will require sample sizes that are of the same order as n. In this light, the results of Theorems 7 and 8 are more applicable in early stages of the algorithms where the samples sizes are small (relative to n). However, sub-sampling, even if done at intermediary steps and prior to switching to

a full algorithm, can still offer valuable computational savings, e.g., see [36,37] and references therein for hybrid approaches to solve large-scale inverse problems.

Inexact update We now consider the unconstrained case, where $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$, and study iterations generated by inexact solutions to (2a) with $\alpha = 1$, i.e., $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{p}_k$, where $\mathbf{p}_k$ is as in (15a). In Theorem 2, we showed that a reasonably mild inexactness condition on θ_1 still gives a globally convergent method with convergence rate similar to that of the method with exact update (13a). We now show that similar inexactness condition is, indeed, sufficient to also guarantee a problem-independent local Q-linear convergence rate.

Theorem 9 *(*Q-Linear convergence of Algorithm 3*: inexact update) Let Assumptions (6) and (7) hold and* $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$. *Consider any* $0 < \rho_0 < \rho < 1$ *and*

$$\theta_1 \leq \rho_0 \sqrt{(1-\varepsilon)/4\tilde{\kappa}}.$$

Consider Algorithm 3 with inexact update (15a) in place of (13a)*, and* $\varepsilon \leq \rho_0/(2+\rho_0)$. *Further assume that, to solve (15a), we use CG initialized at zero. Then if (22) holds, we get Q-linear convergence as in* (23)*, with probability* $(1-\delta)^{k_0}$.

Proof For such inexact iteration we have $\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| = \|\mathbf{x}^{(k)} + \mathbf{p}_k - \mathbf{p}_k^\star + \mathbf{p}_k^\star - \mathbf{x}^\star\| \leq \|\mathbf{p}_k - \mathbf{p}_k^\star\| + \|\Delta_{k+1}\| \leq \|\mathbf{p}_k - \mathbf{p}_k^\star\| + \rho_0 \|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2$, where Δ_{k+1}, ρ_0, and ξ are as in Lemma 4 and $\mathbf{p}_k^\star$ is the exact solution, i.e., $\mathbf{p}_k^\star = -[H(\mathbf{x}^{(k)})]^{-1} \nabla F(\mathbf{x}^{(k)})$. Recall that by Lemma 2, with high probability, we have $\mathbf{v}^T H(\mathbf{x}^{(k)}) \mathbf{v} \geq (1-\varepsilon)\gamma \|\mathbf{v}\|^2$. Now, CG with $\mathbf{p}_k^{(0)} = \mathbf{0}$, gives

$$\|\mathbf{p}_k^{(t)} - \mathbf{p}_k^\star\| \leq \frac{2}{\sqrt{(1-\varepsilon)\gamma}} \left(\frac{\sqrt{\tilde{\kappa}/(1-\varepsilon)} - 1}{\sqrt{\tilde{\kappa}/(1-\varepsilon)} + 1} \right)^t \|\mathbf{p}_k^\star\|_{H(\mathbf{x}^{(k)})},$$

where $\mathbf{p}_k^{(t)}$ is the t^{th} iterate of CG, $\tilde{\kappa}$ is as in (9c), and $\|\mathbf{p}\|_A = \sqrt{\mathbf{p}^T A \mathbf{p}}$. We have $\|\mathbf{p}_k^\star\|_{H(\mathbf{x}^{(k)})} = \sqrt{\nabla F(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \nabla F(\mathbf{x}^{(k)})} \leq \|\nabla F(\mathbf{x}^{(k)})\| / \sqrt{(1-\varepsilon)\gamma}$. By (6b), $\|\nabla F(\mathbf{x}^{(k)})\| = \|\nabla F(\mathbf{x}^{(k)}) - \nabla F(\mathbf{x}^\star)\| \leq K \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|$. Hence,

$$\|\mathbf{p}_k^{(t)} - \mathbf{p}_k^\star\| \leq \frac{2}{(1-\varepsilon)} \kappa \left(\frac{\sqrt{\tilde{\kappa}/(1-\varepsilon)} - 1}{\sqrt{\tilde{\kappa}/(1-\varepsilon)} + 1} \right)^t \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|,$$

where κ is as in (8). Now t as in (16) gives $\|\mathbf{p}_k^{(t)} - \mathbf{p}_k^\star\| \leq \theta_1 \kappa \|\mathbf{x}^{(k)} - \mathbf{x}^\star\| / (\sqrt{(1-\varepsilon)\tilde{\kappa}})$. The assumption on θ_1, and noting $\kappa \leq \tilde{\kappa}$, implies $\|\mathbf{p}_k^{(t)} - \mathbf{p}_k^\star\| \leq \rho_0 \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|/2$. The rest of the proof follows by assumption on ε and similar to Theorem 6. □

3.2.2 Local convergence: sub-sampled Hessian and gradient

We now study (2) with both Hessian and gradient sub-sampling. We consider the setting where the gradient and Hessian are sub-sampled independently of each other.

The alternative is to use the same collection of indices and perform simultaneous sub-sampling for both. This latter case is not considered here, as in our opinion and experience, it does not seem to offer any practical and theoretical advantages.

We now present Lemma 5, as a structural lemma, followed by our main theorems.

Lemma 5 (Structural Lemma B) *Let Assumptions (7) and (20) hold. For the update (2) with* $\alpha_k = 1$, $H(\mathbf{x}^{(k)})$ *and* $\mathbf{g}(\mathbf{x}^{(k)})$ *as in (10) and (11), respectively, we have* $\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| \leq \eta + \rho_0\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi\|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2$, *where* ρ_0, ξ *are as in Lemma 4 and* $\eta = \|\nabla F(\mathbf{x}^{(k)}) - \mathbf{g}(\mathbf{x}^{(k)})\|_{\mathscr{K}}/\lambda^{\mathscr{K}}_{\min}(H(\mathbf{x}^{(k)}))$, *with* $\|\mathbf{v}\|_{\mathscr{K}}$ *as in (4).*

Proof The result follows as in the proof of Lemma 4, and using the identity $\mathbf{g}(\mathbf{x}^{(k)}) = \mathbf{g}(\mathbf{x}^{(k)}) - \nabla F(\mathbf{x}^{(k)}) + \nabla F(\mathbf{x}^{(k)})$, and noting that $|\Delta_{k+1}^T(\mathbf{g}(\mathbf{x}^{(k)}) - \nabla F(\mathbf{x}^{(k)}))| \leq \|\mathbf{g}(\mathbf{x}^{(k)}) - \nabla F(\mathbf{x}^{(k)})\|_{\mathscr{K}}\|\Delta_{k+1}\|$, simply follows from the definition of $\|.\|_{\mathscr{K}}$ in (4). □

Theorem 10 (Error recursion of (2) with $\mathbf{g}(\mathbf{x}^{(k)})$ and $H(\mathbf{x}^{(k)})$) *Let Assumptions (6) and (7) hold, and let* $0 < \delta, \varepsilon_1, \varepsilon_2 < 1$ *be given. Set* $|\mathscr{S}_H|$ *as in Lemma 2 with* (ε_1, δ) *and* $|\mathscr{S}_{\mathbf{g}}|$ *as in Lemma 3 with* (ε_2, δ) *and* $G(\mathbf{x}^{(k)})$. *Independently, choose* $\mathscr{S}_H$ *and* $\mathscr{S}_{\mathbf{g}}$, *and let* $H(\mathbf{x}^{(k)})$ *and* $\mathbf{g}(\mathbf{x}^{(k)})$ *be as in (10) and (11), respectively. For the update (2) with* $\alpha_k = 1$, *with probability* $(1-\delta)^2$ *we have* $\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| \leq \eta + \rho_0\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi\|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2$, *where* $\eta = \varepsilon_2/((1-\varepsilon_1)\gamma)$, $\rho_0 = \varepsilon_1/(1-\varepsilon_1)$, *and* $\xi = L/(2(1-\varepsilon_1)\gamma)$.

Similar to Theorem 5, the bounds given here exhibit a composite behavior where the error is, at first, dominated by a quadratic term, which transitions to a linear term, and finally is dominated by the approximation error in the gradient. R-linear convergence of Algorithms 5 is given in Theorem 11.

Algorithm 5 Locally Linearly Convergent SSN with Hessian and Gradient Sub-Sampling

Input: $\mathbf{x}^{(0)}$, $0 < \delta < 1$, $0 < \varepsilon_1 < 1$, $0 < \varepsilon_2 < 1$ and $0 < \rho < 1$
- Set the sample size, $|\mathscr{S}_H|$, with ε_1 and δ as in Lemma 2
for $k = 0, 1, 2, \cdots$ until termination **do**
  - Select a sample set, $\mathscr{S}_H$, of size $|\mathscr{S}_H|$ and form $H(\mathbf{x}^{(k)})$ as in (10)
  - Set $\varepsilon_2^{(k)} = \rho^k\varepsilon_2$
  - Set the sample size, $|\mathscr{S}_{\mathbf{g}}^{(k)}|$, with $\varepsilon_2^{(k)}$, δ and $\mathbf{x}^{(k)}$ as in Lemma 3 with $G(\mathbf{x}^{(k)})$
  - Select a sample set, $\mathscr{S}_{\mathbf{g}}^{(k)}$ of size $|\mathscr{S}_{\mathbf{g}}^{(k)}|$ and form $\mathbf{g}(\mathbf{x}^{(k)})$ as in (11)
  - Update $\mathbf{x}^{(k+1)}$ as in (2) with $H(\mathbf{x}^{(k)})$, $\mathbf{g}(\mathbf{x}^{(k)})$ and $\alpha_k = 1$
end for

Theorem 11 (R-linear convergence of Algorithm 5) *Let Assumptions (6) and (7) hold. Consider any* $0 < \rho, \rho_0, \rho_1 < 1$ *such that* $\rho_0 + \rho_1 < \rho$. *Let* $\varepsilon_1 \leq \rho_0/(1+\rho_0)$, *and define* $c \triangleq 2\big(\rho - (\rho_0 + \rho_1)\big)(1-\varepsilon_1)\gamma/L$. *Using Algorithm 5 with* $\varepsilon_2 \leq (1-\varepsilon_1)\gamma\rho_1 c$, *if the initial iterate,* $\mathbf{x}^{(0)}$, *satisfies* $\|\mathbf{x}^{(0)} - \mathbf{x}^\star\| \leq c$, *we get R-linear convergence*

$$\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq c\rho^k, \tag{27}$$

with probability $(1-\delta)^{2k}$.

Proof Using Theorem 10, the particular choice of ε_1 and $\varepsilon_2^{(k)} = \rho^k \varepsilon_2$, for each k, gives $\|\mathbf{x}^{(k+1)} - \mathbf{x}^\star\| \leq \eta^{(k)} + \rho_0 \|\mathbf{x}^{(k)} - \mathbf{x}^\star\| + \xi \|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2$, where $\eta^{(k)} = \rho \eta^{(k-1)}$ and $\eta^{(0)} \leq \rho_1 c$. We prove the result by induction on k. Define $\Delta_k \triangleq \mathbf{x}^{(k)} - \mathbf{x}^\star$. For $k = 0$, by the assumption on $\mathbf{x}^{(0)}$ and the definition of $c = \big(\rho - (\rho_0 + \rho_1)\big)/\xi$, we have $\|\Delta_1\| \leq \eta^{(0)} + \rho_0 \|\Delta_0\| + \xi \|\Delta_0\|^2 \leq \rho_1 c + \rho_0 c + \xi c^2 = \rho c$. Now assume that (27) holds for k. For $k+1$, we get $\|\Delta_{k+1}\| \leq \eta^{(k)} + \rho_0 \|\Delta_k\| + \xi \|\Delta_k\|^2 = \rho^k \eta^{(0)} + \rho_0 \|\Delta_k\| + \xi \|\Delta_k\|^2 \leq \rho^k \rho_1 c + \rho_0 \rho^k c + \xi \rho^{2k} c^2 \leq \rho^k \left(\rho_1 c + \rho_0 c + \xi c^2\right) = \rho^{k+1} c$, where the first and the second inequalities follow, respectively, from the induction hypothesis and $\rho < 1$, and the final equality is by definition of c. □

Theorem 11 implies that, to get linear convergence rate, estimation of the gradient must be done, progressively, more accurately, whereas the sample size for the Hessian can remain unchanged across iterations. This is in line with the common knowledge where, as the iterations get closer to the optimal solution, the accuracy of the gradient estimation is more important than that of the Hessian.

Similar to Theorem 9, it is possible to obtain results for a variant of Algorithm 5 with inexact updates. We simply state the following result and we omit the proof.

Theorem 12 (R-linear convergence of Algorithm 5: inexact update) *Let Assumptions* (6) *and* (7) *hold and* $\mathscr{D} = \mathscr{C} = \mathbb{R}^p$. *Consider any* $0 < \rho, \rho_0, \rho_1 < 1$ *such that* $\rho_0 + \rho_1 < \rho$ *and*

$$\theta_1 \leq \rho_0 \sqrt{(1-\varepsilon)/4\tilde{\kappa}}.$$

Let $\varepsilon_1 \leq \rho_0/(2+\rho_0)$, *define* $c \triangleq 2\big(\rho - (\rho_0 + \rho_1)\big)(1-\varepsilon_1)\gamma/L$ *and set* $\varepsilon_2 \leq (1-\varepsilon_1)\gamma\rho_1 c$. *Consider Algorithm 5 with inexact update* (19) *in place of* (18a) *and, assume that, to solve* (19), *we use CG initialized at zero. If* $\|\mathbf{x}^{(0)} - \mathbf{x}^\star\| \leq c$, *we get locally R-linear convergence as in* (27), *with probability* $(1-\delta)^{2k}$.

3.3 Putting it all together

Theorem 1 guarantees the global convergence of Algorithm 1 with a linear rate that depends on the problem specific quantities i.e., κ and $\tilde{\kappa}$. However, by Theorem 6, the locally linear convergence rate of such SSN variant with $\alpha_k = 1$ is indeed *problem-independent*. In fact, it is possible to combine both results to show that Algorithm 1 is globally convergent with a (super)linear and problem-independent local rate. We also show that after certain number of iterations, Armijo line search automatically adopts the natural step size of the classical Newton's method, i.e., $\alpha_k = 1$, for *all subsequent iterations*. We note that here, we give unifying results for the case of exact update. Since the extensions to inexact updates is done similarly, we omit the details.

Theorem 13 (Global convergence of Algorithm 1 with problem-independent local rate) *Let Assumptions* (6) *and* (7) *hold, and consider any* $0 < \rho_0 < \rho < 1$. *Using Algorithm 1 with any* $\mathbf{x}^{(0)} \in \mathbb{R}^p$,

$$\varepsilon \leq \min\left\{(1-2\beta)/(2(1-\beta)),\, \rho_0/(4(1+\rho_0)\sqrt{\kappa_1})\right\},$$

and $0 < \beta < 1/2$, *after at most* $k \in \mathcal{O}(\kappa\tilde{\kappa}/(1-\varepsilon))$ *iterations, with probability* $(1-\delta)^k$ *we get problem-independent Q-linear convergence, i.e.,* $\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq \rho\|\mathbf{x}^{(k-1)} - \mathbf{x}^\star\|$, *where* κ, κ_1 *and* $\tilde{\kappa}$ *are defined in* (8), (9a) *and* (9c), *respectively. Moreover, the step size of* $\alpha_k = 1$ *is selected in* (13b) *for all subsequent iterations.*

Proof The choice of ε is to meet a requirement of Theorem 6 and account for the differences between Lemmas 1 and 2. The rest of the proof follows similar line of reasoning as in [6, Section 9.5.3]. Define $\mathbf{x}_\alpha = \mathbf{x}^{(k)} + \alpha\mathbf{p}_k$. By (7), we get $\mathbf{p}_k^T(\nabla^2 F(\mathbf{x}_\alpha) - \nabla^2 F(\mathbf{x}^{(k)}))\mathbf{p}_k \leq \alpha L\|\mathbf{p}_k\|^3$, which, gives $\mathbf{p}_k^T\nabla^2 F(\mathbf{x}_\alpha)\mathbf{p}_k \leq \mathbf{p}_k^T\nabla^2 F(\mathbf{x}^{(k)})\mathbf{p}_k + \alpha L\|\mathbf{p}_k\|^3$. Defining $\widehat{F}(\alpha) \triangleq F(\mathbf{x}^{(k)} + \alpha\mathbf{p}_k)$, we get $\widehat{F}''(\alpha) \leq \widehat{F}''(0) + \alpha L\|\mathbf{p}_k\|^3$. Integrating this inequality gives $\widehat{F}'(\alpha) \leq \widehat{F}'(0) + \alpha\widehat{F}''(0) + \alpha^2 L\|\mathbf{p}_k\|^3/2$. Integrating again yields $\widehat{F}(\alpha) \leq \widehat{F}(0) + \alpha\widehat{F}'(0) + \alpha^2\widehat{F}''(0)/2 + \alpha^3 L\|\mathbf{p}_k\|^3/6$. We also have that $\|\mathbf{p}_k\|^2 = \|[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)})\|^2 \leq \nabla F(\mathbf{x}^{(k)})^T[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)})/((1-\varepsilon)\gamma)$. In addition, we get $\widehat{F}'(0) = \mathbf{p}_k^T\nabla F(\mathbf{x}^{(k)}) = -\nabla F(\mathbf{x}^{(k)})^T[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)})$ and

$$\begin{aligned}\widehat{F}''(0) = \mathbf{p}_k^T\nabla^2 F(\mathbf{x}^{(k)})\mathbf{p}_k &= \nabla F(\mathbf{x}^{(k)})^T[H(\mathbf{x}^{(k)})]^{-1}\nabla^2 F(\mathbf{x}^{(k)})[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)}) \\ &\leq \nabla F(\mathbf{x}^{(k)})^T[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)})/(1-\varepsilon).\end{aligned}$$

For the last inequality, recall that by the choice of ε and Lemma 2, we have $\|H(\mathbf{x}^{(k)}) - \nabla^2 F(\mathbf{x}^{(k)})\| \leq \varepsilon\gamma$. Hence, for any $\mathbf{v}$, we get

$$\begin{aligned}\mathbf{v}^T[H(\mathbf{x}^{(k)})]^{-1}\nabla^2 F(\mathbf{x}^{(k)})[H(\mathbf{x}^{(k)})]^{-1}\mathbf{v} - \mathbf{v}^T[H(\mathbf{x}^{(k)})]^{-1}\mathbf{v} &\leq \varepsilon\gamma\mathbf{v}^T[H(\mathbf{x}^{(k)})]^{-2}\mathbf{v} \\ &\leq \varepsilon\mathbf{v}^T[H(\mathbf{x}^{(k)})]^{-1}\mathbf{v}/(1-\varepsilon).\end{aligned}$$

This latter inequality in turn gives

$$\mathbf{v}^T[H(\mathbf{x}^{(k)})]^{-1}\nabla^2 F(\mathbf{x}^{(k)})[H(\mathbf{x}^{(k)})]^{-1}\mathbf{v} \leq \mathbf{v}^T[H(\mathbf{x}^{(k)})]^{-1}\mathbf{v}/(1-\varepsilon).$$

Hence, with $\alpha = 1$ and denoting $c(\mathbf{x}) \triangleq \nabla F(\mathbf{x})^T[H(\mathbf{x})]^{-1}\nabla F(\mathbf{x})$, we have

$$\begin{aligned}F(\mathbf{x}^{(k)} + \mathbf{p}_k) &\leq F(\mathbf{x}^{(k)}) + \left(\frac{1}{2(1-\varepsilon)} - 1\right)c(\mathbf{x}^{(k)}) + \frac{L}{6}\left(\frac{1}{(1-\varepsilon)\gamma}c(\mathbf{x}^{(k)})\right)^{3/2} \\ &\leq F(\mathbf{x}^{(k)}) + c(\mathbf{x}^{(k)})\left(\frac{1}{2(1-\varepsilon)} - 1 + \frac{L}{6}\left(\frac{1}{(1-\varepsilon)\gamma}\right)^{3/2}c(\mathbf{x}^{(k)})^{1/2}\right).\end{aligned}$$

From $c(\mathbf{x}) \leq \|\nabla F(\mathbf{x})\|^2/((1-\varepsilon)\gamma)$, we see that if

$$\|\nabla F(\mathbf{x}^{(k)})\| \leq 3(1-\varepsilon)\gamma^2\big(1 - 2\varepsilon - 2(1-\varepsilon)\beta\big)/L, \tag{28}$$

then we get $F(\mathbf{x}^{(k)} + \mathbf{p}_k) \leq F(\mathbf{x}^{(k)}) - \beta\nabla F(\mathbf{x}^{(k)})^T[H(\mathbf{x}^{(k)})]^{-1}\nabla F(\mathbf{x}^{(k)}) = F(\mathbf{x}^{(k)}) + \beta\mathbf{p}_k^T\nabla F(\mathbf{x}^{(k)})$, which implies that (13b) is satisfied with $\alpha = 1$.

The proof is complete if we can find k such that both the sufficient condition of Theorem 6 as well as (28) is satisfied. First, note that from Theorem 1, Assumption (6b)

and the iteration-independent lower bound on α_k, we get $\|\nabla F(\mathbf{x}^{(k)})\|^2 \leq 2K(1-\hat{\rho})^k(F(\mathbf{x}^{(0)}) - F(\mathbf{x}^\star))$, where $\hat{\rho} = 4\beta(1-\beta)(1-\varepsilon)/(\tilde{\kappa}\kappa)$. In order to satisfy (28), we require that

$$2K(1-\hat{\rho})^k(F(\mathbf{x}^{(0)}) - F(\mathbf{x}^\star)) \leq 4L^{-2}(1-\varepsilon)^2\gamma^4\big(1-2\varepsilon-2(1-\varepsilon)\beta\big)^2(\rho_1-\rho_0)^2,$$

which is satisfied as long as $k \in \Omega(\kappa\tilde{\kappa}/(1-\varepsilon))$. From Theorem 1 and Assumption (6b),we get $\|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2 \leq 2(1-\hat{\rho})^k(F(\mathbf{x}^{(0)}) - F(\mathbf{x}^\star))/\gamma$, which implies that

$$\begin{aligned}\|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2 &\leq 4(1-\varepsilon)^2\gamma^3\big(1-2\varepsilon-2(1-\varepsilon)\beta\big)^2(\rho_1-\rho_0)^2/(KL^2)\\ &\leq 4L^{-2}(1-\varepsilon)^2\gamma^2(\rho_1-\rho_0)^2.\end{aligned}$$

Hence, the condition of Theorem 6 holds and the results follow. □

Note that the ε required by Theorem 13 implies a sample size of $\tilde{\mathcal{O}}(\kappa_1^2)$. It is also possible to obtain a globally convergent algorithm with locally superlinear rate of convergence using Algorithm 1 with iteration dependent $\hat{\varepsilon}^{(k)}$ as $\hat{\varepsilon}^{(k)} \leq \varepsilon^{(k)}/(4\sqrt{\kappa_1})$, where $\varepsilon^{(k)}$ is chosen as in Theorems 7 or 8. The details are similar to Theorem 13 and are omitted here.

We now give similar results for Algorithm 2.

Theorem 14 (Global convergence of Algorithm 2 with problem-independent local rate) *Let Assumptions* (6) *and* (7) *hold. Consider any* $0 < \rho_0, \rho_1, \rho_2 < 1$ *such that* $\rho_0 + \rho_1 < \rho_2$, $\beta \leq 1/2$, $\sigma \geq 4\tilde{\kappa}/(1-\beta)$, *and*

$$\begin{aligned}\varepsilon_1 &\leq \min\left\{(1-2\beta)/(2(1-\beta)),\, \rho_0/(4(1+\rho_0)\sqrt{\kappa_1})\right\},\\ \varepsilon_2^{(0)} &\leq \begin{cases}\varepsilon\gamma^2/(L\sqrt{\tilde{\kappa}}), & \text{If } \gamma^2 \leq L\sqrt{\tilde{\kappa}},\\ \varepsilon L\sqrt{\tilde{\kappa}}/\gamma^2, & \text{Otherwise},\end{cases}\\ \varepsilon_2^{(k)} &= \rho_2\varepsilon_2^{(k-1)}, \quad k = 1, 2, \ldots,\end{aligned}$$

where $\varepsilon \leq (1-\varepsilon_1)^2(1-2\varepsilon_1-2(1-\varepsilon_1)\beta)^2\rho_1(\rho_2-(\rho_0+\rho_1))/3$. *Using Algorithm 2 with any* $\mathbf{x}^{(0)} \in \mathbb{R}^p$ *and Step 7 replaced by* $\|\mathbf{g}(\mathbf{x}^{(k)})\| < \sigma\sqrt{\rho_2^k\varepsilon}$, *after at most* $k_0 \in \mathcal{O}(\kappa\tilde{\kappa}/(1-\varepsilon_1))$ *iterations, we have the following with probability* $(1-\delta)^{2k}$ *for* $k \geq k_0$: *if "STOP", then* $\|\nabla F(\mathbf{x}^{(k)})\| < (1+\sigma)\sqrt{\rho_2^k\varepsilon}$, *otherwise, we get problem-independent R-linear convergence, i.e.,* $\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq c\rho_2^{(k-k_0)}$, *where c is as defined in Theorem 11. Moreover,* $\alpha_k = 1$ *is selected in* (18b) *for all subsequent iterations.*

Proof The choice of ε_1 and $\varepsilon_2^{(k)}$ is to meet a requirement of Theorem 11 and account for the differences between Lemma 1 and 2. As in the proof of Theorem 13, we get $\widehat{F}(\alpha) \leq \widehat{F}(0) + \alpha\widehat{F}'(0) + \alpha^2\widehat{F}''(0)/2 + \alpha^3L\|\mathbf{p}_k\|^3/6$. We also have $\|\mathbf{p}_k\|^2 = \|[H(\mathbf{x}^{(k)})]^{-1}\mathbf{g}(\mathbf{x}^{(k)})\|^2 \leq \mathbf{g}(\mathbf{x}^{(k)})^T[H(\mathbf{x}^{(k)})]^{-1}\mathbf{g}(\mathbf{x}^{(k)})/\big((1-\varepsilon_1)\gamma\big)$. By self-duality of the vector ℓ_2 norm, i.e., $\|\mathbf{v}\|_2 = \sup\{\mathbf{w}^T\mathbf{v};\ \|\mathbf{w}\|_2 = 1\}$, and $\|\nabla F(\mathbf{x}^{(k)}) - \mathbf{g}(\mathbf{x}^{(k)})\| \leq$

$\varepsilon_2^{(k)}$, we get $\mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) \leq \mathbf{p}_k^T \mathbf{g}(\mathbf{x}^{(k)}) + \varepsilon_2^{(k)} \|\mathbf{p}_k\|$, and so

$$\widehat{F}'(0) = \mathbf{p}_k^T \nabla F(\mathbf{x}^{(k)}) \leq -\mathbf{g}(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \mathbf{g}(\mathbf{x}^{(k)}) + \varepsilon_2^{(k)} \|\mathbf{p}_k\|.$$

As in the proof of Theorem 13, we also have

$$\widehat{F}''(0) = \mathbf{p}_k^T \nabla^2 F(\mathbf{x}^{(k)}) \mathbf{p}_k \leq \mathbf{g}(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \mathbf{g}(\mathbf{x}^{(k)})/(1-\varepsilon_1).$$

Hence, with $\alpha = 1$ and denoting $h(\mathbf{x}) \triangleq \mathbf{g}(\mathbf{x})^T [H(\mathbf{x})]^{-1} \mathbf{g}(\mathbf{x})$, we get

$$\begin{aligned} F(\mathbf{x}_\alpha) &\leq F(\mathbf{x}^{(k)}) + \Big(\frac{1}{2(1-\varepsilon_1)} - 1\Big) h(\mathbf{x}^{(k)}) + \frac{L}{6}\Big(\frac{h(\mathbf{x}^{(k)})}{(1-\varepsilon_1)\gamma}\Big)^{3/2} + \varepsilon_2^{(k)}\Big(\frac{h(\mathbf{x}^{(k)})}{(1-\varepsilon_1)\gamma}\Big)^{1/2} \\ &\leq F(\mathbf{x}^{(k)}) + h(\mathbf{x}^{(k)}) \left[\frac{1}{2(1-\varepsilon_1)} - 1 + \frac{L}{6}\Big(\frac{h(\mathbf{x}^{(k)})^{1/3}}{(1-\varepsilon_1)\gamma}\Big)^{3/2} + \varepsilon_2^{(k)}\Big(\frac{h(\mathbf{x}^{(k)})^{-1}}{(1-\varepsilon_1)\gamma}\Big)^{1/2} \right] \\ &\leq F(\mathbf{x}^{(k)}) + h(\mathbf{x}^{(k)}) \left[\frac{1}{2(1-\varepsilon_1)} - 1 + \frac{L\|\mathbf{g}(\mathbf{x}^{(k)})\|}{6(1-\varepsilon_1)^2\gamma^2} + \varepsilon_2^{(k)}\Big(\frac{\tilde{\kappa}\|\mathbf{g}(\mathbf{x}^{(k)})\|^{-2}}{1-\varepsilon_1}\Big)^{1/2} \right], \end{aligned}$$

where the last inequality follows from $\|\mathbf{g}(\mathbf{x})\|^2/\widehat{K}_{|\mathscr{S}|} \leq h(\mathbf{x}) \leq \|\mathbf{g}(\mathbf{x})\|^2/((1-\varepsilon_1)\gamma)$. Now denoting $y \triangleq \|\mathbf{g}(\mathbf{x}^{(k)})\|$ and $A \triangleq L/(6(1-\varepsilon_1)^2\gamma^2)$, $B \triangleq 0.5/(1-\varepsilon_1) - 1 + \beta$, and $C \triangleq \varepsilon_2^{(k)}(\tilde{\kappa}/(1-\varepsilon_1))^{1/2}$, we require that $Ay^2 + By + C \leq 0$. After a little bit of algebra, the roots of this polynomial can be written as

$$\begin{aligned} y_1, y_2 = \; & \frac{3(1-\varepsilon_1)\gamma^2(1-2\varepsilon_1-2(1-\varepsilon_1)\beta)}{2L} \\ & \pm \frac{\sqrt{9(1-\varepsilon_1)^2\gamma^4(1-2\varepsilon_1-2(1-\varepsilon_1)\beta)^2 - 24(1-\varepsilon_1)^{3/2}\gamma^2 L\varepsilon_2^{(k)}\tilde{\kappa}^{1/2}}}{2L}. \end{aligned}$$

Let us define $q_1(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L) \triangleq (q - \sqrt{q^2 - r})/(2L)$ and $q_2(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L) \triangleq (q + \sqrt{q^2 - r})/(2L)$, where $q \triangleq 3(1-\varepsilon_1)\gamma^2(1-2\varepsilon_1-2(1-\varepsilon_1)\beta)$ and also $r \triangleq 24(1-\varepsilon_1)^{3/2}\gamma^2 L\varepsilon_2^{(k)}\tilde{\kappa}^{1/2}$. It is easy to see that $q_1(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L)$ is increasing with $\varepsilon_2^{(k)}$ and $q_1(\varepsilon_1, 0, \beta, \tilde{\kappa}, L) = 0$, while $q_2(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L)$ is decreasing with $\varepsilon_2^{(k)}$ and also $q_2(\varepsilon_1, 0, \beta, \tilde{\kappa}, L)$ is equal to the right hand side of (28). In order to ensure that q_1 and q_2 are real, we also need to have $\varepsilon_2^{(k)} \leq 3\sqrt{(1-\varepsilon_1)}\gamma^2(1-2\varepsilon_1-2(1-\varepsilon_1)\beta)^2/(8L\sqrt{\tilde{\kappa}})$. Now if

$$q_1(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L) \leq \|\mathbf{g}(\mathbf{x}^{(k)})\| \leq q_2(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L), \tag{29}$$

we get $F(\mathbf{x}^{(k)} + \mathbf{p}_k) \leq F(\mathbf{x}^{(k)}) - \beta \mathbf{g}(\mathbf{x}^{(k)})^T [H(\mathbf{x}^{(k)})]^{-1} \mathbf{g}(\mathbf{x}^{(k)}) = F(\mathbf{x}^{(k)}) + \beta \mathbf{p}_k^T \mathbf{g}(\mathbf{x}^{(k)})$, which implies that (18b) is satisfied with $\alpha = 1$. The left hand side of (29) is enforced by the stopping criterion of the algorithm as for any $\varepsilon_2^{(k)}$,

$q_1(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L) \leq \sigma\sqrt{\varepsilon_2^{(k)}}$. Indeed, from $\sqrt{q^2 - r} \geq q - \sqrt{r}$, it implies that

$$q_1(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L) \leq \sqrt{r}/(2L) \leq \sqrt{6\gamma^2\rho_2^k\varepsilon_2^{(0)}\tilde{\kappa}/(L\sqrt{\tilde{\kappa}})} = \sqrt{6\rho_2^k\varepsilon\tilde{\kappa}} \leq \sigma\sqrt{\rho_2^k\varepsilon}.$$

The proof is complete if we can find k such that both the sufficient condition of Theorem 11 as well as the right hand side of (29) is satisfied. First note that from Theorem 3, Assumption (6b) and by using the iteration-independent lower bound on α_k, it follows that $\|\nabla F(\mathbf{x}^{(k)})\|^2 \leq 2K(1-\hat{\rho})^k(F(\mathbf{x}^{(0)}) - F(\mathbf{x}^\star))$, where $\hat{\rho} = 8\beta(1-\beta)(1-\varepsilon_1)/(9\tilde{\kappa}\kappa)$. If the stopping criterion succeeds, then since $\varepsilon_2^{(k)} \leq 1$, by $\|\mathbf{g}(\mathbf{x}^{(k)})\| \geq \|\nabla F(\mathbf{x}^{(k)})\| - \varepsilon_2^{(k)}$, we get $\|\nabla F(\mathbf{x}^{(k)})\| < (1+\sigma)\sqrt{\rho_2^k\varepsilon}$. Otherwise, by $\|\mathbf{g}(\mathbf{x}^{(k)})\| \leq \|\nabla F(\mathbf{x}^{(k)})\| + \varepsilon_2^{(k)} \leq \|\nabla F(\mathbf{x}^{(k)})\| + \sqrt{\varepsilon_2^{(k)}}$, we get $(\sigma-1)\sqrt{\varepsilon_2^{(k)}} \leq \|\nabla F(\mathbf{x}^{(k)})\|$, which implies that $\|\mathbf{g}(\mathbf{x}^{(k)})\| \leq \sigma\|\nabla F(\mathbf{x}^{(k)})\|/(\sigma-1) \leq 2\|\nabla F(\mathbf{x}^{(k)})\|$. Now, to satisfy the right hand side of (29), we require that

$$8K(1-\hat{\rho})^k(F(\mathbf{x}^{(0)}) - F(\mathbf{x}^\star)) \leq 16(\rho_2 - (\rho_0+\rho_1))^2 q_2^2(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L)/9,$$

which is satisfied as long as $k \in \Omega(\kappa\tilde{\kappa}/(1-\varepsilon_1))$. Again, from Theorem 3 and Assumption (6b), we get $\|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2 \leq 2(1-\hat{\rho})^k(F(\mathbf{x}^{(0)}) - F(\mathbf{x}^\star))/\gamma$, which implies that

$$\begin{aligned}
\|\mathbf{x}^{(k)} - \mathbf{x}^\star\|^2 &\leq \frac{16(\rho_2 - (\rho_0+\rho_1))^2 q_2^2(\varepsilon_1, \varepsilon_2^{(k)}, \beta, \tilde{\kappa}, L)}{36\gamma K} \\
&\leq \frac{4(\rho_2 - (\rho_0+\rho_1))^2(1-\varepsilon_1)^2\gamma^4(1-2\varepsilon_1 - 2(1-\varepsilon_1)\beta)^2}{\gamma K L^2} \\
&\leq \frac{4(\rho_2 - (\rho_0+\rho_1))^2(1-\varepsilon_1)^2\gamma^2}{L^2} = c^2.
\end{aligned}$$

Hence, the sufficient condition of Theorem 11 is also satisfied. □

Sampling complexities implied by Theorem 14 can be made more explicit as follows. For simplicity, assume that $\rho_0 = \rho_1 = 1/8$, $\rho_2 = 1/2$, $\gamma^2 \leq L\sqrt{\tilde{\kappa}}$ and $\beta \leq 1/10$, for which we have $(1-\varepsilon_1)^2(1 - 2\varepsilon_1 - 2(1-\varepsilon_1)\beta)^2 \geq 1/4$, $\forall \varepsilon_1 \leq 1/10$, i.e., $\varepsilon \in \mathcal{O}(1)$. Further, for $G(\mathbf{x})$ in Lemma 3, assume that we have a uniform estimate as $G(\mathbf{x}) \leq G < \infty$. The requirements on ε_1 and $\varepsilon_2^{(k)}$ in Theorem 14 imply $|\mathcal{S}_H| \in \tilde{\mathcal{O}}(\kappa_1^2)$, and $|\mathcal{S}_{\mathbf{g}}^{(k)}| \in \tilde{\mathcal{O}}(4^k G^2 L^2 \tilde{\kappa}/\gamma^4)$, respectively.

3.4 Comparison of computational complexities

We now present a brief overview of the computational complexities implied by the main results of this paper. We consider both exact and inexact variants of all these algorithms for unconstrained variant of (1), i.e., $\mathcal{D} = \mathcal{C} = \mathbb{R}^p$. For inexact solutions, we consider the tolerances of $\theta_1 \leq \sqrt{(1-\varepsilon)/(4\tilde{\kappa})}$ and $\theta_2 = 1/2$ in (15) and (19). We

Table 3 Complexity comparison of various Newton-type methods for unconstrained version of (1) to achieve sub-optimality $F(\mathbf{x}^{(k)}) - F(\mathbf{x}^\star) \leq \varsigma$ for some $\varsigma \leq 1$. For step-size, we consider iterations with the worst-case α_k, as prescribed by the corresponding theorem. The notation $\tilde{\mathcal{O}}$ implies hidden logarithmic factors, e.g., $\ln(\kappa)$, $\ln(\tilde{\kappa})$, $\ln(p)$, and $\ln(1/\delta)$. Constants $\gamma, \kappa, \tilde{\kappa}$ and κ_1 are defined in Sect. 1.5. Also, ε is the Hessian approximation accuracy parameter from Lemma 1. For $G(\mathbf{x})$ in Lemma 3, we assume that we have a uniform estimate as $G(\mathbf{x}) \leq G < \infty$

Method	Evaluating $\mathbf{g}(\mathbf{x})$	Evaluating one Hessian-Vector Product, $H(\mathbf{x})\mathbf{v}$	# of Iterations of CG on (2a)	Iteration Complexity	Reference
Newton-Exact	$\mathcal{O}(np)$	$\mathcal{O}(np)$	$\mathcal{O}(p)$	$\mathcal{O}(\kappa^2 \ln \frac{1}{\varsigma})$	Folklore
Newton-CG	$\mathcal{O}(np)$	$\mathcal{O}(np)$	$\tilde{\mathcal{O}}(\sqrt{\kappa})$	$\mathcal{O}(\kappa^2 \ln \frac{1}{\varsigma})$	Folklore
Alg 1 with (13a)	$\mathcal{O}(np)$	$\tilde{\mathcal{O}}(\frac{\kappa_1}{\varepsilon^2} p)$	$\mathcal{O}(p)$	$\mathcal{O}(\frac{\kappa\tilde{\kappa}}{1-\varepsilon} \ln \frac{1}{\varsigma})$	Theorem 1
Alg 1 with (15)	$\mathcal{O}(np)$	$\tilde{\mathcal{O}}(\frac{\kappa_1}{\varepsilon^2} p)$	$\tilde{\mathcal{O}}(\sqrt{\frac{\tilde{\kappa}}{1-\varepsilon}})$	$\mathcal{O}(\frac{\kappa\tilde{\kappa}}{1-\varepsilon} \ln \frac{1}{\varsigma})$	Theorem 2
Alg 2 with (18a)	$\tilde{\mathcal{O}}(\frac{G^2\tilde{\kappa}^2}{\varsigma\gamma} p)$	$\tilde{\mathcal{O}}(\frac{\kappa_1}{\varepsilon^2} p)$	$\mathcal{O}(p)$	$\mathcal{O}(\frac{\kappa\tilde{\kappa}}{1-\varepsilon} \ln \frac{1}{\varsigma})$	Theorem 3
Alg 2 with (19)	$\tilde{\mathcal{O}}(\frac{G^2\tilde{\kappa}^2}{\varsigma\gamma} p)$	$\tilde{\mathcal{O}}(\frac{\kappa_1}{\varepsilon^2} p)$	$\tilde{\mathcal{O}}(\sqrt{\frac{\tilde{\kappa}}{1-\varepsilon}})$	$\mathcal{O}(\frac{\kappa\tilde{\kappa}}{1-\varepsilon} \ln \frac{1}{\varsigma})$	Theorem 4

Table 4 Complexity comparison of various Newton-type methods for unconstrained version of (1) to achieve sub-optimality $\|\mathbf{x}^{(k)}) - \mathbf{x}^\star\| \leq \varsigma$ for some $\varsigma \leq 1$, assuming $\mathbf{x}^{(0)}$ is close enough to $\mathbf{x}^\star$. For the local convergence rate as in (23) and (27), we set $\rho = 1/e$, where e is the Euler's number. The notation $\tilde{\mathcal{O}}$ implies hidden logarithmic factors, e.g., $\ln(\kappa)$, $\ln(\tilde{\kappa})$, $\ln(p)$, and $\ln(1/\delta)$. Constants $\gamma, \kappa, \tilde{\kappa}$ and κ_1 are defined in Sect. 1.5. Also, ε is the Hessian approximation accuracy parameter from Lemma 2. For $G(\mathbf{x})$ in Lemma 3, we assume that we have a uniform estimate as $G(\mathbf{x}) \leq G < \infty$

Method	Evaluating $\mathbf{g}(\mathbf{x})$	Evaluating one Hessian-Vector Product, $H(\mathbf{x})\mathbf{v}$	# of Iterations of CG on (2a)	Iteration Complexity	Reference
Newton-Exact	$\mathcal{O}(np)$	$\mathcal{O}(np)$	$\mathcal{O}(p)$	$\mathcal{O}(\ln\ln \frac{1}{\varsigma})$	Folklore
Newton-CG	$\mathcal{O}(np)$	$\mathcal{O}(np)$	$\tilde{\mathcal{O}}(\sqrt{\kappa})$	$\mathcal{O}(\ln \frac{1}{\varsigma})$	Folklore
Alg 3 with (13a)	$\mathcal{O}(np)$	$\tilde{\mathcal{O}}(\frac{\kappa_1^2}{\varepsilon^2} p)$	$\mathcal{O}(p)$	$\mathcal{O}(\ln \frac{1}{\varsigma})$	Theorem 6
Alg 3 with (15)	$\mathcal{O}(np)$	$\tilde{\mathcal{O}}(\frac{\kappa_1^2}{\varepsilon^2} p)$	$\tilde{\mathcal{O}}(\sqrt{\frac{\tilde{\kappa}}{1-\varepsilon}})$	$\mathcal{O}(\ln \frac{1}{\varsigma})$	Theorem 9
Alg 5 with (18a)	$\tilde{\mathcal{O}}(\frac{G^2L^2}{(1-\varepsilon)^4\varsigma^2\gamma^4} p)$	$\tilde{\mathcal{O}}(\frac{\kappa_1^2}{\varepsilon^2} p)$	$\mathcal{O}(p)$	$\mathcal{O}(\ln \frac{1}{\varsigma})$	Theorem 11
Alg 5 with (19)	$\tilde{\mathcal{O}}(\frac{G^2L^2}{(1-\varepsilon)^4\varsigma^2\gamma^4} p)$	$\tilde{\mathcal{O}}(\frac{\kappa_1^2}{\varepsilon^2} p)$	$\tilde{\mathcal{O}}(\sqrt{\frac{\tilde{\kappa}}{1-\varepsilon}})$	$\mathcal{O}(\ln \frac{1}{\varsigma})$	Theorem 12

assume that the cost of one Hessian-vector product is of the same order as evaluating a gradient, e.g, [20,34–36,42]. From Tables 3 and 4, the overall worst-case running-time of an algorithm to achieve the prescribed sub-optimality is estimated as $\big($Column #2 + Column #3 $\times$ Column #4$\big)$ $\times$ Column #5. We remind that the complexity results for the proposed algorithms in Tables 3 and 4 are given assuming that the underlying probabilistic events occur successfully over the required finite number of iterations to achieve the desired sub-optimality.

In Tables 3 and 4, ε is the Hessian approximation accuracy parameter from Lemmas 1 and 2. As for the overall failure probability, recall that in order to get an accumulative success probability of $1 - \delta_0$ for $k \in \mathcal{O}(T)$ iterations, the per-iteration failure probability is set as $\delta = 1 - \sqrt[T]{(1 - \delta_0)} \in \Omega(\delta_0/T)$. Since the overall iteration complexity is affected by ε (see Table 3), the per-iteration failure probability, δ, should also be chosen in relation to ε (of course, the overall failure probability, δ_0, can be chosen arbitrarily). However, since this dependence manifest itself only logarithmically, it is of negligible consequence in overall complexity.

Here, we only compare worst-case complexities of the algorithms studied in this paper with their deterministic counterparts, i.e., Newton's method; see e.g., [5,42] for tables which include complexities of other methods. We note that these complexities are, not only, for worst-case analysis, but also they are very pessimistic. For example, the worst-case complexity required to incorporate gradient sub-sampling might give the impression that such sampling is advantageous merely in some marginal cases. However, this unfortunate impression is a by-product of our analysis more so than it is an inherent property of an algorithm that incorporates gradient sampling. In this light, any conclusions from these tables should be made with great care.

Table 3 gives complexities involved in various algorithms for achieving sub-optimality in objective value, i.e., $F(\mathbf{x}^{(k)}) - F(\mathbf{x}^\star) \leq \varsigma$ for some $\varsigma \leq 1$. We consider complexity of iterations with the worst-case step-size, α_k, as prescribed by the corresponding theorem, which alleviates the need to perform line-search. One can make several observations regarding Table 3. As expected, it is advisable to perform gradient and Hessian sub-sampling only when $n \gg 1$; Table 3, very pessimistically, suggests that gradient and Hessian sub-sampling offer computational savings when $n \geq G^2\tilde{\kappa}^2/(\varsigma\gamma)$ and $n \geq \kappa_1\varepsilon^{-2}$, respectively. Also, if $n \in \mathcal{O}(\kappa_1\varepsilon^{-2}p)$, then one can consider using the full gradient. Notice that, as expected, gradient sampling complexity depends on the sub-optimality parameter ς. Uniform sampling gives similar worst-case iteration complexity as classical Newton's method only if $\tilde{\kappa} \in \mathcal{O}(\kappa)$, where κ and $\tilde{\kappa}$ are as in (8) and (9c), respectively. Otherwise, if problem admits favorable structure, non-uniform sampling can offer better iteration complexity than uniform sampling; see [42]. From Table 3, one can make comparisons among these methods in terms of total worst-case running-time. For example, if $n \leq \kappa_1\tilde{\kappa}/(\varepsilon^2(1-\varepsilon)\kappa)$, then the exact variant of Newton's method, has lower worst-case running-time than Algorithm 1 using (13a), i.e., Hessian sub-sampling might not help!

Table 4 gives similar results to achieve sub-optimality in iterates, i.e., $\|\mathbf{x}^{(k)} - \mathbf{x}^\star\| \leq \varsigma$ for some $\varsigma \leq 1$. For this. we assume that $\mathbf{x}^{(0)}$ is close enough to $\mathbf{x}^\star$, and the local convergence rates in (23) and (27) are set to $\rho = 1/e$, where e is the Euler's number. Although, Newton-CG can be made super-linearly convergent [17], in Table 4, for simplicity, we also do not consider super-linearly convergent algorithms. As a result, computational costs of Algorithms 4 is not considered. In Step 5 of Algorithm 5, the gradient accuracy is increased at every iteration. Hence, to calculate the gradient sampling complexity in Table 4, we consider the accuracy at the final iteration, i.e., $\varepsilon_2^{(k)}$ for $k \in \mathcal{O}(\ln(1/\varsigma))$. From Table 4, one can also make similar observations.

In Table 3, there is a noteworthy trade-off in choosing ε in terms of its effect on sampling and iteration complexities. Indeed, smaller ε yields larger samples but this,

in turn, not only implies fewer iterations of CG (if (2a) is solved inexactly), but also it involves fewer overall iterations to achieve a desired sub-optimality. More subtly, if the distribution of Hessians are very skewed and we perform sampling without replacement, decreasing ε would also decrease $\tilde{\kappa}$, which helps with many aspects of underlying complexities. For local convergence, however, as indicated by Table 4, as long as ε is chosen small enough (so we can appeal to the corresponding theorems), overall iteration complexity is unaffected by ε (this is indeed implied by the problem-independent local rates); although there is still a trade-off between sampling complexity and CG iterations.

From Tables 3 and 4, it is also clear that in the absence of a good preconditioner, if $\tilde{\kappa} \geq (1-\varepsilon)p^2$, solving (2a) exactly can be potentially more efficient than resorting to any inexact method.

4 Conclusions and further thoughts

Our primary objective here was to contribute in painting a more complete picture of the theory of sub-sampled Newton-type algorithms. For that, we considered large-scale optimization problems of the form (1) where $n, p \gg 1$, and we theoretically studied the global as well as local convergence behavior of Newton-type algorithms, where the Hessian and/or gradient are sub-sampled. We studied sub-sampling strategies to obtain an algorithm which enjoys desirable theoretical properties, in that, not only it is globally convergent, but also it enjoys a fast and problem-independent local convergence rate. We also showed that when the sub-problem is solved approximately to a milder inexactness tolerance than what is found in the similar literature, we can maintain the convergence properties of the methods with exact updates.

An important distinction of the "high-probability" style of analysis from classical convergence results is that, here, no sensible statement about the *asymptotic* convergence of an *infinite* sequence of these random iterates can be made. More specifically, it is clear that the overall success probability for T iterations, each having failure probability of δ is $(1-\delta)^k$. Now for an infinite number of iterations, i.e., $k \to \infty$, this probability goes to zero, implying that, regardless of how small δ is chosen, at some point along the way, the Hessian and/or gradient approximations fail to deliver the desired estimates. Arguably, this can be regarded as a disadvantage for such style of analysis. However, in practice, one almost always terminates the iterations either if a certain algorithmic condition is met or after a pre-prescribed maximum number of iterations is reached. In this case, the overall failure probability can be set as small as desired to fit one's required level of confidence.

Even in a finite number of iterations, one might still wonder what can happen when, at any one iteration, sub-sampled approximations fail to deliver the required properties, i.e., when the "good" probabilistic events do not occur. After all, regardless of how small δ is, there is always a positive probability that "bad" events happen. Fortunately, the algorithms incorporating line-search are inherently resilient to misestimation of the Hessian. More specifically, if at some point the sub-sampled Hessian fails to satisfy the invertibility promised by Lemma 1, then Algorithms 1 or 2 do not fail, e.g., do not diverge. In such as an unfortunate situation where the sub-sampled Hessian

contains a non-trivial null-space, (augmented) CG algorithm [23,24] can reliably be used to either find a solution of the underlying linear system (a solution exists when the gradient is orthogonal to the null space of Hessian) or the corresponding pseudo-inverse solution. In either case, since the sub-sampled matrix is positive semi-definite, even if the obtained direction cannot not yield a sufficient decrease in the objective function, it will certainly not cause an increase, and with high probability, in the very next iteration, the algorithm will recover from such a stall. Misestimation of the gradient at any step, however, can be quite serious and result in divergence. In this case, additional safeguards needs to be put in place to avoid such unwanted behavior, e.g., restarting from the previous iterate if the objective is to increase. Investigating alternative strategies to line-search, e.g., trust-region [13], which can potentially provide more robustness to such misestimations is an interesting direction for future research.

Finally, an alternative to the high-probability style of analysis considered here, is the convergence in expectation, e.g., [5], which has the advantage of giving asymptotic results. However, convergence in expectation has its own disadvantages among which, to interpret the results, it is implied that one must run the algorithm (possibly infinitely) many times, and then average all the outcomes. In other words, unlike high-probability analysis, convergence in expectation provides no statement on the results of individual runs. We believe that these two styles of analysis are, not only, related in many ways, but also they are complementary and, together, can paint a much more complete picture of the behavior of these randomized algorithms.

References

1. Agarwal, N., Bullins, B., Hazan, E.: Second order stochastic optimization in linear time. arXiv preprint arXiv:1602.03943 (2016)
2. Berahas, A.S., Bollapragada, R., Nocedal, J.: An investigation of Newton-sketch and subsampled Newton methods. arXiv preprint arXiv:1705.06211 (2017)
3. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific, Belmont (1999)
4. Bertsekas, D.P.: Convex Optimization Theory. Athena Scientific, Belmont (2009)
5. Bollapragada, R., Byrd, R., Nocedal, J.: Exact and inexact subsampled Newton methods for optimization. arXiv preprint arXiv:1609.08502 (2016). **(To appear in IMA Journal of Numerical Analysis)**
6. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
7. Byrd, R.H., Chin, G.M., Neveitt, W., Nocedal, J.: On the use of stochastic Hessian information in optimization methods for machine learning. SIAM J. Optim. **21**(3), 977–995 (2011)
8. Byrd, R.H., Chin, G.M., Nocedal, J., Wu, Y.: Sample size selection in optimization methods for machine learning. Math. Program. **134**(1), 127–155 (2012)
9. Byrd, R.H., Nocedal, J., Oztoprak, F.: An inexact successive quadratic approximation method for convex L-1 regularized optimization. arXiv preprint arXiv:1309.3529 (2013)
10. Cartis, C., Gould, N.I.M., Toint, PhL: On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems. SIAM J. Optim. **20**(6), 2833–2852 (2010)
11. Cartis, C., Gould, N.I.M., Toint, Ph.L.: An example of slow convergence for Newton's method on a function with globally Lipschitz continuous Hessian. Technical report, ERGO 13-008, School of Mathematics, Edinburgh University (2013)
12. Cohen, M.B., Lee, Y.T., Musco, C., Musco, C., Peng, R., Sidford, A.: Uniform sampling for matrix approximation. In: Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science, pp. 181–190. ACM (2015)
13. Conn, A.R., Gould, N.I.M., Toint, PhL: Trust Region Methods. SIAM, Philadelphia (2000)
14. Dembo, R.S., Eisenstat, S.C., Steihaug, T.: Inexact Newton methods. SIAM J. Numer. Anal. **19**(2), 400–408 (1982)

15. Drineas, P., Kannan, R., Mahoney, M.W.: Fast Monte Carlo algorithms for matrices I: approximating matrix multiplication. SIAM J. Comput. **36**(1), 132–157 (2006)
16. Eisen, M., Mokhtari, A., Ribeiro, A.: Large scale empirical risk minimization via truncated adaptive Newton method. In: Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics, PMLR,vol. 84, pp. 1447–1455 (2018)
17. Eisenstat, S.C., Walker, H.F.: Choosing the forcing terms in an inexact Newton method. SIAM J. Sci. Comput. **17**(1), 16–32 (1996)
18. Erdogdu, M.A., Montanari, A.: Convergence rates of sub-sampled Newton methods. Adv. Neural Inf. Process. Syst. **28**, 3034–3042 (2015)
19. Friedlander, M.P., Schmidt, M.: Hybrid deterministic-stochastic methods for data fitting. SIAM J. Sci. Comput. **34**(3), A1380–A1405 (2012)
20. Griewank, A.: Some Bounds on the Complexity of Gradients, Jacobians, and Hessians. Complexity in Nonlinear Optimization, pp. 128–161. World Scientific Publisher, Singapore (1993)
21. Gross, D., Nesme, V.: Note on sampling without replacing from a finite collection of matrices. arXiv preprint arXiv:1001.2738 (2010)
22. Haber, E., Chung, M.: Simultaneous source for non-uniform data variance and missing data. arXiv preprint arXiv:1404.5254 (2014)
23. Hestenes, M.R.: Pseudoinversus and conjugate gradients. Commun. ACM **18**(1), 40–43 (1975)
24. Hestenes, M.R.: Conjugate Direction Methods in optimization, vol. 12. Springer, Berlin (2012)
25. Holodnak, J.T., Ipsen, I.C.: Randomized approximation of the Gram matrix: exact computation and probabilistic bounds. SIAM J. Matrix Anal. Appl. **36**(1), 110–137 (2015)
26. Lee, J.D., Sun, Y., Saunders, M.A.: Proximal Newton-type methods for minimizing composite functions. SIAM J. Optim. **24**(3), 1420–1443 (2014)
27. Liu, X., Hsieh, C.J., Lee, J.D., Sun, Y.: An inexact subsampled proximal Newton-type method for large-scale machine learning. arXiv preprint arXiv:1708.08552 (2017)
28. Mahoney, M.W.: Randomized algorithms for matrices and data. Found. Trends® Mach. Learn. **3**(2), 123–224 (2011)
29. Martens, J.: Deep learning via Hessian-free optimization. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp. 735–742 (2010)
30. McCullagh, P., Nelder, J.A.: Generalized Linear Models, vol. 37. CRC Press, Boca Raton (1989)
31. Mutnỳ, M.: Stochastic second-order optimization via von Neumann series. arXiv preprint arXiv:1612.04694 (2016)
32. Nesterov, Y.: Introductory Lectures on Convex Optimization, vol. 87. Springer, Berlin (2004)
33. Nesterov, Y., Polyak, B.T.: Cubic regularization of Newton method and its global performance. Math. Program. **108**(1), 177–205 (2006)
34. Pearlmutter, B.A.: Fast exact multiplication by the Hessian. Neural Comput. **6**(1), 147–160 (1994)
35. Pilanci, M., Wainwright, M.J.: Newton sketch: a linear-time optimization algorithm with linear-quadratic convergence. arXiv preprint arXiv:1505.02250 (2015)
36. Roosta-Khorasani, F., van den Doel, K., Ascher, U.: Stochastic algorithms for inverse problems involving PDEs and many measurements. SIAM J. Sci. Comput. **36**(5), S3–S22 (2014). https://doi.org/10.1137/130922756
37. Roosta-Khorasani, F., Székely, G.J., Ascher, U.: Assessing stochastic algorithms for large scale nonlinear least squares problems using extremal probabilities of linear combinations of gamma random variables. SIAM/ASA J. Uncertain. Quantif. **3**(1), 61–90 (2015)
38. Tropp, J.A.: Improved analysis of the subsampled randomized Hadamard transform. Adv. Adapt. Data Anal. **3**(01n02), 115–126 (2011)
39. Tropp, J.A.: User-friendly tail bounds for sums of random matrices. Found. Comput. Math. **12**(4), 389–434 (2012)
40. Wang, C.C., Huang, C.H., Lin, C.J.: Subsampled Hessian Newton methods for supervised learning. Neural Comput. **27**, 1766–1795 (2015)
41. Xu, P., Roosta-Khorasan, F., Mahoney, M.W.: Second-order optimization for non-convex machine learning: an empirical study. arXiv preprint arXiv:1708.07827 (2017)
42. Xu, P., Yang, J., Roosta-Khorasani, F., Ré, C., Mahoney, M.W.: Sub-sampled Newton methods with non-uniform sampling. In: Advances in Neural Information Processing Systems 29, pp. 3000–3008 (2016)