Sketched Newton–Raphson

Rui Yuan, Alessandro Lazaric, Robert M Gower

SIAM J. OPTIM.
Vol. 32, No. 3, pp. 1555–1583

SKETCHED NEWTON–RAPHSON*

RUI YUAN†, ALESSANDRO LAZARIC‡, AND ROBERT M. GOWER§

Abstract. We propose a new globally convergent stochastic second-order method. Our starting point is the development of a new sketched Newton–Raphson (SNR) method for solving large scale nonlinear equations of the form $F(x) = 0$ with $F : \mathbb{R}^p \to \mathbb{R}^m$. We then show how to design several stochastic second-order optimization methods by rewriting the optimization problem of interest as a system of nonlinear equations and applying SNR. For instance, by applying SNR to find a stationary point of a generalized linear model, we derive completely new and scalable stochastic second-order methods. We show that the resulting method is very competitive as compared to state-of-the-art variance reduced methods. Furthermore, using a variable splitting trick, we also show that the *stochastic Newton method* (SNM) is a special case of SNR and use this connection to establish the first global convergence theory of SNM. We establish the global convergence of SNR by showing that it is a variant of the online stochastic gradient descent (SGD) method, and then leveraging proof techniques of SGD. As a special case, our theory also provides a new global convergence theory for the original Newton–Raphson method under strictly weaker assumptions as compared to the classic monotone convergence theory.

Key words. nonlinear systems, stochastic methods, iterative methods, stochastic Newton method, randomized Kaczmarz

MSC codes. 90C53, 74S60, 90C06, 62L20, 68W20, 15B52, 65Y20, 68W40

DOI. 10.1137/21M139788X

1. Introduction.

One of the fundamental problems in numerical computing is to find roots of systems of nonlinear equations such as

$$F(x) = 0, \tag{1.1}$$

where $F : \mathbb{R}^p \to \mathbb{R}^m$. We assume throughout that $F : \mathbb{R}^p \to \mathbb{R}^m$ is continuously differentiable and that there exists a solution to (1.1), as follows.

Assumption 1.1. $\exists x^* \in \mathbb{R}^p$ such that $F(x^*) = 0$.

Our main interest here is to solve nonlinear minimization problems in machine learning. Most convex optimization problems such as those arising from training a generalized linear model (GLM), can be rewritten as a system of nonlinear equations (1.1) either by manipulating the stationarity conditions or as the Karush–Kuhn–Tucker equations.[1] The building block of many iterative methods for solving nonlinear equations is the Newton–Raphson (NR) method given by

$$x^{k+1} = x^k - \gamma \left(DF(x^k)^\top\right)^\dagger F(x^k) \tag{1.2}$$

at the kth iteration, where $DF(x) \overset{\text{def}}{=} [\nabla F_1(x) \ \cdots \ \nabla F_m(x)] \in \mathbb{R}^{p\times m}$ is the transpose of the Jacobian matrix of F at x, $(DF(x^k)^\top)^\dagger$ is the Moore–Penrose pseudoinverse of $DF(x^k)^\top$, and $\gamma > 0$ is the stepsize.

*Received by the editors February 9, 2021; accepted for publication (in revised form) May 4, 2022; published electronically July 13, 2022.
https://doi.org/10.1137/21M139788X

†Meta AI, LTCI, Télécom Paris, and Institut Polytechnique de Paris, Paris, France (ruiyuan@fb.com).
‡Meta AI, Paris, France (lazaric@fb.com).
§CCM, Flatiron Institute, New York, NY 10010 USA (gowerrobert@gmail.com). Part of this work was done when the author was affiliated with Télécom Paris and Meta AI.

[1] Under suitable constraint qualifications [43].

The NR method is at the heart of many commercial solvers for nonlinear equations [44]. The success of NR can be partially explained by its invariance to affine coordinate transformations, which in turn means that the user does not need to tune any parameters (standard NR sets $\gamma = 1$). The downside of NR is that we need to solve a linear least squares problem given in (1.2) which costs $\mathcal{O}(\min\{pm^2, mp^2\})$ when using a direct solver. When both p and m are large, this cost per iteration is prohibitive. Here we develop a randomized NR method based on the sketch-and-project technique [22] which can be applied in large scale, as we show in our experiments.

1.1. The sketched Newton–Raphson method. Our method relies on using *sketching matrices* to reduce the dimension of the Newton system.

DEFINITION 1.2. *The sketching matrix $\mathbf{S} \in \mathbb{R}^{m\times\tau}$ is a random matrix sampled from a distribution $\mathcal{D}$, where $\tau \in \mathbb{N}$ is the sketch size. We use $\mathbf{S}_k \in \mathbb{R}^{m\times\tau}$ to denote a sketching matrix sampled from a distribution $\mathcal{D}_{x^k}$ that can depend on the iterate x^k.*

By sampling a sketching matrix $\mathbf{S}_k \sim \mathcal{D}_{x^k}$ at kth iteration, we *sketch* (row compress) NR update and compute an approximate *sketched Newton–Raphson* (SNR) step; see (1.3) in Algorithm 1. We use $\mathcal{D}_x$ to denote a distribution that depends on x, and allow the distribution of the sketching matrix to change from one iteration to the next.

Algorithm 1. SNR: Sketched Newton–Raphson

parameters: $\mathcal{D}$ = distribution of sketching matrix; stepsize parameter $\gamma > 0$
initialization: Choose $x^0 \in \mathbb{R}^p$
for $k = 0, 1, \cdots$ **do**
  Sample a fresh sketching matrix: $\mathbf{S}_k \sim \mathcal{D}_{x^k}$

$$x^{k+1} = x^k - \gamma DF(x^k)\mathbf{S}_k \left(\mathbf{S}_k^\top DF(x^k)^\top DF(x^k)\mathbf{S}_k\right)^\dagger \mathbf{S}_k^\top F(x^k) \tag{1.3}$$

return: last iterate x^k

Because the sketching matrix $\mathbf{S}_k$ has τ columns, the dominating costs of computing the SNR step (1.3) are linear in p and m. In particular, $DF(x^k)\mathbf{S}_k \in \mathbb{R}^{p\times\tau}$ can be computed by using τ directional derivatives of $F(x^k)$, one for each column of $\mathbf{S}_k$. Using automatic differentiation [11], these directional derivatives cost τ evaluations of the function $F(x)$. Furthermore, it costs $\mathcal{O}(p\tau^2)$ to form the linear system in (1.3) of Algorithm 1 by using the computed matrix $DF(x^k)\mathbf{S}_k$ and $\mathcal{O}(\tau^3)$ to solve it, respectively. Finally the matrix vector product $\mathbf{S}_k^\top F(x^k)$ costs $\mathcal{O}(m\tau)$. Thus, without making any further assumptions to the structure of F or the sketching matrix, the total cost in terms of operations of the update (1.3) is given by

$$\text{Cost(update (1.3))} = \mathcal{O}\left((\text{eval}(F) + m) \times \tau + p\tau^2 + \tau^3\right). \tag{1.4}$$

Thus Algorithm 1 can be applied when both p and m are large and τ is relatively small.

1.2. Background and contributions.

(a) *Stochastic second-order methods.* There is now a concerted effort to develop efficient second-order methods for solving high dimensional and stochastic optimization problems in machine learning. Most recently developed Newton methods fall into one of two categories: *subsampling* and *dimension reduction*. The subsampling methods [17, 48, 31, 7, 60] and [1, 45][2] use minibatches to compute an approximate

[2]Newton sketch [45] and LiSSa [1] use subsampling to build an estimate of the Hessian but require a full gradient evaluation. As such, these methods are not efficient for very large n.

Newton direction. Though these methods can handle a large number of *data points* (n), they do not scale well in the number of *features* (d). On the other hand, second-order methods based on dimension reduction techniques such as [19] apply Newton's method over a subspace of the features, and as such do not scale well in the number of data points. Sketching has also been used to develop second-order methods in the online learning setting [24, 36, 8] and quasi-Newton methods [20].

Contributions. We propose a new family of stochastic second-order methods called SNR. Each choice of the sketching distribution and nonlinear equations used to describe the stationarity conditions leads to a particular algorithm. For instance, we show that a nonlinear variant of the Kaczmarz method is a special case of SNR. We also show that the subsampling based stochastic Newton method (SNM) [32] is a special case of SNR. We provide a concise global convergence theory that when specialized to SNM gives its first global convergence result. Furthermore, the convergence theory of SNR allows for any sketch size, which translates to any minibatch size for the nonlinear Kaczmarz and SNM. In contrast, excluding SNM, the subsampled based Newton methods [17, 48, 31, 7, 60, 1, 45] rely on high probability bounds that in turn require large minibatch sizes.[3] We detail the nonlinear Kaczmarz method in section 6 and the connection with SNM in section 7.

(b) *New method for GLMs.* There exist several specialized methods for solving GLMs, including variance reduced gradient methods such as SAG/SAGA [49, 13] and SVRG [26], and methods based on dual coordinate ascent like SDCA [51], dual free SDCA (dfSDCA) [50], and Quartz [46].

Contributions. We develop a specialized variant of SNR for GLMs in section 8. Our resulting method scales linearly in the number of dimensions d and the number of data points n and has the same cost as stochastic gradient descent (SGD) per iteration in average. We show in experiments that our method is very competitive as compared to state-of-the-art variance reduced methods for GLMs.

(c) *Viewpoints of (sketched) Newton–Raphson.* We show in section 3 that SNR can be seen as SGD applied to an equivalent reformulation of our original problem. We will show that this reformulation is *always* a smooth and interpolated function [37, 53]. These gratuitous properties allow us to establish a simple global convergence theory by only assuming that the reformulation is a *star-convex* function: a class of nonconvex functions that include convexity as a special case [42, 33, 63, 25].

(d) *Classic convergence theory of Newton–Raphson.* The better known convergence theorems for NR (the Newton–Kantorovich–Mysovskikh theorems) only guarantee local or semilocal convergence [28, 44]. To guarantee global convergence of NR, we often need an additional globalization strategy, such as damping sequences or adaptive trust-region methods [12, 35, 15, 29], continuation schemes such as interior point methods [41, 57], and more recently cubic regularization [32, 42, 9]. Globalization strategies are used in conjunction with other second-order methods, such as inexact Newton backtracking type methods [5, 3], Gauss–Newton or Levenberg–Marquardt type methods [62, 61, 58], and quasi-Newton methods [58].[4] The only global convergence theory that does not rely on such a globalization strategy requires strong assumptions on $F(x)$, such as in the monotone convergence theory (MCT) [15].

Contributions. We show in section 5.3 that our main theorem specialized to the standard NR method guarantees a global convergence under *strictly* less assumptions

[3] The batch sizes in these methods scale proportional to a condition number [1] or ϵ^{-1} where ϵ is the desired tolerance.

[4] A recent paper [18] shows that quasi-Newton converges globally for self-concordant functions without globalization strategy.

as compared to the MCT, albeit under a different stepsize. Indeed, MCT holds for stepsize equal to one ($\gamma = 1$) and our theory holds for stepsizes less than one ($\gamma < 1$).

Furthermore, we give an explicit sublinear $O(1/k)$ convergence rate, as opposed to only an asymptotic convergence in MCT. This appears not to have been known before since, as stated by [15] w.r.t. the NR method, "Not even an a-priori estimation for the number of iterations needed to achieve a prescribed accuracy *may* be possible." We show that it is possible by monitoring which iterate achieves the best loss (suboptimality).

(e) *Sketch-and-project.* The sketch-and-project method was originally introduced for solving linear systems in [22, 23], where it was also proven to converge linearly and globally. In [47], the authors then go on to show that the sketch-and-project method is in fact SGD applied to a particular reformulation of the linear system.

Contributions. It is this SGD viewpoint in the linear setting [47] that we extend to the nonlinear setting. Thus the SNR algorithm and our theory are generalizations of the original sketch-and-project method for solving linear equations to solving nonlinear equations, thus greatly expanding the scope of applications of these techniques.

1.3. Notation. In calculating an update of SNR (1.3) and analyzing SNR, the following random matrix is key:

$$\mathbf{H}_{\mathbf{S}}(x) \overset{\text{def}}{=} \mathbf{S}\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)^\dagger \mathbf{S}^\top. \tag{1.5}$$

The sketching matrix $\mathbf{S}$ in (1.5) is sampled from a distribution $\mathcal{D}_x$ and $\mathbf{H}_{\mathbf{S}}(x) \in \mathbb{R}^{m\times m}$ is a random matrix that depends on x. We use $\mathbf{I}_p \in \mathbb{R}^{p\times p}$ to denote the identity matrix of dimension p and use $\|x\|_{\mathbf{M}} \overset{\text{def}}{=} \sqrt{x^\top \mathbf{M} x}$ to denote the seminorm of $x \in \mathbb{R}^p$ induced by a symmetric positive semidefinite matrix $\mathbf{M} \in \mathbb{R}^{p\times p}$. Notice that $\|x\|_{\mathbf{M}}$ is not necessarily a norm as $\mathbf{M}$ is allowed to be noninvertible. We handle this with care in our forthcoming analysis. We also define the following sets: $F(U) = \{F(x) \mid x \in U\}$ for a given set $U \subset \mathbb{R}^p$; $W^\perp = \{v \mid \langle u, v\rangle = 0 \text{ for all } u \in W\}$ to denote the orthogonal complement of a subspace W; $\mathbf{Im}(\mathbf{M}) = \{y \in \mathbb{R}^m \mid \exists x \in \mathbb{R}^p \text{ s.t. } \mathbf{M}x = y\}$ to denote the image space; and $\mathbf{Ker}(\mathbf{M}) = \{x \in \mathbb{R}^p \mid \mathbf{M}x = 0\}$ to denote the null space of a matrix $\mathbf{M} \in \mathbb{R}^{m\times p}$. If $\mathbf{M}$ is a random matrix sampled from a certain distribution $\mathcal{D}$, we use $\mathbb{E}_{\mathbf{S}\sim\mathcal{D}}[\mathbf{M}] = \int_{\mathbf{M}} \mathbf{M} d\mathbb{P}_{\mathcal{D}}(\mathbf{M})$ to denote the expectation of the random matrix. We omit the notation of the distribution $\mathcal{D}$, i.e., $\mathbb{E}[\mathbf{M}]$, when the random source is clear. In particular, when $\mathbf{M}$ is sampled from a discrete distribution with $r \in \mathbb{N}$ s.t. $\mathbb{P}[\mathbf{M} = \mathbf{M}_i] = p_i > 0$, for $i = 1, \dots, r$ and $\sum_{i=1}^r p_i = 1$, then $\mathbb{E}[\mathbf{M}] = \sum_{i=1}^r p_i \mathbf{M}_i$.

1.4. Sketching matrices. Here we provide examples of sketching matrices that can be used in conjunction with SNR. We point the reader to [56] for a detailed exposure and introduction. The most straightforward sketch is given by the Gaussian sketch where every coordinate $\mathbf{S}_{ij}$ of the sketch $\mathbf{S} \in \mathbb{R}^{m\times\tau}$ is sampled independent and identically distributed according to a Gaussian distribution with $\mathbf{S}_{ij} \sim \mathcal{N}(0, \frac{1}{\tau})$ for $i = 1, \dots, m$ and $j = 1, \dots, \tau$. The sketch we mostly use here is the uniform subsampling sketch, whereby

$$\mathbb{P}[\mathbf{S} = \mathbf{I}_C] \;=\; \frac{1}{\binom{m}{\tau}} \qquad \text{for all set } C \subset \{1, \dots, m\} \text{ s.t. } |C| = \tau, \tag{1.6}$$

where $\mathbf{I}_C \in \mathbb{R}^{m\times\tau}$ denotes the concatenation of the columns of the identity matrix $\mathbf{I}_m$ indexed in the set C. More sophisticated sketches that are able to make use of fast Fourier type routines include the random orthogonal sketches [45, 2]. We will not cover random orthogonal sketches here since these sketches are fast when applied only once to a fixed matrix $\mathbf{M}$, as opposed to being resampled at every iteration .

2. The sketch-and-project viewpoint. The viewpoint that motivated the development of Algorithm 1 was the following iterative *sketch-and-project* method applied to the Newton system. For this viewpoint, we assume as follows.

Assumption 2.1. $F(x) \in \mathbf{Im}\left(DF(x)^\top\right)$ for all $x \in \mathbb{R}^p$.

This assumption guarantees that there exists a solution to the Newton system in (1.2). Indeed, we can now rewrite the NR method (1.2) as a projection of the previous iterate x^k onto the solution space of a Newton system

$$x^{k+1} = \operatorname{argmin}_{x\in\mathbb{R}^p} \|x - x^k\|^2 \quad \text{s.t.} \quad DF(x^k)^\top(x - x^k) = -\gamma F(x^k). \tag{2.1}$$

Since this is costly to solve when $DF(x^k)$ has many rows and columns, we *sketch* the Newton system. That is, we apply a random row compression to the Newton system using the sketching matrix $\mathbf{S}_k^\top \in \mathbb{R}^{\tau\times m}$ and then project the previous iterates x^k onto this *sketched* system as follows:

$$x^{k+1} = \operatorname{argmin}_{x\in\mathbb{R}^p} \|x - x^k\|^2 \quad \text{s.t.} \quad \mathbf{S}_k^\top DF(x^k)^\top(x - x^k) = -\gamma \mathbf{S}_k^\top F(x^k). \tag{2.2}$$

That is, x^{k+1} is the projection of x^k onto the solution space of the sketched Newton system. This viewpoint was our motivation for developing the SNR method. Next we establish our core theory. The theory does not rely on the assumption $F(x) \in \mathbf{Im}\left(DF(x)^\top\right)$, though this assumption will appear again in several specialized corollaries. Without this assumption, we can still interpret the Newton step (1.2) as the least squares solution of the linear system (2.1), as we show next.

3. Reformulation as stochastic gradient descent. Our insight into interpreting and analyzing the SNR in Algorithm 1 is through its connection to the SGD. Next, we show how SNR can be seen as SGD applied to a sequence of equivalent reformulations of (1.1). Each reformulation is given by a vector $y \in \mathbb{R}^p$ and the following minimization problem:

$$\min_{x\in\mathbb{R}^p} \mathbb{E}_{\mathbf{S}\sim\mathcal{D}_y}\left[\frac{1}{2}\|F(x)\|^2_{\mathbf{H}_\mathbf{S}(y)}\right], \tag{3.1}$$

where $\mathbf{H}_\mathbf{S}(y)$ is defined in (1.5). To abbreviate notation, let

$$f_{\mathbf{S},y}(x) \overset{\text{def}}{=} \frac{1}{2}\|F(x)\|^2_{\mathbf{H}_\mathbf{S}(y)} \quad \text{and} \quad f_y(x) \overset{\text{def}}{=} \mathbb{E}\left[f_{\mathbf{S},y}(x)\right] = \frac{1}{2}\|F(x)\|^2_{\mathbb{E}[\mathbf{H}_\mathbf{S}(y)]}. \tag{3.2}$$

Every solution $x^* \in \mathbb{R}^p$ to (1.1) is a solution to (3.1), since $f_y(x)$ is nonnegative for every $x \in \mathbb{R}^p$ and $f_y(x^*) = 0$ is thus a global minima. With an extra assumption, we can show that every solution to (3.1) is also a solution to (1.1) in the following lemma.

LEMMA 3.1. *If Assumption* 1.1 *holds and the* reformulation *assumption*

$$F(\mathbb{R}^p) \cap \mathbf{Ker}\left(\mathbb{E}_{\mathbf{S}\sim\mathcal{D}_y}\left[\mathbf{H}_\mathbf{S}(y)\right]\right) = \{0\} \quad \textit{for all } y \in \mathbb{R}^p \tag{3.3}$$

holds, then $\operatorname{argmin}_{x\in\mathbb{R}^p} f_y(x) = \{x \mid F(x) = 0\}$ *for every* $y \in \mathbb{R}^p$.

Proof. Let $y \in \mathbb{R}^p$. Previously, we showed that $\{x \mid F(x) = 0\} \subset \operatorname{argmin}_{x\in\mathbb{R}^p} f_y(x)$. Now let $x^* \in \operatorname{argmin}_{x\in\mathbb{R}^p} f_y(x)$. By Assumption 1.1, we know that any global minimizer x^* of $f_y(x)$ must be s.t. $f_y(x^*) = 0$. This implies that $F(x^*) \in \mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H}_\mathbf{S}(y)\right]\right)$ since $\mathbb{E}\left[\mathbf{H}_\mathbf{S}(y)\right]$ is symmetric. However, $F(x^*) \in F(\mathbb{R}^p)$, and thus from (3.3), we have that $F(x^*) \in F(\mathbb{R}^p) \cap \mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H}_\mathbf{S}(y)\right]\right) = \{0\}$, which implies $F(x^*) = 0$. Thus, we have $\operatorname{argmin}_{x\in\mathbb{R}^p} f_y(x) \subset \{x \mid F(x) = 0\}$, which concludes the proof. ∎

Thus with the extra reformulation assumption in (3.3), we can now use any viable optimization method to solve (3.1) for any fixed $y \in \mathbb{R}^p$ and arrive at a solution to (1.1). In Lemma A.1, we give sufficient conditions on the sketching matrix and on the function $F(x)$ that guarantee (3.3) holds. We also show how (3.3) holds for our forthcoming examples in Appendix A as a direct consequence of Lemma A.1. However, (3.3) imposes for all $y \in \mathbb{R}^p$ which can sometimes be restrictive. In fact, we do *not need* for (3.1) to be equivalent to solving (1.1) for *every* $y \in \mathbb{R}^p$. Indeed, by carefully and iteratively updating y, we can solve (3.1) and obtain a solution to (1.1) *without* relying on (3.3). The trick here is to use an *online* SGD method for solving (3.1).

Since (3.1) is a stochastic optimization problem, SGD is a natural choice for solving (3.1). Let $\nabla f_{\mathbf{S},y}(x)$ denote the gradient of the function $f_{\mathbf{S},y}(\cdot)$ which is

$$\nabla f_{\mathbf{S},y}(x) = DF(x)\mathbf{H}_{\mathbf{S}}(y)F(x). \tag{3.4}$$

Since we are free to choose y, we allow y to *change* from one iteration to the next by setting $y = x^k$ at the start of the kth iteration. We can now take an SGD step by sampling $\mathbf{S}_k \sim \mathcal{D}_{x^k}$ at the kth iteration and updating

$$x^{k+1} = x^k - \gamma \nabla f_{\mathbf{S}_k,x^k}(x^k). \tag{3.5}$$

It is straightforward to verify that the SGD update (3.5) is exactly the same as the SNR update in (1.3).

The objective function $f_{\mathbf{S},y}(x)$ has many properties that makes it very favorable for optimization including the interpolation condition and a gratuitous smoothness property. Indeed, for any $x^* \in \mathbb{R}^p$ s.t. $F(x^*) = 0$, we have that the stochastic gradient is zero, i.e., $\nabla f_{\mathbf{S},y}(x^*) = 0$. This is known as the *interpolation condition*. When it occurs together with strong convexity, it is possible to shows that SGD converges linearly [53, 37]. We will also give a linear convergence result in section 4 by assuming that $f_y(x)$ is quasi-strongly convex. We detail the smoothness property next.

However, we need to be careful, since (3.5) is not a classic SGD method. In fact, from the kth iteration to the $(k+1)$th iteration, we change our objective function from $f_{x^k}(x)$ to $f_{x^{k+1}}(x)$ and the distribution from $\mathcal{D}_{x^k}$ to $\mathcal{D}_{x^{k+1}}$. Thus it is an online SGD. We handle this with care in our forthcoming convergence proofs.

4. Convergence theory. Using the viewpoint of SNR in section 3, we adapt proof techniques of SGD to establish the global convergence of SNR.

4.1. Smoothness property. In our upcoming proof, we rely on the following type of smoothness property thanks to our SGD reformulation (3.1).

LEMMA 4.1. *For every $x \in \mathbb{R}^p$ and any realization $\mathbf{S} \sim \mathcal{D}_x$ associated with any distribution $\mathcal{D}_x$,*

$$\frac{1}{2}\|\nabla f_{\mathbf{S},x}(x)\|^2 = f_{\mathbf{S},x}(x). \tag{4.1}$$

Proof. Turning to the definition of $f_{\mathbf{S},x}$ in (3.2), we have that

$$\begin{aligned}\|\nabla f_{\mathbf{S},x}(x)\|^2 &\overset{(3.4)}{=} \|DF(x)\mathbf{H}_{\mathbf{S}}(x)F(x)\|^2 = F(x)^\top \mathbf{H}_{\mathbf{S}}(x)^\top DF(x)^\top DF(x)\mathbf{H}_{\mathbf{S}}(x)F(x)\\ &= F(x)^\top \mathbf{H}_{\mathbf{S}}(x)F(x) = 2f_{\mathbf{S},x}(x),\end{aligned}$$

where we used the property $\mathbf{M}^\dagger\mathbf{M}\mathbf{M}^\dagger = \mathbf{M}^\dagger$ with $\mathbf{M} = \mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}$ to establish that $\mathbf{H}_{\mathbf{S}}(x)^\top DF(x)^\top DF(x)\mathbf{H}_{\mathbf{S}}(x) \overset{(1.5)}{=} \mathbf{H}_{\mathbf{S}}(x)$. ∎

This is not a standard smoothness property. Indeed, since $\nabla f_{\mathbf{S},x}(x^*) = 0$ and $f_x(x^*) = 0$, we have that (4.1) implies that $\|\nabla f_{\mathbf{S},x}(x) - \nabla f_{\mathbf{S},x}(x^*)\|^2 \leq 2(f_{\mathbf{S},x}(x) - f_{\mathbf{S},x}(x^*))$, which is usually a consequence of assuming that $f_{\mathbf{S},x}(x)$ is convex and 1-smooth (see Theorem 2.1.5 and equation (2.1.7) in [40]). Yet in our case, (4.1) is a direct consequence of the definition of $f_{\mathbf{S},x}$ as opposed to being an extra assumption. This gratuitous property will be key in establishing a global convergence result.

4.2. Convergence for star-convex. We use the shorthand $f_k(x) \overset{\text{def}}{=} f_{x^k}(x)$, $f_{\mathbf{S}_k,k} \overset{\text{def}}{=} f_{\mathbf{S}_k,x^k}$ and $\mathbb{E}_k[\cdot] \overset{\text{def}}{=} \mathbb{E}\left[\cdot \mid x^k\right]$. Here we establish the global convergence of SNR by supposing that f_k is *star-convex*, which is a large class of nonconvex functions that includes convexity as a special case [42, 33, 63, 25].

Assumption 4.2 (star-convexity). Let x^* satisfy Assumption 1.1, i.e., let x^* be a solution to (1.1). For every x^k given by Algorithm 1 with $k \in \mathbb{N}$, we have that

$$f_k(x^*) \geq f_k(x^k) + \left\langle \nabla f_k(x^k), x^* - x^k \right\rangle. \tag{4.2}$$

We now state our main theorem.

THEOREM 4.3. *Let x^* satisfy Assumption* 4.2. *If $0 < \gamma < 1$, then*

$$\mathbb{E}\left[\min_{t=0,\ldots,k-1} f_t(x^t)\right] \leq \frac{1}{k}\sum_{t=0}^{k-1} \mathbb{E}\left[f_t(x^t)\right] \leq \frac{1}{k}\frac{\|x^0 - x^*\|^2}{2\gamma(1-\gamma)}. \tag{4.3}$$

Written in terms of F and for $\gamma = 1/2$ the above gives

$$\mathbb{E}\left[\min_{t=0,\ldots,k-1} \|F(x^t)\|^2_{\mathbb{E}[\mathbf{H}_{\mathbf{S}}(x^t)]}\right] \quad \leq \quad \frac{4\|x^0 - x^*\|^2}{k}.$$

Besides, if the stochastic function $f_{\mathbf{S},x}(x)$ is star-convex along the iterates x^k, i.e.,

$$f_{\mathbf{S}_k,x^k}(x^*) \geq f_{\mathbf{S}_k,x^k}(x^k) + \left\langle \nabla f_{\mathbf{S}_k,x^k}(x^k), x^* - x^k \right\rangle \tag{4.4}$$

for all $\mathbf{S}_k \sim \mathcal{D}_{x^k}$, then the iterates x^k of SNR (1.3) *are bounded with*

$$\|x^k - x^*\| \leq \|x^0 - x^*\|. \tag{4.5}$$

Proof. Let $t \in \{0, \ldots, k-1\}$ and $\delta_t \overset{\text{def}}{=} x^t - x^*$. We have that

$$\begin{aligned}
\mathbb{E}_t\left[\|\delta_{t+1}\|^2\right] &\overset{(3.5)}{=} \mathbb{E}_t\left[\|x^t - \gamma\nabla f_{\mathbf{S}_t,t}(x^t) - x^*\|^2\right] \\
&= \|\delta_t\|^2 - 2\gamma\left\langle \delta_t, \nabla f_t(x^t)\right\rangle + \gamma^2\mathbb{E}_t\left[\|\nabla f_{\mathbf{S}_t,t}(x^t)\|^2\right] \\
&\overset{(4.2)}{\leq} \|\delta_t\|^2 - 2\gamma(f_t(x^t) - f_t(x^*)) + \gamma^2\mathbb{E}_t\left[\|\nabla f_{\mathbf{S}_t,t}(x^t)\|^2\right] \\
&\overset{(4.1)}{=} \|\delta_t\|^2 - 2\gamma(1-\gamma)(f_t(x^t) - f_t(x^*)) \\
&\overset{f_t(x^*)=0}{=} \|\delta_t\|^2 - 2\gamma(1-\gamma) f_t(x^t).
\end{aligned} \tag{4.6}$$

Taking total expectation for all $t \in \{0, \ldots, k-1\}$, we have that

$$\mathbb{E}\left[\|\delta_{t+1}\|^2\right] \leq \mathbb{E}\left[\|\delta_t\|^2\right] - 2\gamma(1-\gamma)\,\mathbb{E}\left[f_t(x^t)\right]. \tag{4.7}$$

Summing both sides of (4.7) from 0 to $k-1$ gives

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] + 2\gamma(1-\gamma)\sum_{t=0}^{k-1}\mathbb{E}\left[f_t(x^t)\right] \leq \|x^0 - x^*\|^2.$$

Dividing through by $2\gamma(1-\gamma) > 0$ and by k, we have that

$$\mathbb{E}\left[\min_{t=0,\dots,k-1} f_t(x^t)\right] \leq \min_{t=0,\dots,k-1} \mathbb{E}\left[f_t(x^t)\right] \leq \frac{1}{k}\sum_{t=0}^{k-1} \mathbb{E}\left[f_t(x^t)\right] \leq \frac{1}{k}\frac{\|x^0 - x^*\|^2}{2\gamma(1-\gamma)},$$

where in the leftmost inequality we used Jensen's inequality.

Finally, if (4.4) holds, then we can repeat the steps leading up to (4.6) without the conditional expectation, so that

$$\|\delta_{t+1}\|^2 \overset{(3.5)+(4.4)+(4.1)}{\leq} \|\delta_t\|^2 - 2\gamma(1-\gamma) f_{\mathbf{S}_t,t}(x^t).$$

Since $f_{S_t,t}(x^t) \geq 0$, we have $\|\delta_{t+1}\|^2 \leq \|\delta_t\|^2$, i.e., (4.5) holds. ∎

Theorem 4.3 is an unusual result for SGD methods. Currently, to get an $\mathcal{O}(1/k)$ convergence rate for SGD, one has to assume smoothness and strong convexity [21] or convexity, smoothness, and interpolation [53]. Here we get an $\mathcal{O}(1/k)$ rate by *only* assuming star-convexity. This is because we have smoothness and interpolation properties as a by-product due to our reformulation (3.1). However, the star-convexity assumption of $f_k(\cdot)$ for all $k \in \mathbb{N}$ is hard to interpret in terms of assumptions on F in general. But, we are able to interpret it in many important extremes. That is, for the full NR method, we show that it suffices for the Newton direction to be 2–cocoercive (see (5.8) in section 5). For the other extreme where the sketching matrix samples a single row, then the star-convexity assumption is even easier to check and is guaranteed to hold so long as $F_i(x)^2$ is convex for all $i = 1, \dots, m$ (see section 6).

Next, we will show the convergence of $F(x^k)$ instead of $f_k(x^k)$ via Theorem 4.3.

4.2.1. Sublinear convergence of the Euclidean norm $\|F\|$. If $\mathbb{E}[\mathbf{H}_\mathbf{S}(x)]$ is invertible for all $x \in \mathbb{R}^p$, we can use Theorem 4.3 with the bound (4.5) to guarantee that $\|F\|$ converges sublinearly. Indeed, when $\mathbb{E}[\mathbf{H}_\mathbf{S}(x)]$ is invertible, $\mathbb{E}[\mathbf{H}_\mathbf{S}(x)]$ is symmetric positive definite. Thus there exists $\lambda > 0$ that bounds the smallest eigenvalue away from zero in any closed bounded set (e.g., $\{x \in \mathbb{R}^p \mid \|x - x^*\| \leq \|x^0 - x^*\|\}$[5]):

$$\min_{x \in \{x \mid \|x - x^*\| \leq \|x^0 - x^*\|\}} \lambda_{\min}\left(\mathbb{E}[\mathbf{H}_\mathbf{S}(x)]\right) = \lambda > 0, \tag{4.8}$$

where $\lambda_{\min}(\cdot)$ is the smallest eigenvalue operator. Consequently, under the assumption of Theorem 4.3 with the condition (4.4), from (4.5) and (4.8), we have

$$\lambda \mathbb{E}\left[\min_{t=0,\dots,k-1} \|F(x^t)\|^2\right] \leq \mathbb{E}\left[\min_{t=0,\dots,k-1} \|F(x^t)\|^2_{\mathbb{E}[\mathbf{H}_\mathbf{S}(x^t)]}\right] \overset{(4.3)}{\leq} \frac{1}{k}\frac{\|x^0 - x^*\|^2}{\gamma(1-\gamma)}. \tag{4.9}$$

It turns out that using the smallest eigenvalue of $\mathbb{E}[\mathbf{H}_\mathbf{S}(x)]$ in the above bound is overly pessimistic. To improve it, first note that we do *not need* that $\mathbb{E}[\mathbf{H}_\mathbf{S}(x)]$ is invertible. Instead, we only need that $F(x) \in \mathbf{Im}(DF(x)^\top) \subset \mathbf{Im}(\mathbb{E}[\mathbf{H}_\mathbf{S}(x)])$, as we show in Corollary 4.5. But first, we need the following lemma.

LEMMA 4.4 (Lemma 10 in [19]). *For any matrix* $\mathbf{W}$ *and symmetric positive semi-definite matrix* $\mathbf{G}$ *s.t.* $\mathbf{Ker}(\mathbf{G}) \subset \mathbf{Ker}(\mathbf{W})$, *we have* $\mathbf{Ker}(\mathbf{W}^\top) = \mathbf{Ker}(\mathbf{W}\mathbf{G}\mathbf{W}^\top)$.

Note $L \overset{\text{def}}{=} \sup_{x \in \{x \mid \|x - x^*\| \leq \|x^0 - x^*\|\}} \|DF(x)\| > 0$. Such L exists because x is in a closed bounded convex set and because we have assumed that $DF(\cdot)$ is continuous. A continuous mapping over a closed bounded convex set is bounded. Now we can state the sublinear convergence results for $\|F\|$.

[5] We can rewrite the set as the closure of the ball $\{x \in \mathbb{R}^p \mid x \in \overline{\mathcal{B}(x^*, \|x^0 - x^*\|)}\}$.

COROLLARY 4.5. *Let*

$$\rho(x) \overset{def}{=} \min_{v \in \mathbf{Im}(DF(x))/\{0\}} \frac{v^\top DF(x)\mathbb{E}\left[\mathbf{H_S}(x)\right] DF(x)^\top v}{\|v\|^2}, \tag{4.10}$$

$$\rho \overset{def}{=} \min_{x \in \{x \mid \|x-x^*\| \leq \|x^0-x^*\|\}} \rho(x). \tag{4.11}$$

It follows that $0 \leq \rho(x) \leq 1$. *If*

$$F(x) \in \mathbf{Im}(DF(x)^\top) \subset \mathbf{Im}(\mathbb{E}\left[\mathbf{H_S}(x)\right]) \quad \textit{for all } x \in \mathbb{R}^p, \tag{4.12}$$

then $\rho(x) = \lambda_{\min}^+\left(DF(x)\mathbb{E}\left[\mathbf{H_S}(x)\right] DF(x)^\top\right) > 0$ *for all* $x \in \mathbb{R}^p$, *and* $\rho > 0$, *where* $\lambda_{\min}^+$ *is the smallest* nonzero *eigenvalue. Furthermore, if the star-convexity for each sketching matrix* (4.4) *holds, then*

$$\mathbb{E}\left[\min_{t=0,\ldots,k-1} \|F(x^t)\|^2\right] \leq \frac{1}{k} \cdot \frac{L^2\|x^0-x^*\|^2}{\rho\gamma\left(1-\gamma\right)}. \tag{4.13}$$

Proof. First recall that $\left(DF(x)\mathbf{H_S}(x)DF(x)^\top\right)^2 = DF(x)\mathbf{H_S}(x)DF(x)^\top$ for all $x \in \mathbb{R}^p$, which is shown in the proof of Lemma 4.1. Thus $DF(x)\mathbf{H_S}(x)DF(x)^\top$ is a projection. By Jensen's inequality, the eigenvalues of an expected projection are between 0 and 1. Thus by the definition of $\rho(x)$, we have $0 \leq \rho(x) \leq 1$. Next, by (4.12), we have $\mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H_S}(x)\right]\right) \subset \mathbf{Ker}\left(DF(x)\right)$. Thus, we have that

$$\mathbf{Im}\left(DF(x)\right) = \left(\mathbf{Ker}\left(DF(x)^\top\right)\right)^\perp = \left(\mathbf{Ker}\left(DF(x)\mathbb{E}\left[\mathbf{H_S}(x)\right] DF(x)^\top\right)\right)^\perp, \tag{4.14}$$

where the second equality is obtained by Lemma 4.4. Now from the definition of $\rho(x)$ in (4.10), we have

$$\begin{aligned}\rho(x) &\overset{(4.14)}{=} \min_{v \in (\mathbf{Ker}(DF(x)\mathbb{E}[\mathbf{H_S}(x)]DF(x)^\top))^\perp/\{0\}} \frac{v^\top DF(x)\mathbb{E}\left[\mathbf{H_S}(x)\right] DF(x)^\top v}{\|v\|^2} \\ &= \lambda_{\min}^+\left(DF(x)\mathbb{E}\left[\mathbf{H_S}(x)\right] DF(x)^\top\right) > 0.\end{aligned}$$

It now follows that $\rho > 0$, since the definition of ρ in (4.11) is given by minimizing $\rho(x)$ over the closed bounded set $\{x \mid \|x-x^*\| \leq \|x^0-x^*\|\}$. Next, given $x \in \{x \mid \|x-x^*\| \leq \|x^0-x^*\|\}$, since $F(x) \in \mathbf{Im}(DF(x)^\top)$ by (4.12) and noticing that $\mathbf{Im}(DF(x)^\top) = \mathbf{Im}(DF(x)^\top DF(x))$, there exists $v \in \mathbb{R}^m$ s.t. $F(x) = DF(x)^\top DF(x)v$.

If $F(x) \neq 0$, then $DF(x)v \in \mathbf{Im}\left(DF(x)\right)/\{0\}$, we have

$$\begin{aligned}\|F(x)\|^2_{\mathbb{E}[\mathbf{H_S}(x)]} &= v^\top DF(x)^\top DF(x)\mathbb{E}\left[\mathbf{H_S}(x)\right] DF(x)^\top DF(x)v \\ &\overset{(4.10)}{\geq} \rho(x)v^\top DF(x)^\top DF(x)v. \end{aligned} \tag{4.15}$$

Since $F(x) = DF(x)^\top DF(x)v$ and $\mathbf{Im}(DF(x)^\top) \oplus \mathbf{Ker}(DF(x)) = \mathbb{R}^m$,[6] we have that

$$\exists!\ y \in \mathbf{Ker}(DF(x)) \subset \mathbb{R}^m \text{ s.t. } v = (DF(x)^\top DF(x))^\dagger F(x) + y.$$

Thus $DF(x)v = DF(x)(DF(x)^\top DF(x))^\dagger F(x) = (DF(x)^\top)^\dagger F(x)$.

Substituting this in (4.15), we have that

$$\|F(x)\|^2_{\mathbb{E}[\mathbf{H_S}(x)]} \geq \rho(x)\|F(x)\|^2_{(DF(x)^\top DF(x))^\dagger} \geq \frac{\rho}{L^2}\|F(x)\|^2, \tag{4.16}$$

[6] The operator $\oplus$ denotes the direct sum of two vector spaces.

where in the last inequality we use that $\sup_{x\in\{x\mid\|x-x^*\|\le\|x^0-x^*\|\}}\|DF(x)\|\le L$ and $\rho(x)\ge\rho$ by the definition of ρ in (4.11) .

If $F(x)=0$, (4.16) still holds. Thus, for all $x\in\{x\mid\|x-x^*\|\le\|x^0-x^*\|\}$, (4.16) holds. Consequently by Theorem 4.3 and (4.5) under the star-convexity condition (4.4) with $\|x^t-x^*\|\le\|x^0-x^*\|$ for all $t\in\{0,\dots,k-1\}$, we have that

$$\frac{\rho}{L^2}\mathbb{E}\left[\min_{t=0,\dots,k-1}\|F(x^t)\|^2\right]\overset{(4.16)}{\le}\mathbb{E}\left[\min_{t=0,\dots,k-1}\|F(x^t)\|^2_{\mathbb{E}[\mathbf{H_S}(x^t)]}\right]\overset{(4.3)}{\le}\frac{1}{k}\frac{\|x^0-x^*\|^2}{\gamma(1-\gamma)},$$

which after multiplying through by $L^2/\rho>0$ concludes the proof. ∎

Thus with Corollary 4.5, we show that $F(x^t)$ converges to zero. This lemma relies on the inclusion (4.12), which in turn imposes some restrictions on the sketching matrix and $F(x)$. In our forthcoming examples in sections 5 and 6, we can directly verify the inclusion of (4.12). For other examples in sections 7 and 8, we provide the following Lemma 4.6, where we give sufficient conditions for (4.12) to hold.

LEMMA 4.6. *Let $F(x)\in\mathbf{Im}(DF(x)^\top)$. Furthermore, we suppose that $\mathbf{S}\sim\mathcal{D}_x$ is* adapted *to $DF(x)$ by which we mean*

$$\mathbf{Ker}\left(\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]\right)\subset\mathbf{Ker}(DF(x))\subset\mathbf{Ker}\left(\mathbf{S}^\top\right)\quad\text{for all }\mathbf{S}\sim\mathcal{D}_x. \tag{4.17}$$

Then it follows that (4.12) *holds for all $x\in\mathbb{R}^p$.*

Proof. Since $\mathbf{Ker}(DF(x)^\top DF(x))=\mathbf{Ker}(DF(x))\overset{(4.17)}{\subset}\mathbf{Ker}(\mathbf{S}^\top)$, we have

$$\mathbf{Ker}\left(\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)^\dagger\right)=\mathbf{Ker}\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)=\mathbf{Ker}(\mathbf{S}), \tag{4.18}$$

where the last equality is obtained by Lemma 4.4 with $\mathbf{Ker}(DF(x)^\top DF(x))\subset\mathbf{Ker}(\mathbf{S}^\top)$. Thus, using Lemma 4.4 again with $\mathbf{G}=\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)^\dagger$, $\mathbf{W}=\mathbf{S}$, and $\mathbf{Ker}(\mathbf{G})\subset\mathbf{Ker}(\mathbf{W})$ given by (4.18), we have that

$$\mathbf{Ker}(\mathbf{H_S}(x))\overset{(1.5)}{=}\mathbf{Ker}(\mathbf{S}\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)^\dagger\mathbf{S}^\top)=\mathbf{Ker}(\mathbf{S}^\top)=\mathbf{Ker}(\mathbf{S}\mathbf{S}^\top). \tag{4.19}$$

As $\mathbf{H_S}(x)$ is symmetric positive semidefinite for all $\mathbf{S}\sim\mathcal{D}_x$, we have that

$$\begin{aligned}
&v\in\mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H_S}(x)\right]\right)\Longleftrightarrow\mathbb{E}\left[\mathbf{H_S}(x)\right]v=0\Longleftrightarrow\|v\|^2_{\mathbb{E}[\mathbf{H_S}(x)]}=0\quad(\text{as }\mathbb{E}\left[\mathbf{H_S}(x)\right]\succeq 0)\\
&\Longleftrightarrow\mathbb{E}\left[\|v\|^2_{\mathbf{H_S}(x)}\right]=0\Longleftrightarrow\int_{\mathbf{S}}\|v\|^2_{\mathbf{H_S}(x)}d\mathbb{P}_{\mathcal{D}_x}(\mathbf{S})=0\\
&\Longleftrightarrow\|v\|^2_{\mathbf{H_S}(x)}=0\ \text{ for all }\ \mathbf{S}\sim\mathcal{D}_x\quad\left(\text{as }\|v\|^2_{\mathbf{H_S}(x)}\ge 0\ \text{ for all }\ \mathbf{S}\right)\\
&\overset{\mathbf{H_S}(x)\succeq 0}{\Longleftrightarrow}v\in\mathbf{Ker}\left(\mathbf{H_S}(x)\right)\ \text{ for all }\ \mathbf{S}\sim\mathcal{D}_x\Longleftrightarrow v\in\bigcap_{\mathbf{S}\sim\mathcal{D}_x}\mathbf{Ker}\left(\mathbf{H_S}(x)\right),
\end{aligned}$$

where we use $\bigcap_{\mathbf{S}\sim\mathcal{D}_x}\mathbf{Ker}(\mathbf{H_S}(x))$ to note the intersection of the random subsets $\mathbf{Ker}(\mathbf{H_S}(x))$ for all $\mathbf{S}\sim\mathcal{D}_x$. Similarly, we have $\mathbf{Ker}(\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right])=\bigcap_{\mathbf{S}\sim\mathcal{D}_x}\mathbf{Ker}(\mathbf{S}\mathbf{S}^\top)$ because $\mathbf{S}\mathbf{S}^\top$ is also symmetric, positive semidefinite for all $\mathbf{S}\sim\mathcal{D}_x$. Thus we have

$$\begin{aligned}
\mathbf{Ker}(\mathbb{E}\left[\mathbf{H_S}(x)\right])&=\bigcap_{\mathbf{S}\sim\mathcal{D}_x}\mathbf{Ker}(\mathbf{H_S}(x))\\
&\overset{(4.19)}{=}\bigcap_{\mathbf{S}\sim\mathcal{D}_x}\mathbf{Ker}(\mathbf{S}\mathbf{S}^\top)=\mathbf{Ker}(\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right])\overset{(4.12)}{\subset}\mathbf{Ker}(DF(x)).
\end{aligned}$$

Consequently, by considering the complement of the above, we arrive at (4.12). ∎

We refer to a sketching matrix $\mathbf{S} \sim \mathcal{D}_x$ that satisfies (4.17) as a sketch that is *adapted to* $DF(x)$. One easy way to design such adapted sketches is the following.

LEMMA 4.7. *Let* $\hat{\mathbf{S}} \in \mathbb{R}^{p\times\tau}$ *s.t.* $\hat{\mathbf{S}} \sim \mathcal{D}$ *a fixed distribution independent to* x *and* $\mathbf{Ker}(\mathbb{E}[\hat{\mathbf{S}}\hat{\mathbf{S}}^\top]) \subset \mathbf{Ker}(DF(x)^\top)$. *Thus,* $\mathbf{S} = DF(x)^\top\hat{\mathbf{S}} \in \mathbb{R}^{m\times\tau}$ *is adapted to* $DF(x)$.

Proof. First, $\mathbf{Ker}(DF(x)) \subset \mathbf{Ker}(\hat{\mathbf{S}}^\top DF(x)) = \mathbf{Ker}(\mathbf{S}^\top)$. Furthermore, from Lemma 4.4 with $\mathbf{Ker}(\mathbb{E}[\hat{\mathbf{S}}\hat{\mathbf{S}}^\top]) \subset \mathbf{Ker}(DF(x)^\top)$, we conclude the proof with

$$\mathbf{Ker}\left(\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]\right) = \mathbf{Ker}(DF(x)^\top\mathbb{E}[\hat{\mathbf{S}}\hat{\mathbf{S}}^\top]DF(x)) \subset \mathbf{Ker}(DF(x)). \qquad \square$$

The condition $\mathbf{Ker}(\mathbb{E}[\hat{\mathbf{S}}\hat{\mathbf{S}}^\top]) \subset \mathbf{Ker}(DF(x)^\top)$ in Lemma 4.7 holds for many standard sketches including Gaussian and subsampling sketches presented as follows.

LEMMA 4.8. *For Gaussian and uniform subsampling sketches defined in section* 1.4, *we have that* $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right] = c\mathbf{I}_m$ *with* $c > 0$ *a fixed constant depending on the sketch.*

Proof. For Gaussian sketches with $\mathbf{S}_{ij} \sim \mathcal{N}(0, \frac{1}{\tau})$, we have that $c = 1$. Indeed, since the mean is zero, off-diagonal elements of $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ are all zero. We note $\mathbf{S}_{i:}$ the ith row of $\mathbf{S}$, then the ith diagonal element of the matrix $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ is given by

$$\mathbb{E}\left[\mathbf{S}_{i:}\mathbf{S}_{i:}^\top\right] = \sum_{j=1}^{\tau}\mathbb{E}\left[\mathbf{S}_{ij}^2\right] = \sum_{j=1}^{\tau}\frac{1}{\tau} = 1.$$

For the uniform subsampling sketch (1.6), we have again that off-diagonal elements are zero since the rows of $\mathbf{S}$ are orthogonal. The diagonal elements are constant with

$$\mathbb{E}\left[\mathbf{S}_{i:}\mathbf{S}_{i:}^\top\right] = \frac{1}{\binom{m}{\tau}}\sum_{C\subset\{1,\dots,m\},|C|=\tau,i\in C} 1 = \frac{\binom{m-1}{\tau-1}}{\binom{m}{\tau}} = \frac{\tau}{m} \quad \text{for all } i = 1,\dots,m. \qquad \square$$

From Lemma 4.8, we know that $\mathbb{E}[\hat{\mathbf{S}}\hat{\mathbf{S}}^\top] = c\mathbf{I}_p$ invertible with $c > 0$. Thus $\mathbf{Ker}(\mathbb{E}[\hat{\mathbf{S}}\hat{\mathbf{S}}^\top]) = \{0\} \subset \mathbf{Ker}(DF(x)^\top)$ holds for *any* sketch size τ.

4.3. Convergence for strongly convex. Here we establish a global linear convergence of `SNR` when assuming that f_y is strongly quasi-convex.

Assumption 4.9 (μ-strongly quasi-convexity). Let x^* satisfy Assumption 1.1 and

(4.20)
$$\exists\ \mu > 0 \text{ s.t. } f_y(x^*) \geq f_y(x) + \langle \nabla f_y(x), x^* - x\rangle + \frac{\mu}{2}\|x^* - x\|^2 \quad \text{for all} \quad x, y \in \mathbb{R}^p.$$

Assumption 4.9 is strong, so much so that we have the following lemma.

LEMMA 4.10. *Assumption* 4.9 *implies* (3.3) *and that the solution to* (1.1) *is unique.*

Proof. Let $y \in \mathbb{R}^p$ and let $u \in F(\mathbb{R}^p) \cap \mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(y)\right]\right)$. $u \in F(\mathbb{R}^p)$ implies that $\exists x \in \mathbb{R}^p$ s.t. $F(x) = u$. Besides, $u \in \mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(y)\right]\right)$ implies that $\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(y)\right]F(x) = 0$. Now we apply (4.20) at point x knowing that $f_y(x^*) = 0$:

$$\begin{aligned} &0 \geq f_y(x) + \langle \nabla f_y(x), x^* - x\rangle + \frac{\mu}{2}\|x^* - x\|^2 \\ \implies &0 \geq 0 + \langle 0, x^* - x\rangle + \frac{\mu}{2}\|x^* - x\|^2 \quad (\text{as } \mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(y)\right]F(x) = 0) \iff x = x^*. \end{aligned}$$

Thus $F(x) = u = 0$. We conclude $F(\mathbb{R}^p) \cap \mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(y)\right]\right) = \{0\}$, i.e., (3.3) holds.

Besides, let x' be a global minimizer of $f_y(\cdot)$. Then $f_y(x') = f_y(x^*) = 0$ and $\nabla f_y(x') = 0$. Similarly, by applying (4.20) at point x', we obtain $x' = x^*$. Consequently, x^* is the unique minimizer of $f_y(\cdot)$ for all y, thus the unique solution to (1.1), according to (3.3) and Lemma 3.1. $\square$

Under Assumption 4.9 , choosing $\gamma = 1$ guarantees a fast global linear convergence.

THEOREM 4.11. *If x^* satisfies Assumption 4.9 and $\gamma \leq 1$, then SNR converges linearly:*

$$\mathbb{E}\left[\|x^{k+1} - x^*\|^2\right] \leq (1-\gamma\mu)^{k+1}\|x^0 - x^*\|^2 \qquad \text{with} \quad \mu \leq 1. \tag{4.21}$$

Proof. Let $\delta_k \overset{\text{def}}{=} x^k - x^*$. By expanding the squares, similarly we have that

$$\begin{aligned}
\mathbb{E}_k\left[\|\delta_{k+1}\|^2\right] &= \|\delta_k\|^2 - 2\gamma\left\langle \delta_k, \nabla f_k(x^k)\right\rangle + \gamma^2\mathbb{E}_k\left[\|\nabla f_{\mathbf{S}_k,k}(x^k)\|^2\right] \\
&\overset{(4.20)}{\leq} (1-\gamma\mu)\|\delta_k\|^2 - 2\gamma(f_k(x^k) - f_k(x^*)) + \gamma^2\mathbb{E}_k\left[\|\nabla f_{\mathbf{S}_k,k}(x^k)\|^2\right] \\
&\overset{(4.1)}{\leq} (1-\gamma\mu)\|\delta_k\|^2 - 2\gamma\,(1-\gamma)\,(f_k(x^k) - f_k(x^*)) \\
&\leq (1-\gamma\mu)\|\delta_k\|^2 \qquad \left(\text{since } \gamma\,(1-\gamma)\,(f_k(x^k) - f_k(x^*)) \geq 0\right).
\end{aligned}$$

Now by taking total expectation, we have that

$$\mathbb{E}\left[\|x^{k+1} - x^*\|^2\right] \;\leq\; (1-\gamma\mu)\mathbb{E}\left[\|x^k - x^*\|^2\right] \;\leq\; (1-\gamma\mu)^{k+1}\|x^0 - x^*\|^2.$$

Next, we show that $\mu \leq 1$. In fact, when we imply (4.20) at the point x^k, it shows

$$\begin{aligned}
(4.20) \quad &\overset{(4.1)}{\Longrightarrow} \quad f_k(x^*) \geq \frac{1}{2}\mathbb{E}_k\left[\|\nabla f_{\mathbf{S}_k,k}(x^k)\|^2\right] + \left\langle x^* - x^k, \nabla f_k(x^k)\right\rangle + \frac{\mu}{2}\|x^* - x^k\|^2 \\
&\iff \quad f_k(x^*) \geq \frac{1}{2}\mathbb{E}_k\left[\|x^* - \left(x^k - \nabla f_{\mathbf{S}_k,k}(x^k)\right)\|^2\right] - \frac{1-\mu}{2}\|x^* - x^k\|^2 \\
&\overset{f_k(x^*)=0}{\Longrightarrow} (1-\mu)\|x^* - x^k\|^2 \geq \mathbb{E}_k\left[\|x^* - \left(x^k - \nabla f_{\mathbf{S}_k,k}(x^k)\right)\|^2\right] \geq 0.
\end{aligned}$$

Thus $\mu \leq 1$. ∎

5. New global convergence theory of the NR method. As a direct consequence of our general convergence theorems, in this section we develop a new global convergence theory for the original NR method. We first provide the results in one dimension in section 5.1, then a general result in higher dimensions in the subsequent section 5.2, and we compare this result to the classic MCT in section 5.3.

5.1. A single nonlinear equation. Consider the case where $F(x) = \phi(x) \in \mathbb{R}$ is a one-dimensional function and $x \in \mathbb{R}$. This includes common applications of the NR method such as calculating square roots of their reciprocal[7] and finding roots of polynomials. Even in this simple one-dimensional case, we find that our assumptions of global convergence given in Corollary 4.5 are *strictly weaker* than the standard assumptions used to guarantee NR convergence, as we explain next.

The NR method in one dimension at every iteration k is given by

$$x^{k+1} \;=\; x^k - \frac{\phi(x^k)}{\phi'(x^k)} \;\overset{\text{def}}{=}\; g(x^k).$$

To guarantee that this is well defined, we assume that $\phi'(x^k) \neq 0$ for all k. A sufficient condition for this procedure to converge locally is that $|g'(x)| < 1$ with $x \in I$

[7] Used in particular to compute angles of incidence and reflection in games such as Quake (https://en.wikipedia.org/wiki/Fast_inverse_square_root).

where I is a given interval containing the solution x^*. See, for example, section 1.1 in [15] or Chapter 12 in [44]. We can extend this to a global convergence by requiring that $|g'(x)| < 1$ globally. In the case of NR, since $g'(x) = 1 - \frac{\phi'(x)^2 - \phi(x)\phi''(x)}{\phi'(x)^2} = \frac{\phi(x)\phi''(x)}{\phi'(x)^2}$, this condition amounts to requiring

$$\frac{|\phi(x)\phi''(x)|}{\phi'(x)^2} < 1. \tag{5.1}$$

Curiously, condition (5.1) has an interesting connection to convexity. In fact, condition (5.1) implies that $\phi^2(x)$ is convex and twice continuously differentiable. To see this, note that $\frac{d^2}{dx^2}\phi^2(x) \geq 0$ is equivalent to

$$\frac{d^2}{dx^2}\phi^2(x) \;=\; 2\frac{d}{dx}\phi'(x)\phi(x) \;=\; 2\left(\phi(x)\phi''(x) + \phi'(x)^2\right) \;\geq\; 0. \tag{5.2}$$

Now it is easy to see that (5.1) implies (5.2). Finally (5.2) also implies that $\phi^2(x)$ is *star-convex*, which is exactly what is required by our convergence theory in Corollary 4.5.

Indeed, in this one-dimensional setting, Assumption 4.2 is equivalent to (4.4) and our reformulation in (3.1) boils down to minimizing $f_y(x) = \left(\phi(x)/\phi'(y)\right)^2$. Thus by Corollary 4.5, the NR method converges globally if $f_{x^k}(x)$, or simply if $\phi(x)^2$ is star-convex and $\phi'(x^k) \neq 0$ for all iterates of NR, which shows that our condition is strictly weaker than the other conditions, because there exist functions that are star-convex but not convex, e.g. $\phi(x)^2 = |x|(1 - \exp(-|x|))$ from [42, 33].

For future reference and convenience, we can rewrite the star-convexity of each $\phi(x)^2$ as

$$0 = \phi(x^*)^2 \;\geq\; \phi^2(x) + 2\phi(x)\phi'(x)(x^* - x),$$

where x^* is the global minimum of $\phi(x)^2$, i.e., $\phi(x^*) = 0$. This can be rewritten as

$$0 \geq \phi(x)\left(\phi(x) + 2\phi'(x)(x^* - x)\right). \tag{5.3}$$

By verifying (5.3) and that $\phi'(x^k) \neq 0$ on the iterates of NR, we can guarantee that the method converges globally.

5.2. The full NR. Now let $F(x) \in \mathbb{R}^m$ and consider the full NR method (1.2). Similarly, since $\mathbf{S} = \mathbf{I}_m$, Assumption 4.2 is equivalent to (4.4). Corollary 4.5 sheds some new light on the convergence of NR. In this case, our reformulation (3.1) is given by

$$f_y(x) \;=\; \frac{1}{2}F(x)^\top (DF(y)^\top DF(y))^\dagger F(x) \;=\; \frac{1}{2}\| \left(DF(y)^\top\right)^\dagger F(x)\|^2 \tag{5.4}$$

and Corollary 4.5 states that NR converges if $f_{x^k}(x)$ is star-convex for all the iterates $x^k \in \mathbb{R}^p$. This has a curious reinterpretation in this setting. Indeed, let

$$n(x) \stackrel{\text{def}}{=} -(DF(x)^\top)^\dagger F(x) \tag{5.5}$$

be the Newton direction. From (5.4) and (5.5), we have that

$$f_x(x) = \frac{1}{2}\|n(x)\|^2. \tag{5.6}$$

Using (5.6), Corollary 4.5 can be stated in this special case as the following corollary.

COROLLARY 5.1. *Consider x^k given by the* NR *(1.2) with $\gamma < 1$. If we have*

$$F(x) \in \mathbf{Im}(DF(x)^\top), \tag{5.7}$$

$$\frac{1}{2}\|n(x)\|^2 \leq \langle n(x), x^* - x\rangle \tag{5.8}$$

hold for every $x = x^k$ with solution x^*, then it exists $L > 0$ s.t. $\|DF(x^k)\| \leq L$ and

$$\min_{t=0,\ldots,k-1} \|F(x^t)\|^2 \leq \frac{1}{k} \cdot \frac{L^2\|x^0 - x^*\|^2}{\gamma\,(1-\gamma)}. \tag{5.9}$$

Proof. From (3.4), we have that

$$\nabla f_x(x) \;=\; DF(x)(DF(x)^\top DF(x))^\dagger F(x) \;=\; (DF(x)^\top)^\dagger F(x) \;=\; -n(x). \tag{5.10}$$

Substituting (5.6) and (5.10) in (4.4) yields (5.8). Next, for $\mathbf{S} = \mathbf{I}_m$, we have that

$$\mathbf{Im}(\mathbb{E}\,[\mathbf{H_S}(x)])) = \mathbf{Im}((DF(x)^\top DF(x))^\dagger) = \mathbf{Im}(DF(x)^\top DF(x)) = \mathbf{Im}(DF(x)^\top).$$

Thus, we have that $F(x) \in \mathbf{Im}(DF(x)^\top) \subset \mathbf{Im}(\mathbb{E}\,[\mathbf{H_S}(x)]))$, i.e., (4.12) holds. So all the conditions in Corollary 4.5 are verified. Since $\mathbf{S} = \mathbf{I}_m$, we have that $\rho(x) = 1$ for all x, so $\rho = 1$. Furthermore, because we assume that $DF(\cdot)$ is continuous and the iterates x^k are in a closed bounded convex set (4.5) which is implied by (4.4) from Theorem 4.3, there exists $L > 0$ s.t. $\|DF(x^k)\| \leq L$ for all the iterates. Finally, by Corollary 4.5, the iterates converge sublinearly according to (4.13), which in this case is given by (5.9). ∎

Condition (5.8) can be seen as a co-coercivity property of the Newton direction. This co-coercivity establishes a curious link with the modern proofs of convergence of gradient descent which rely on the co-coercivity of the gradient direction. That is, if $f(x)$ is convex and L-smooth, then we have that the gradient is L–co-coercive with

$$\frac{1}{L}\|\nabla f(x)\|^2 \;\leq\; \langle \nabla f(x), x - x^*\rangle\,.$$

This is the key property for proving convergence of gradient descent; see, e.g., section 5.2.4 in [4]. To the best of our knowledge, this is the first time that the co-coercivity of the Newton direction has been identified as a key property for proving convergence of the Newton's method. In particular, global convergence results for the NR method such as the MCT only hold for functions $F : \mathbb{R}^p \to \mathbb{R}^m$ with $p = m$ and rely on a stepsize $\gamma = 1$; see [44, 15]. Corollary 5.1 accommodates "nonsquare" functions $F : \mathbb{R}^p \to \mathbb{R}^m$. Excluding the difference in stepsizes and focusing on "square" functions $F : \mathbb{R}^p \to \mathbb{R}^m$ with $p = m$, next we show in Theorem 5.2 that our assumptions are strictly weaker than those used for establishing the global convergence of NR with constant stepsizes through the MCT.

5.3. Comparing to the classic monotone convergence theory of NR. Consider $m = p$. Here we show that Assumption 1.1, (5.7), and (5.8) are strictly weaker than the classic assumptions used for establishing the global convergence of NR with constant stepsize. To show this, we take the assumptions used in the MCT in section 13.3.4 in [44] and compare with our assumptions in the following theorem.

THEOREM 5.2. *Let $F : \mathbb{R}^p \to \mathbb{R}^p$ and let x^k be the iterates of the* NR *method with stepsize $\gamma = 1$, that is,*

$$x^{k+1} = x^k - \left(DF(x^k)^\top\right)^\dagger F(x^k). \tag{5.11}$$

Consider the following two sets of assumptions:

(I) *$F(x)$ is componentwise convex, $(DF(x)^\top)^{-1}$ exists and is elementwise positive for all $x \in \mathbb{R}^p$. There exist x and y s.t. $F(x) \le 0 \le F(y)$ elementwise.*

(II) *There exists a unique $x^* \in \mathbb{R}^p$ s.t. $F(x^*) = 0$, (5.7) and (5.8) hold for $k \ge 1$.*

If (I) *holds, then* (II) *always holds. Furthermore, there exist problems for which* (II) *holds and* (I) *does not hold.*

Proof. First, we prove (I) $\implies$ (II). Assume that (I) holds. Since $DF(x)$ is invertible, (5.7) holds trivially. By section 13.3.4 in [44], we know that there exists a unique $x^* \in \mathbb{R}^p$ s.t. $F(x^*) = 0$. It remains to verify if (5.8) holds for $k \ge 1$. First, note that the invertibility of $DF(x^k)$ gives

$$f_k(x^k) \;=\; \frac{1}{2}\|F(x^k)\|^2_{\left(DF_k^\top DF_k\right)^\dagger} \;=\; \frac{1}{2}\|(DF_k^\top)^{-1}F_k\|^2 \overset{(5.11)}{=} \frac{1}{2}\|x^{k+1}-x^k\|^2, \tag{5.12}$$

with abbreviations $f_k(x^k) \equiv f_{x^k}(x^k)$, $F_k \equiv F(x^k)$, and $DF_k \equiv DF(x^k)$. Furthermore,

$$\nabla f_k(x^k) \;=\; DF_k(DF_k^\top DF_k)^{-1}F(x^k) \;=\; (DF_k^\top)^{-1}F(x^k) \overset{(5.11)}{=} x^k - x^{k+1}. \tag{5.13}$$

Thus we can rewrite the right-hand side of the star-convexity assumption (4.2) as

$$\begin{aligned} f_k(x^k) + \left\langle \nabla f_k(x^k), x^* - x^k \right\rangle &\overset{(5.12)+(5.13)}{=} \frac{1}{2}\|x^{k+1}-x^k\|^2 + \left\langle x^k - x^{k+1}, x^* - x^k \right\rangle \\ &= \frac{1}{2}\|x^{k+1}-x^k\|^2 + \left\langle x^k - x^{k+1}, x^{k+1} - x^k + x^* - x^{k+1} \right\rangle \\ &= -\frac{1}{2}\|x^{k+1}-x^k\|^2 + \left\langle x^k - x^{k+1}, x^* - x^{k+1} \right\rangle. \end{aligned}$$

From (I), we induce by Lemma 3.1 in [15] that NR is componentwise monotone with $x^* \le x^{k+1} \le x^k$ for $k \ge 1$. Thus $x^k - x^{k+1} \ge 0$ and $x^* - x^{k+1} \le 0$ componentwise and consequently, $\left\langle x^k - x^{k+1}, x^* - x^{k+1} \right\rangle \le 0$. Thus it follows that

$$f_k(x^k) + \left\langle \nabla f_k(x^k), x^* - x^k \right\rangle \;\le\; 0 \;=\; f_k(x^*).$$

Thus (5.8) holds for $k \ge 1$ and this concludes that (I) $\implies$ (II).

We now prove that (II) does *not* imply (I). Consider the example $F(x) = Ax - b$, where $A \in \mathbb{R}^{p\times p}$ is invertible and $b \in \mathbb{R}^p$. Thus, $DF(x) = A^\top$ is invertible and (5.7) holds. As for (5.8), let x^* be the solution, i.e., $Ax^* = b$; we have that

$$f_k(x) = \frac{1}{2}\|F(x)\|^2_{(DF(x_k)^\top DF(x_k))^{-1}} = \frac{1}{2}\|A(x-x^*)\|^2_{(AA^\top)^{-1}} = \frac{1}{2}\|x-x^*\|^2,$$

which is a convex function and so (5.8) holds and thus (II) holds. However, (I) does not necessarily hold. Indeed, if $A = -\mathbf{I}_p$, then $DF(x)$ is not elementwise positive. ∎

We observe that our assumptions are also strictly weaker than the affine covariates formulations of convex functions given in Lemma 3.1 in [15]. The proof is verbatim to the above.

Theorem 5.2 only considers the case that the stepsize $\gamma = 1$. We also investigate the case where the stepsize $\gamma < 1$ in particular in one dimension and show that MCT does not hold in this case. Since this analysis is not the main interest of the paper, please refer to Appendix C in [59] for more details. Thus we claim that our assumptions are strictly weaker than the assumptions used in MCT [44, 15] for establishing the global convergence of NR, albeit for different stepsizes.

6. Single row sampling: The nonlinear Kaczmarz method. The SNR enjoys many interesting instantiations. From these, we have chosen three to present in the main text: the nonlinear Kaczmarz method in this section, SNM [32] in section 7, and a new specialized variant for solving GLMs in section 8.

Here we present the new nonlinear Kaczmarz method as a variant of SNR. Consider the original problem (1.1). We use a single row importance weighted subsampling sketch to sample rows of $F(x) = 0$. That is, let $\mathbb{P}[\mathbf{S} = e_i] = p_i$ with the ith unit coordinate vector $e_i \in \mathbb{R}^m$ for $i = 1, \ldots, m$. Then the SNR update (1.3) is given by

$$x^{k+1} = x^k - \gamma \frac{F_i(x^k)}{\|\nabla F_i(x^k)\|^2} \nabla F_i(x^k). \tag{6.1}$$

We dub (6.1) the *nonlinear Kaczmarz method*, as it can be seen as an extension of the randomized Kaczmarz method [27, 52] for solving linear systems to the nonlinear case.[8] By (3.2), this nonlinear Kaczmarz method is simply SGD applied to minimizing

$$f_{x^k}(x) \;=\; \sum_{i=1}^{m} \mathbb{P}[\mathbf{S} = e_i] f_{e_i, x^k}(x) \;\overset{(3.2)+(1.5)}{=}\; \frac{1}{2} \sum_{i=1}^{m} p_i \frac{F_i(x)^2}{\|\nabla F_i(x^k)\|^2}.$$

A sufficient condition for (3.3) to hold is that the diagonal matrix

$$\mathbb{E}_{e_i}\left[\mathbf{H}_{e_i}(x^k)\right] \overset{(1.5)}{=} \sum_{i=1}^{m} p_i \frac{e_i e_i^\top}{\|\nabla F_i(x^k)\|^2} = \mathbf{Diag}\left(\frac{p_i}{\|\nabla F_i(x^k)\|^2}\right) \tag{6.2}$$

is invertible. Thus $\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(x^k)\right]$ is invertible if $\nabla F_i(x^k) \neq 0$ for all $i \in \{1, \ldots, m\}$ and $x^k \in \mathbb{R}^p$, in which case $\mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(y)\right]\right) = \{0\}$ for all $y \in \mathbb{R}^p$ and (3.3) holds.

Finally, to guarantee that (6.1) converges through Theorem 4.3, we need $f_{x^k}(x)$ to be star-convex on x^k at every iteration. In this case, it suffices for each $F_i(x)^2$ to be star-convex, since any conic combination of star-convex functions is star-convex [33]. This is a straightforward abstraction of the one-dimensional case, in that, if (5.3) holds for every F_i in the place of ϕ, we can guarantee the convergence of (6.1). This is also equivalent to assuming the star-convexity for each sketching matrix (4.4). Furthermore, if $F(x) \in \mathbf{Im}(DF(x)^\top)$ holds for all x, then (4.12) holds, as $\mathbf{Ker}\left(\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(y)\right]\right) = \{0\}$. We can guarantee the convergence of (6.1) through Corollary 4.5.

7. The stochastic Newton method. We now show that SNM [32] is a special case of SNR. This connection combined with the global convergence theory of SNR gives us the first global convergence theory of SNM, which we detail in section 7.2.

SNM [32] is a stochastic second-order method that takes a Newton-type step at each iteration to solve optimization problems with a finite-sum structure

$$\min_{w \in \mathbb{R}^d} \left[P(w) \overset{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} \phi_i(w) \right], \tag{7.1}$$

where each $\phi_i : \mathbb{R}^d \to \mathbb{R}$ is twice differentiable and strictly convex. Briefly, the updates in SNM at the kth iteration are given by

$$w^{k+1} \;=\; \left(\frac{1}{n} \sum_{i=1}^{n} \nabla^2 \phi_i(\alpha_i^k) \right)^{-1} \left(\frac{1}{n} \sum_{i=1}^{n} \nabla^2 \phi_i(\alpha_i^k) \alpha_i^k - \frac{1}{n} \sum_{i=1}^{n} \nabla \phi_i(\alpha_i^k) \right), \tag{7.2}$$

[8] We note that there exists a nonlinear variant of the Kaczmarz method which is referred to as the Landweber–Kaczmarz method [34]. Though the Landweber–Kaczmarz is very similar to Kaczmarz, it is not truly an extension since it does not adaptively reweight the stepsize by $\|\nabla F_i(x^k)\|^2$.

$$\alpha_i^{k+1} \;=\; \begin{cases} w^{k+1} & \text{if } i \in B_n, \\ \alpha_i^k & \text{if } i \notin B_n, \end{cases} \tag{7.3}$$

where $\alpha_1^k, \ldots, \alpha_n^k$ are auxiliary variables, initialized in SNM, and $B_n \subset \{1, \ldots, n\}$ is a subset of size τ chosen uniformly on average from all subsets of size τ.

7.1. Rewrite SNM as a special case of SNR. Since $P(w)$ is strictly convex, every minimizer of P satisfies $\nabla P(w) = \frac{1}{n}\sum_{i=1}^n \nabla\phi_i(w) = 0$. Our main insight to deducing SNM is that we can rewrite this stationarity condition using a *variable splitting trick*. That is, by introducing a new variable $\alpha_i \in \mathbb{R}^d$ for each gradient $\nabla\phi_i$, and letting $p := (n+1)d$ and $x = \begin{bmatrix} w \,;\alpha_1 \,;\cdots\,;\alpha_n \end{bmatrix} \in \mathbb{R}^p$ be the stacking[9] of the w and α_i variables, we have that solving $\nabla P(w) = 0$ is equivalent to finding the *roots* of the following nonlinear equations:

$$F(x) \;=\; F(w \,;\alpha_1 \,;\cdots\,;\alpha_n) \;\overset{\text{def}}{=}\; \left[\frac{1}{n}\sum_{i=1}^n \nabla\phi_i(\alpha_i) \;;\; w-\alpha_1 \;;\; \cdots \;;\; w-\alpha_n\right], \tag{7.4}$$

where $F : \mathbb{R}^{(n+1)d} \to \mathbb{R}^{(n+1)d}$. Our objective now becomes solving $F(x) = 0$ with $p = m = (n+1)d$. To apply SNR to (7.4), we are going to use a structured sketching matrix. But first, we need some additional notation.

Divide $\mathbf{I}_{nd} \in \mathbb{R}^{nd\times nd}$ into n contiguous blocks of size $nd \times d$ as follows:

$$\mathbf{I}_{nd} \overset{\text{def}}{=} [\; \mathbf{I}_{nd,1}\; \mathbf{I}_{nd,2} \cdots \mathbf{I}_{nd,n} \;]$$

where $\mathbf{I}_{nd,i}$ is the ith block of $\mathbf{I}_{nd}$. Let $B_n \subset \{1, \ldots, n\}$ with $|B_n| = \tau$ chosen uniformly at average. Let $\mathbf{I}_{B_n} \in \mathbb{R}^{nd\times\tau d}$ denote the concatenation of the blocks $\mathbf{I}_{nd,i}$ such that the indices $i \in B_n$.

At the kth iteration of SNR, denoting $x^k = [w^k;\ \alpha_1^k;\ \cdots;\ \alpha_n^k]$, we define our sketching matrix $\mathbf{S} \sim \mathcal{D}_{x^k}$ as

$$\mathbf{S} = \begin{bmatrix} \mathbf{I}_d & 0 \\ \frac{1}{n}\nabla^2\phi_1(\alpha_1^k) & \\ \vdots & \mathbf{I}_{B_n} \\ \frac{1}{n}\nabla^2\phi_n(\alpha_n^k) & \end{bmatrix} \in \mathbb{R}^{(n+1)d\times(\tau+1)d}. \tag{7.5}$$

Here the distribution $\mathcal{D}_{x^k}$ depends on the iterates x^k. The sketch size of $\mathbf{S}$ is $(\tau+1)d$ with *any* $\tau \in \{1, \ldots, n\}$. Now we can state the following lemma.

LEMMA 7.1. *Let ϕ_i be strictly convex for $i = 1, \ldots, n$. At each iteration k, the updates of SNR* (1.3) *with F defined in* (7.4)*, the sketching matrix $\mathbf{S}_k$ defined in* (7.5)*, and stepsize $\gamma = 1$ are equal to the updates* (7.2) *and* (7.3) *of SNM.*

In our upcoming proof of Lemma 7.1, we still need the following lemma.

LEMMA 7.2. *Let ϕ_i be twice differentiable and strictly convex for $i = 1, \ldots, n$. The Jacobian $DF(x)^\top$ of $F(x)$ defined in* (7.4) *is invertible for all $x \in \mathbb{R}^{(n+1)d}$.*

Proof. Let $x \in \mathbb{R}^{(n+1)d}$. Let $y \overset{\text{def}}{=} (u; v_1; \cdots; v_n) \in \mathbb{R}^{(n+1)d}$ with $u, v_1, \ldots, v_n \in \mathbb{R}^d$ such that $DF(x)y = 0$. The transpose of the Jacobian of $F(x)$ is given by

$$DF(x) = \begin{bmatrix} 0 & \mathbf{I}_d & \cdots & \mathbf{I}_d \\ \frac{1}{n}\nabla^2\phi_1(\alpha_1) & & & \\ \vdots & & -\mathbf{I}_{nd} & \\ \frac{1}{n}\nabla^2\phi_n(\alpha_n) & & & \end{bmatrix}. \tag{7.6}$$

[9] In this paper, vectors are columns by default, and given $x_1, \ldots, x_n \in \mathbb{R}^q$, we note $[x_1; \ldots; x_n] \in \mathbb{R}^{qn}$ the (column) vector stacking the x_i's on top of each other with $q \in \mathbb{N}$.

From $DF(x)y = 0$ and (7.6), we obtain

$$\sum_{i=1}^{n} v_i = 0 \quad \text{and} \quad \frac{1}{n}\nabla^2\phi_i(\alpha_i)u = v_i \quad \text{for all } i = 1, \dots, n.$$

Plugging the second equation into the first one gives $(\frac{1}{n}\sum_{i=1}^{n}\nabla^2\phi_i(\alpha_i))u = 0$. Since every ϕ_i is twice differentiable and strictly convex, we have $\nabla^2\phi_i(\alpha_i) > 0$. This implies $\frac{1}{n}\sum_{i=1}^{n}\nabla^2\phi_i(\alpha_i) > 0$ and is thus invertible. Consequently $u = 0$ and $v_i = 0$, from which we conclude that the Jacobian $DF(x)^\top$ is invertible. ∎

Now we can give the proof of Lemma 7.1.

Proof. Consider an update of SNR (1.3) with F defined in (7.4), the sketching matrix $\mathbf{S}_k$ defined in (7.5) , and stepsize $\gamma = 1$ at the kth iteration. By Lemma 7.2, we have that $DF(x)$ is invertible and thus Assumption 2.1 holds. By (2.2), the SNR update (1.3) can be rewritten as

(7.7)
$$x^{k+1} = \operatorname{argmin} \|w - w^k\|^2 + \sum_{i=1}^{n} \|\alpha_i - \alpha_i^k\|^2 \text{ s.t. } \mathbf{S}_k^\top DF(x^k)^\top (x - x^k) = -\mathbf{S}_k^\top F(x^k).$$

Plugging (7.4), (7.5), and (7.6) into the constraint in (7.7) and simplifying the matrix multiplications, we have that (7.7) is given by

$$\begin{aligned} x^{k+1} = [w^{k+1}; \alpha_1^{k+1}; \cdots; \alpha_n^{k+1}] = \operatorname{argmin} \|w - w^k\|^2 + \sum_{i=1}^{n} \|\alpha_i - \alpha_i^k\|^2 \\ \text{s. t.} \quad \frac{1}{n}\sum_{i=1}^{n}\nabla^2\phi_i(\alpha_i^k)(w - \alpha_i^k) = -\frac{1}{n}\sum_{i=1}^{n}\nabla\phi_i(\alpha_i^k), \\ w = \alpha_j \quad \text{for } j \in B_n. \end{aligned} \tag{7.8}$$

To solve (7.8), first note that $\alpha_i^{k+1} = \alpha_i^k$ for $i \notin B_n$, since there is no constraint on the variable α_i in this case. Furthermore, by the invertibility of $\frac{1}{n}\sum_{i=1}^{n}\nabla^2\phi_i(\alpha_i^k)$, we have that (7.8) has a unique solution s.t. $\alpha_j = w$ for all $j \in B_n$ and

$$w = \left(\frac{1}{n}\sum_{i=1}^{n}\nabla^2\phi_i(\alpha_i^k)\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\nabla^2\phi_i(\alpha_i^k)\alpha_i^k - \frac{1}{n}\sum_{i=1}^{n}\nabla\phi_i(\alpha_i^k)\right).$$

Thus the SNR update (7.8) is exactly the SNM updates (7.2) and (7.3) in [32]. ∎

7.2. Global convergence theory of SNM. Let $x' \overset{\text{def}}{=} (w'; \alpha_1'; \cdots; \alpha_n') \in \mathbb{R}^{(n+1)d}$ and $\mathbf{S} \sim \mathcal{D}_x$ defined in (7.5). By applying the global convergence theory of SNR, we can now provide the first global convergence theory for SNM.

COROLLARY 7.3. *Let w^* be a solution to $\nabla P(w) = 0$. Consider the iterate $x^k = (w^k; \alpha_1^k; \cdots; \alpha_n^k)$ given by SNM (7.2) and (7.3) and note $x^* \overset{def}{=} (w^*; w^*; \cdots; w^*) \in \mathbb{R}^{(n+1)d}$. If there exists $\mu > 0$ such that for all x, $x' \in \mathbb{R}^{(n+1)d}$,*

$$\begin{aligned} f_{x'}(x^*) \;&\geq\; f_{x'}(x) + \langle \nabla f_{x'}(x), x^* - x\rangle + \frac{\mu}{2}\|x^* - x\|^2 \\ &=\; f_{x'}(x) + \langle \nabla f_{x'}(x), x^* - x\rangle + \frac{\mu}{2}\left(\|w^* - w\|^2 + \sum_{i=1}^{n}\|w^* - \alpha_i\|^2\right), \end{aligned} \tag{7.9}$$

then the iterates $\{x^k\}$ of SNM converge linearly according to

$$\mathbb{E}\left[\|x^{k+1} - x^*\|^2\right] \leq (1 - \mu)^{k+1}\|x^0 - x^*\|^2. \tag{7.10}$$

Proof. As $\nabla P(w^*) = 0$, this implies immediately that x^* is a solution of F. Besides, (7.9) satisfies Assumption 4.9. Thus by Theorem 4.11, we get (7.10). ∎

Even though (7.9) is a strong assumption, this is the first global convergence theory of SNM, since only local convergence results of SNM are addressed in [32].

As a by-product, we find that the function $F(x)$ in (7.4) and the sketch $\mathbf{S}$ defined in (7.5) actually satisfy (4.12) through Lemma 4.6, namely as the following lemma.

LEMMA 7.4. *Consider the function F defined in* (7.4) *and the sketching matrix $\mathbf{S}$ defined in* (7.5)*; then we have the condition* (4.12) *hold.*

Proof. First, we show that $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ is invertible for all $x \in \mathbb{R}^{(n+1)d}$. By the definition of $\mathbf{S}$ in (7.5),

$$\mathbf{S}\mathbf{S}^\top = \begin{bmatrix} \mathbf{I}_d & \frac{1}{n}\nabla^2\phi_1(\alpha_1) & \cdots & \frac{1}{n}\nabla^2\phi_n(\alpha_n) \\ \frac{1}{n}\nabla^2\phi_1(\alpha_1) & & & \\ \vdots & & \mathbf{I}_{B_n}\mathbf{I}_{B_n}^\top + \mathbf{M} & \\ \frac{1}{n}\nabla^2\phi_n(\alpha_n) & & & \end{bmatrix},$$

where $\mathbf{M} = \{\mathbf{M}_{ij}\}_{1\le i\le n, 1\le j\le n}$ is divided into $n \times n$ contiguous blocks of size $d \times d$ with each block $\mathbf{M}_{ij}$ defined as the following:

$$\mathbf{M}_{ij} \overset{\text{def}}{=} \frac{1}{n}\nabla^2\phi_i(\alpha_i) \cdot \frac{1}{n}\nabla^2\phi_j(\alpha_j) \in \mathbb{R}^{d\times d} \qquad \text{and} \qquad \mathbf{M} \in \mathbb{R}^{nd\times nd}.$$

Taking the expectation over $\mathbf{S}$ w.r.t. the distribution $\mathcal{D}_{(w,\alpha_1,\cdots,\alpha_n)}$ gives

$$\begin{aligned} \mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right] &= \begin{bmatrix} \mathbf{I}_d & \frac{1}{n}\nabla^2\phi_1(\alpha_1) & \cdots & \frac{1}{n}\nabla^2\phi_n(\alpha_n) \\ \frac{1}{n}\nabla^2\phi_1(\alpha_1) & & & \\ \vdots & & \frac{\tau}{n}\mathbf{I}_{nd} + \mathbf{M} & \\ \frac{1}{n}\nabla^2\phi_n(\alpha_n) & & & \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_d \\ \frac{1}{n}\nabla^2\phi_1(\alpha_1) \\ \vdots \\ \frac{1}{n}\nabla^2\phi_n(\alpha_n) \end{bmatrix} \begin{bmatrix} \mathbf{I}_d \\ \frac{1}{n}\nabla^2\phi_1(\alpha_1) \\ \vdots \\ \frac{1}{n}\nabla^2\phi_n(\alpha_n) \end{bmatrix}^\top + \frac{\tau}{n}\begin{bmatrix} 0 & 0 \\ 0 & \mathbf{I}_{nd}. \end{bmatrix}, \end{aligned} \tag{7.11}$$

where $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ is symmetric, positive semidefinite. Let $(u; v_1; \cdots; v_n) \in \mathbb{R}^{(n+1)d}$ s.t.

$$(u; v_1; \cdots; v_n)^\top \mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right](u; v_1; \cdots; v_n) = 0.$$

From (7.11), we obtain

$$\left\| \left[\mathbf{I}_d; \frac{1}{n}\nabla^2\phi_1(\alpha_1); \cdots; \frac{1}{n}\nabla^2\phi_n(\alpha_n)\right]^\top [u; v_1; \cdots; v_n] \right\|^2 + \frac{\tau}{n}\sum_{i=1}^n \|v_i\|^2 = 0.$$

Since both terms are nonnegative, we obtain $\sum_{i=1}^n \|v_i\|^2 = 0 \implies$ for all $i, v_i = 0$, and then $u = 0$. This confirms that $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ is positive definite, thus invertible and $\mathbf{Ker}(\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]) = \{0\}$. Besides, from Lemma 7.2, we get $DF(x)$ invertible. Thus $F(x) \in \mathbf{Im}(DF(x)^\top)$ and $\mathbf{Ker}(DF(x)) = \{0\}$. We have that (4.17) holds. By Lemma 4.6, we have that (4.12) holds for all $x \in \mathbb{R}^{(n+1)d}$. ∎

From Lemma 7.4, we know that for *any* size of the subset sampling $|B_n| = \tau \in \{1, \cdots, n\}$, the condition (4.12) holds. The corresponding sketch size of $\mathbf{S}$ is $(\tau+1)d$.

8. Applications to GLMs—*tossing-coin-sketch* method. Consider the problem of training a GLM

$$w^* = \arg\min_{w\in\mathbb{R}^d} P(w) \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} \phi_i(a_i^\top w) + \frac{\lambda}{2}\|w\|^2, \tag{8.1}$$

where $\phi_i : \mathbb{R} \to \mathbb{R}^+$ is a convex and continuously twice differentiable loss function, $a_i \in \mathbb{R}^d$ are data samples, and $w \in \mathbb{R}^d$ is the parameter to optimize. As the objective function is strongly convex, the unique minimizer satisfies $\nabla P(w) = 0$, that is,

$$\nabla P(w) = \frac{1}{n}\sum_{i=1}^{n} \phi_i'(a_i^\top w)a_i + \lambda w = 0. \tag{8.2}$$

Let $\Phi(w) \overset{\text{def}}{=} \left[\phi_1'(a_1^\top w) \ \cdots \ \phi_n'(a_n^\top w)\right]^\top \in \mathbb{R}^n$ and $\mathbf{A} \overset{\text{def}}{=} \left[a_1 \ \cdots \ a_n\right] \in \mathbb{R}^{d\times n}$. By introducing auxiliary variables $\alpha_i \in \mathbb{R}$ s.t. $\alpha_i \overset{\text{def}}{=} -\phi_i'(a_i^\top w)$, we can rewrite (8.2) as

$$w \ = \ \frac{1}{\lambda n}\mathbf{A}\alpha \qquad \text{and} \qquad \alpha \ = \ -\Phi(w). \tag{8.3}$$

Note $x = [\alpha; w] \in \mathbb{R}^{n+d}$. The objective of finding the minimum of (8.1) is now equivalent to finding *zeros* for the function

$$F(x) \ = \ F(\alpha; w) \overset{\text{def}}{=} \begin{bmatrix} \frac{1}{\lambda n}\mathbf{A}\alpha - w \\ \alpha + \Phi(w) \end{bmatrix}, \tag{8.4}$$

where $F : \mathbb{R}^{n+d} \to \mathbb{R}^{n+d}$. Our objective now becomes solving $F(x) = 0$ with $p = m = n + d$. For this, we will use a variant of the SNR. The advantage in representing (8.2) as the nonlinear system (8.4) is that we now have one row per data point (see the second equation in (8.3)). This allows us to use sketching to *subsample* the data.

Since the function F has a block structure, we will use a structured sketching matrix, which we refer to as a *tossing-coin-sketch*. But first, we need the following definition of a block sketch.

DEFINITION 8.1 ((n,τ)–block sketch). *Let $B_n \subset \{1,\dots,n\}$ be a subset of size τ uniformly sampling at random. We say that $\mathbf{S} \in \mathbb{R}^{n\times\tau}$ is a (n,τ)–block sketch if $\mathbf{S} = \mathbf{I}_{B_n}$ where $\mathbf{I}_{B_n}$ denotes the column concatenation of the columns of the identity matrix $\mathbf{I}_n \in \mathbb{R}^{n\times n}$ whose indices are in B_n.*

Our tossing-coin-sketch is a sketch that alternates between two blocks depending on the result of a coin toss.

DEFINITION 8.2 (tossing-coin-sketch). *Let $\mathbf{S}_d \in \mathbb{R}^{d\times\tau_d}$ and $\mathbf{S}_n \in \mathbb{R}^{n\times\tau_n}$ be a (d,τ_d)–block sketch and a (n,τ_n)–block sketch, respectively. Let $b \in (0;1)$. Now each time we sample $\mathbf{S}$, we "toss a coin" to determine the structure of $\mathbf{S} \in \mathbb{R}^{(d+n)\times(\tau_d+\tau_n)}$. That is, $\mathbf{S} = \left[\begin{smallmatrix} \mathbf{S}_d & 0 \\ 0 & 0 \end{smallmatrix}\right]$ with probability $1-b$ and $\mathbf{S} = \left[\begin{smallmatrix} 0 & 0 \\ 0 & \mathbf{S}_n \end{smallmatrix}\right]$ with probability b.*

By applying the SNR method with a tossing-coin-sketch for solving (8.4), we arrive at an efficient method for solving (8.1) that we call the *TCS* method. By using a tossing-coin-sketch, we can alternate between solving a linear system based on the first d rows of (8.4) and a nonlinear system based on the last n rows of (8.4).

TCS is inspired by the first-order stochastic dual ascent methods [51, 50, 46]. Indeed, (8.3) can be seen as primal-dual systems with primal variables w and dual variables α. Stochastic dual ascent methods are efficient to solve (8.3). At each

iteration, they update alternatively the primal and the dual variables w and α with the first-order informations. Thus, by sketching alternatively the primal and the dual systems and updating accordingly with the Newton-type steps, TCS's updates can be seen as the second-order stochastic dual ascent methods.

We show in the next section that the TCS method verifies (4.12). Using sketch sizes s.t. $\tau_n \ll n$, the TCS method has the same cost as `SGD` in the case $d \ll n$. The low computational cost per iteration is thus another advantage of the TCS method. See Appendix B the complexity analysis. For a detailed derivation and implementation of the TCS method which is straightforward, please refer to Appendix F and Algorithm 4 in Appendix G in [59] for more details.

8.1. The condition (4.12) in the case of the TCS method. In this section, we show that the TCS method verifies (4.12) through Lemma 4.6 in the following.

LEMMA 8.3. *Consider the function F defined in* (8.4) *and the tossing-coin-sketch* $\mathbf{S}$ *defined in Definition* 8.2*; then* (4.12) *holds.*

Proof. First, we show that $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ is invertible. By Definition 8.2, it is straightforward to verify that

$$\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right] = \begin{bmatrix} \frac{(1-b)\tau_d}{n}\mathbf{I}_d & 0 \\ 0 & \frac{b\tau_n}{n}\mathbf{I}_n \end{bmatrix}$$

is invertible and $\mathbf{Ker}(\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]) = \{0\}$. Now we show the Jacobian $DF^\top(x)$ invertible. Let $x = [\alpha; w] \in \mathbb{R}^{n+d}$ with $\alpha \in \mathbb{R}^n$ and $w \in \mathbb{R}^d$. Then $DF(x)$ is written as

$$DF(x)^\top = \begin{bmatrix} \frac{1}{\lambda n}\mathbf{A} & -\mathbf{I}_d \\ \mathbf{I}_n & \nabla\Phi(w)^\top \end{bmatrix}, \tag{8.5}$$

where $\nabla\Phi(w)^\top = \mathbf{Diag}\left(\phi_1''(a_1^\top w), \ldots, \phi_n''(a_n^\top w)\right)\mathbf{A}^\top \in \mathbb{R}^{n\times d}$. Denote the diagonal matrix $D(w) \overset{\text{def}}{=} \mathbf{Diag}\left(\phi_1''(a_1^\top w), \ldots, \phi_n''(a_n^\top w)\right)$. Since ϕ_i is continuously twice differentiable and convex, $\phi_i''(a_i^\top w) \geq 0$ for all i. Thus, $D(w) \geq 0$.

Let $(u; v) \in \mathbb{R}^{n+d}$ with $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^d$ such that $DF(x)^\top[u; v] = 0$. We have

$$DF(x)^\top \begin{bmatrix} u \\ v \end{bmatrix} = 0 \overset{(8.5)}{\Longleftrightarrow} \begin{bmatrix} \frac{1}{\lambda n}\mathbf{A} & -\mathbf{I}_d \\ \mathbf{I}_n & \nabla\Phi(w)^\top \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = 0 \Longrightarrow \left(\mathbf{I}_n + \frac{1}{\lambda n} D(w)\mathbf{A}^\top\mathbf{A}\right) u = 0. \tag{8.6}$$

If $D(w)$ is invertible, (8.6) becomes

$$D(w)\left(D(w)^{-1} + \frac{1}{\lambda n}\mathbf{A}^\top\mathbf{A}\right) u = 0 \Longleftrightarrow \left(D(w)^{-1} + \frac{1}{\lambda n}\mathbf{A}^\top\mathbf{A}\right) u = 0. \tag{8.7}$$

Since $D(w)$ is invertible, i.e., $D(w) > 0$, we obtain $D(w)^{-1} > 0$. As $\frac{1}{\lambda n}\mathbf{A}^\top\mathbf{A} \geq 0$, we get $D(w)^{-1} + \frac{1}{\lambda n}\mathbf{A}^\top\mathbf{A} > 0$, thus invertible. From (8.7), we get $u = 0$.

Otherwise, $D(w)$ is not invertible. Without losing generality, we assume that $\phi_1''(a_1^\top w) \geq \phi_2''(a_2^\top w) \geq \cdots \geq \phi_n''(a_n^\top w) = 0$. Let j be the largest index for which $\phi_j''(a_j^\top w) > 0$. If j does not exist, then $D(w) = 0$. From (8.6), we get $u = 0$ directly. If j exists, we have $1 \leq j < n$ and

$$\begin{aligned} D(w)\mathbf{A}^\top\mathbf{A} &= \mathbf{Diag}\left(\phi_1''(a_1^\top w), \ldots, \phi_j''(a_j^\top w), 0, \ldots, 0\right)\mathbf{A}^\top\mathbf{A} \\ &= \begin{bmatrix} \mathbf{Diag}\left(\phi_1''(a_1^\top w), \ldots, \phi_j''(a_j^\top w)\right)\mathbf{A}_{1:j}^\top\mathbf{A}_{1:j} & 0 \\ 0 & 0 \end{bmatrix}, \end{aligned} \tag{8.8}$$

TABLE 1
Details of the data sets for binary classification.

Dataset	Dimension (d)	Samples (n)	C.N. of the model	L
covetype	54	581012	7.45×10^{12}	1.28×10^{7}
a9a	123	32561	5.12×10^{4}	1.57
fourclass	2	862	4.86×10^{6}	5.66×10^{3}
artificial	50	10000	3.91×10^{4}	3.91
ijcnn1	22	49990	2.88×10^{3}	5.77×10^{-2}
webspam	254	350000	7.47×10^{4}	2.13×10^{-1}
epsilon	2000	400000	3.51×10^{4}	8.76×10^{-2}
phishing	68	11055	1.04×10^{3}	9.40×10^{-2}

where $\mathbf{A}_{1:j} \overset{\text{def}}{=} [a_1 \ \cdots \ a_j] \in \mathbb{R}^{d\times j}$. Note $u = [u_1; \cdots; u_n] \in \mathbb{R}^n$. Plugging (8.8) into (8.6), we get

$$\left(\mathbf{I}_n + \frac{1}{\lambda n}\begin{bmatrix}\mathbf{Diag}\left(\phi_1''(a_1^\top w), \ldots, \phi_j''(a_j^\top w)\right)\mathbf{A}_{1:j}^\top \mathbf{A}_{1:j} & 0 \\ 0 & 0\end{bmatrix}\right) u = 0$$

$$\iff \begin{cases} \left(\mathbf{I}_j + \frac{1}{\lambda n}\mathbf{Diag}\left(\phi_1''(a_1^\top w), \ldots, \phi_j''(a_j^\top w)\right)\mathbf{A}_{1:j}^\top \mathbf{A}_{1:j}\right) u_{1:j} = 0 \\ u_{(j+1):n} = 0 \end{cases}, \tag{8.9}$$

where $u_{1:j} \overset{\text{def}}{=} [u_1; \cdots; u_j] \in \mathbb{R}^j$ and $u_{(j+1):n} \overset{\text{def}}{=} [u_{j+1}; \cdots; u_n] \in \mathbb{R}^{n-j}$. From (8.9), $u_{(j+1):n} = 0$. Now $\mathbf{Diag}\left(\phi_1''(a_1^\top w), \ldots, \phi_j''(a_j^\top w)\right)$ is invertible in the subspace $\mathbb{R}^j$ as every coordinate in the diagonal $\phi_i''(a_i^\top w)$ is strictly positive for all $1 \le i \le j$. Similarly, we obtain $u_{1:j} = 0$ from the first equation of (8.9). Overall we get $u = 0$.

Thus, in all cases, $u = 0$, then $v = \frac{1}{\lambda n} Au = 0$. We can thus induce that $DF(\alpha; w)^\top$ is invertible for all α and w. Similar to Lemma 7.4, we have that (4.17) holds, and by Lemma 4.6, we have that (4.12) holds. ∎

From Lemma 8.3, we know that for *any* size of the block sketch $\tau_d \in \{1, \ldots, d\}$ and $\tau_n \in \{1, \cdots, n\}$, (4.12) holds. The corresponding sketch size of $\mathbf{S}$ is $\tau_d + \tau_n$.

8.2. Experiments for TCS method applied for GLM. We consider the logistic regression problem with eight datasets[10] taken from LibSVM [10], except for one artificial dataset. Table 1 provides the details of these datasets, including the *condition number* (C.N.) of the model and the smoothness constant L of the model. The C.N. of the logistic regression problem is given by C.N. $\overset{\text{def}}{=} \frac{\lambda_{\max}(\mathbf{A}\mathbf{A}^\top)}{4n\lambda} + 1$, where $\lambda_{\max}(\cdot)$ is the largest eigenvalue operator. The smoothness constant L is given by $L \overset{\text{def}}{=} \frac{\lambda_{\max}(\mathbf{A}\mathbf{A}^\top)}{4n} + \lambda$. As for the logistic regression problem, we consider the loss function ϕ_i in (8.1) in the form $\phi_i(t) = \ln(1 + \mathrm{e}^{-y_i t})$ where y_i are the target values for $i = 1, \ldots, n$.

The artificial dataset. The artificial dataset $\mathbf{A}^\top \in \mathbb{R}^{n\times d}$ in Table 1 is of size 10000×50 and is generated by a Gaussian distribution whose mean is zero and covariance is a Toeplitz matrix. Toeplitz matrices are completely determined by their diagonal. We set the diagonal of our Toeplitz matrix as $[c^0; c^1; \cdots; c^{d-1}] \in \mathbb{R}^d$ where $c \in \mathbb{R}^+$ is a parameter. We choose $c = 0.9$ (closed to 1) which results in $\mathbf{A}$ having highly correlated columns, which in turn makes $\mathbf{A}$ an ill-conditioned dataset. We set the ground truth coefficients of the model $\mathbf{w} = [(-1)^0 \cdot \mathrm{e}^{-\frac{0}{10}}; \cdots; (-1)^{d-1} \cdot \mathrm{e}^{-\frac{d-1}{10}}] \in \mathbb{R}^d$

[10] All datasets except for the artificial dataset can be found on https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/. Some of the datasets can be found in [30, 6, 38, 55, 16].

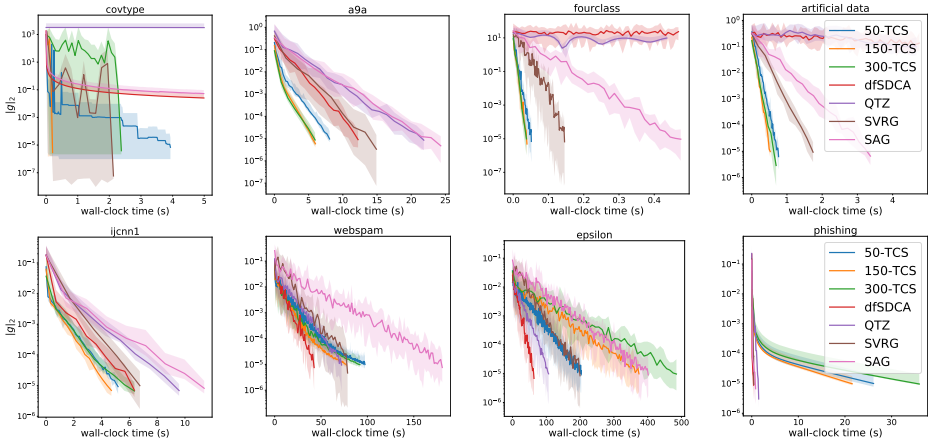

FIG. 1. *Experiments for TCS method applied for GLM. Color available online.*

TABLE 2
Details of the parameters' choices (γ and b) for 50-TCS, 150-TCS, and 300-TCS.

		50-TCS	150-TCS	300-TCS
Dataset	Stepsize	Bernoulli	Bernoulli	Bernoulli
covetype	1.0	$\frac{n}{n+\tau_n*3}$	$\frac{n}{n+\tau_n*3}$	$\frac{n}{n+\tau_n*3}$
a9a	1.5	$\frac{n}{n+\tau_n} - 0.03$	$\frac{n}{n+\tau_n} - 0.03$	$\frac{n}{n+\tau_n} - 0.11$
fourclass	1.0	$\frac{n}{n+\tau_n} - 0.11$	$\frac{n}{n+\tau_n} - 0.11$	$\frac{n}{n+\tau_n} - 0.11$
artificial	1.0	$\frac{n}{n+\tau_n} - 0.03$	$\frac{n}{n+\tau_n} - 0.11$	$\frac{n}{n+\tau_n} - 0.11$
ijcnn1	1.8	$\frac{n}{n+\tau_n} - 0.03$	$\frac{n}{n+\tau_n} - 0.11$	$\frac{n}{n+\tau_n} - 0.11$
webspam	1.8	$\frac{n}{n+\tau_n*3}$	$\frac{n}{n+\tau_n*3}$	$\frac{n}{n+\tau_n*3}$
epsilon	1.8	$\frac{n}{n+\tau_n*3}$	$\frac{n}{n+\tau_n*3}$	$\frac{n}{n+\tau_n*3}$
phishing	1.8	$\frac{n}{n+\tau_n} - 0.03$	$\frac{n}{n+\tau_n} - 0.11$	$\frac{n}{n+\tau_n} - 0.11$

and the target values of the dataset $\mathbf{y} = \operatorname{sgn}\left(\mathbf{A}^\top \mathbf{w} + \mathbf{r}\right) \in \mathbb{R}^n$ where $\mathbf{r} \in \mathbb{R}^n$ is the noise generated from a standard normal distribution.

We compare the TCS method with SAG [49], SVRG [26], dfSDCA [50], and Quartz [46]. All experiments were initialized at $w^0 = 0 \in \mathbb{R}^d$ (and/or $\alpha^0 = 0 \in \mathbb{R}^n$ for TCS/dfSDCA methods). For all methods, we used the stepsize that was shown to work well in practice. For instance, the common rule of thumb for SAG and SVRG is to use a stepsize $\frac{1}{L}$, where L is the smoothness constant. This rule of thumb stepsize is not supported by theory. Indeed for SAG, the theoretical stepsize is $\frac{1}{16L}$ and it should be even smaller for SVRG depending on the C.N. For dfSDCA and Quartz, we used the stepsize suggested in the experiments in [50] and [46], respectively. For TCS, we used two types of stepsize, related to the C.N. of the model. If the C.N. is big (Figure 1, top row), we used $\gamma = 1$ except for a9a with $\gamma = 1.5$. If the C.N. is small (Figure 1, bottom row), we used $\gamma = 1.8$. We also set the Bernoulli parameter b (probability of the coin toss) depending on the size of the dataset (see Table 2 in Appendix B), and $\tau_d = d$. We tested three different sketch sizes $\tau_n = 50, 150, 300$. More details of the parameter settings are presented in Appendix B.

We used $\lambda = \frac{1}{n}$ regularization parameter, evaluated each method 10 times, and stopped once the gradient norm[11] was below 10^{-5} or some maximum time had been

[11] We evaluated the true gradient norm every 1000 iterations. We also paused the timing when computing the performance evaluation of the gradient norm.

reached. In Figure 1, we plotted the central tendency as a solid line and all other executions as a shaded region for the wall-clock time versus gradient norm.

From Figure 1, TCS outperforms all other methods on ill-conditioned problems (Figure 1, top row), but not always the case on well-conditioned problems (Figure 1, bottom row). This is because in ill-conditioned problems, the curvature of the optimization function is not uniform over directions and varies in the input space. Second-order methods effectively exploit information of the local curvature to converge faster than first-order methods. To further illustrate the performance of TCS on ill-conditioned problems, we compared the performance of TCS on the artificial dataset in the top right of Figure 1. Note as well that for reaching an approximate solution at an early stage (i.e., $tol = 10^{-3}, 10^{-4}$), TCS is very competitive on all problems. TCS also has the smallest variance compared to the first-order methods based on eyeballing the shaded error bars in Figure 1, especially compared to SVRG. Among the three tested sketch sizes, 150 performed the best except on the *epsilon* dataset.

9. Conclusion and future work. We introduced the SNR method, for which we provided strong convergence guarantees. We also developed several promising applications of SNR to show that SNR is very flexible and tested one of these specialized variants for training GLMs. SNR is flexible by the fact that its primitive goal is to solve efficiently nonlinear equations. Since there are many ways to rewrite an optimization problem as nonlinear equations, each rewrite leads to a distinct method, thus leads to a specific implementation in practice (e.g., SNM, TCS methods) when using SNR. Besides, the convergence theories presented in section 4 guarantee a large variety of choices for the sketch. This flexibility allows us to discover many applications of SNR and their induced consequences, especially providing new global convergence theories. As such, we believe that SNR and its global convergence theory will open the way to designing and analyzing a host of new stochastic second-order methods. Further venues of investigation include exploring the use of adaptive norms for projections and leveraging efficient sketches (e.g., the fast Johnson–Lindenstrauss sketch [45], sketches with determinantal sampling [39]) to design even faster variants of SNR or cover other stochastic second-order methods. Since SNR can be seen as SGD, it might be possible to design and develop efficient accelerated SNR or SNR with momentum methods. On the experimental side, it would be interesting to apply our method to the training of deep neural networks.

Appendix A. Sufficient conditions for reformulation assumption (3.3)**.** To give sufficient conditions for (3.3) to hold, we need to study the spectra of $\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(x)\right]$. The expected matrix $\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(x)\right]$ has made an appearance in several references [19, 39, 14] in different contexts and with different sketches. We build upon some of these past results and adapt them to our setting.

First note that (3.3) holds if $\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(x)\right]$ is invertible. The invertibility of $\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(x)\right]$ was already studied in detail in the linear setting in Theorem 3 in [23] when $\mathbf{S}$ is sampled from a discrete distribution. Here we can state a sufficient condition of (3.3) for sketching matrices that have a continuous distribution.

LEMMA A.1. *For every $x \in \mathbb{R}^p$, if $\mathbb{E}_{\mathbf{S}\sim\mathcal{D}_x}\left[\mathbf{S}\mathbf{S}^\top\right]$ and $DF(x)^\top DF(x)$ are invertible, then $\mathbb{E}_{\mathbf{S}\sim\mathcal{D}_x}\left[\mathbf{H}_{\mathbf{S}}(x)\right]$ is invertible.*

Proof. Let $x \in \mathbb{R}^p$ and $\mathbf{S} \sim \mathcal{D}_x$. Let $\mathbf{G} = DF(x)^\top DF(x)$ which is thus symmetric positive definite and $\mathbf{W} = \mathbf{S}^\top$. In this case, since $\mathbf{G}$ is invertible we have that $\mathbf{Ker}(\mathbf{G}) = \{0\} \subset \mathbf{Ker}(\mathbf{W})$ verified, and by Lemma 4.4, we have that

$$\mathbf{Ker}\left(\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)^\dagger\right) = \mathbf{Ker}\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right) = \mathbf{Ker}(\mathbf{S}). \tag{A.1}$$

Consequently, using Lemma 4.4 again with $\mathbf{G} = \left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)^\dagger$, $\mathbf{W} = \mathbf{S}$, and $\mathbf{Ker}(\mathbf{G}) \subset \mathbf{Ker}(\mathbf{W})$ given by (A.1), we have that

(A.2)
$$\mathbf{Ker}(\mathbf{H}_{\mathbf{S}}(x)) = \mathbf{Ker}\left(\mathbf{S}\left(\mathbf{S}^\top DF(x)^\top DF(x)\mathbf{S}\right)^\dagger \mathbf{S}^\top\right) = \mathbf{Ker}(\mathbf{S}^\top) = \mathbf{Ker}(\mathbf{S}\mathbf{S}^\top).$$

Following the same steps in the proof of Lemma 4.6 right after (4.19), we obtain

$$\mathbf{Ker}(\mathbb{E}\left[\mathbf{H}_{\mathbf{S}}(x)\right]) = \bigcap_{\mathbf{S}\sim\mathcal{D}_x} \mathbf{Ker}(\mathbf{H}_{\mathbf{S}}(x)) \overset{(A.2)}{=} \bigcap_{\mathbf{S}\sim\mathcal{D}_x} \mathbf{Ker}(\mathbf{S}\mathbf{S}^\top) = \mathbf{Ker}(\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]) = \{0\},$$

where the last equality follows as $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ is invertible, which concludes the proof. ∎

The invertibility of $\mathbb{E}\left[\mathbf{S}\mathbf{S}^\top\right]$ states that the sketching matrices need to "span every dimension of the space" in expectation. This is the case for Gaussian and subsampling sketches which are shown in Lemma 4.8. This is also the case for our applications `SNM` and TCS which are shown in the proofs of Lemmas 7.4 and 8.3, respectively.

As for the invertibility of $DF(x)^\top DF(x) \in \mathbb{R}^{m\times m}$, this imposes that $DF(x)$ has full-column rank for all $x \in \mathbb{R}^p$, thus $m \leq p$. This excludes the regime of solving $F(x) = 0$ with $m > p$. However, our applications `SNM` and TCS also satisfy this condition and are again shown in the proofs of Lemmas 7.4 and 8.3, respectively.

Consequently, by Lemma A.1, we have that `SNM` and TCS satisfy (3.3).

Appendix B. Additional experimental details and the complexity. See Table 2 for the parameters we chose for TCS in the experiments in Figure 1. Such choices are due to TCS's cost per iteration which involves the feature dimension d, the number of the data samples n, the sketch sizes (τ_d, τ_n), and the Bernoulli parameter b. Here we only consider datasets with $d \ll n$.

It is beneficial to first understand TCS's cost in the simple setting where $\tau_d = \tau_n = 1$. The cost per iteration is stochastic and depends on the nature of the sketch.

When performing the updates (1.3) with (d, τ_d)–block sketch, we sketch the first d rows of (8.4). As $\tau_d = 1$, let $\mathbf{S}_d = e'_j \in \mathbb{R}^{d\times 1}$ with e'_j the jth unit coordinate in $\mathbb{R}^d$. For $\mathbf{S}_k = \left[\begin{smallmatrix}\mathbf{S}_d & 0\\ 0 & 0\end{smallmatrix}\right]$ with $\mathbf{S}_d = e'_j$ at the kth iteration, we get

$$\alpha_i^{k+1} = \alpha_i^k - \gamma \cdot \frac{\frac{\mathbf{A}_{ji}}{\lambda n}\left(\frac{1}{\lambda n}\left[\mathbf{A}\alpha^k\right]_j - w_j^k\right)}{\frac{1}{\lambda^2 n^2}[\mathbf{A}\mathbf{A}^\top]_{jj} + 1} \quad \text{for all } i = 1, \ldots, n, \tag{B.1}$$

$$w_j^{k+1} = w_j^k + \gamma \cdot \frac{\frac{1}{\lambda n}\left[\mathbf{A}\alpha^k\right]_j - w_j^k}{\frac{1}{\lambda^2 n^2}[\mathbf{A}\mathbf{A}^\top]_{jj} + 1}. \tag{B.2}$$

The cost of this iteration can be $\mathcal{O}(n)$ with n coordinates' updates of the auxiliary variable α. Indeed, the term $\frac{1}{\lambda^2 n^2}[\mathbf{A}\mathbf{A}^\top]_{jj}$ is precomputed and stored in the fixed matrix $\frac{1}{\lambda^2 n^2}\mathbf{A}\mathbf{A}^\top \in \mathbb{R}^{d\times d}$. We also introduce an auxiliary variable $\overline{\alpha}^k$ to keep tracking the term $\overline{\alpha}^k = \frac{1}{\lambda n}\mathbf{A}\alpha^k \in \mathbb{R}^d$. The vector $\overline{\alpha}^k$ can be efficiently updated by

$$\overline{\alpha}^{k+1} = \overline{\alpha}^k - \gamma \cdot \frac{\overline{\alpha}_j^k - w_j^k}{\frac{1}{\lambda^2 n^2}[\mathbf{A}\mathbf{A}^\top]_{jj} + 1} \cdot [\mathbf{A}\mathbf{A}^\top]_j, \tag{B.3}$$

where $[\mathbf{A}\mathbf{A}^\top]_j$ is the jth column of $\mathbf{A}\mathbf{A}^\top$. Thus the update of $\overline{\alpha}^k$ costs $\mathcal{O}(d)$. Notice that we only update one single coordinate of w from this sketch, i.e., w_j.

Alternatively, when performing the updates (1.3) with (n, τ_n)–block sketch, we sketch the last n rows of (8.4). For $\mathbf{S}_k = \left[\begin{smallmatrix} 0 & 0 \\ 0 & \mathbf{S}_n \end{smallmatrix}\right]$ with $\mathbf{S}_n = e_i$ the ith unit coordinate in $\mathbb{R}^n$ at kth iteration, we get

$$\alpha_i^{k+1} = \alpha_i^k - \gamma \cdot \frac{\alpha_i^k + \phi_i'(a_i^\top w^k)}{\|a_i\|_2^2 \phi_i''(a_i^\top w^k)^2 + 1}, \tag{B.4}$$

$$w^{k+1} = w^k - \gamma \cdot \frac{\alpha_i^k + \phi_i'(a_i^\top w^k)}{\|a_i\|_2^2 \phi_i''(a_i^\top w^k)^2 + 1} \cdot \phi_i''(a_i^\top w^k) a_i, \tag{B.5}$$

$$\overline{\alpha}^{k+1} = \overline{\alpha}^k - \gamma \cdot \frac{\alpha_i^k + \phi_i'(a_i^\top w^k)}{\|a_i\|_2^2 \phi_i''(a_i^\top w^k)^2 + 1} \cdot \frac{1}{\lambda n} a_i. \tag{B.6}$$

The cost of this iteration is $\mathcal{O}(d)$ with the full coordinates' updates of $w, \overline{\alpha}$ and a single coordinate update of α. If we choose $b = n/(n+d)$ to sample one row of (8.4) uniformly, the cost per iteration in expectation will be

$$\text{Cost(update TCS)} = \mathcal{O}(n) * (1-b) + \mathcal{O}(d) * b = \mathcal{O}(nd/(n+d)) = \mathcal{O}(\min(n, d)). \tag{B.7}$$

Consequently, the TCS method on average has the same cost as `SGD`, i.e., $\mathcal{O}(d)$ in the case $d \ll n$. Increasing τ_d and τ_n drops significantly the number of iterations, but increases the total cost per iteration. Thus there is a trade-off between increasing the sketch sizes and keeping the total cost per iteration low. For the total cost in general with different choices of τ_d, τ_n, and b, please refer to Appendix H in [59] for more details.

Different to our global convergence theories, in practice, choosing constant stepsize $\gamma > 1$ may converge faster for certain datasets. Here we need to be careful that the stepsize we mentioned is the stepsize used for (n, τ_n)–block sketch. As for (d, τ_d)–block sketch, we always choose $\gamma = 1$, which solves exactly the linear system. In our experiments, we found that the choice of the stepsize is related to the C.N. of the model. If the dataset is ill-conditioned with a big C.N., $\gamma = 1$ is a good choice (Figure 1, top row, except for a9a); if the dataset is well-conditioned with a small C.N., all $\gamma \in (1, 1.8]$ still converges. In practice, $\gamma = 1.8$ is a good choice for well-conditioned datasets (Figure 1, bottom row). For a9a, we did a grid search for the stepsize. To avoid tuning the stepsizes, it is possible to apply a stochastic line search process [54] in our method TCS without increasing its complexity. Please refer to Appendix I in [59] for more details.

REFERENCES

[1] N. Agarwal, B. Bullins, and E. Hazan, *Second-order stochastic optimization for machine learning in linear time*, J. Mach. Learn. Res., 18 (2017), pp. 1–40.

[2] N. Ailon and B. Chazelle, *The fast Johnson-Lindenstrauss transform and approximate nearest neighbors*, SIAM J. Comput., 39 (2009), pp. 302–322.

[3] H. An and Z. Bai, *A globally convergent Newton-GMRES method for large sparse systems of nonlinear equations*, Appl. Numer. Math., 57 (2007), pp. 235–252.

[4] F. Bach, *Learning Theory from First Principles*, MIT Press, Cambridge, MA, to appear.

[5] S. Bellavia and B. Morini, *A globally convergent Newton-GMRES subspace method for systems of nonlinear equations*, SIAM J. Sci. Comput., 23 (2001), pp. 940–960.

[6] J. Blackard and D. Dean, *Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables*, Computers Electronics Agriculture, 24 (1999), pp. 131–151.

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

[8] D. Calandriello, A. Lazaric, and M. Valko, *Efficient second-order online kernel learning with adaptive embedding*, in Advances in Neural Information Processing Systems 30, 2017, pp. 6140–6150.
[9] C. Cartis, N. I. M. Gould, and P. L. Toint, *Adaptive cubic regularisation methods for unconstrained optimization. Part I: Motivation, convergence and numerical results*, Math. Program., 127 (2009), pp. 1–38.
[10] C.-C. Chang and C.-J. Lin, *LIBSVM: A library for support vector machines*, ACM Trans. Intelligent Systems Technology, 2 (2011), pp. 27:1–27:27.
[11] B. Christianson, *Automatic Hessians by reverse accumulation*, IMA J. Numer. Anal., 12 (1992), pp. 135–150.
[12] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust-Region Methods*, SIAM, Philadelphia, 2000.
[13] A. Defazio, F. Bach, and S. Lacoste-julien, *SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives*, in Advances in Neural Information Processing Systems 27, 2014.
[14] M. Derezinski, F. T. Liang, Z. Liao, and M. W. Mahoney, *Precise expressions for random projections: Low-rank approximation and randomized Newton*, in Advances in Neural Information Processing Systems, 2020.
[15] P. Deuflhard, *Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms*, Springer, New York, 2011.
[16] D. Dua and C. Graff, *UCI Machine Learning Repository*, http://archive.ics.uci.edu/ml, 2017.
[17] M. A. Erdogdu and A. Montanari, *Convergence rates of sub-sampled Newton methods*, in Advances in Neural Information Processing Systems 28, 2015, pp. 3052–3060.
[18] W. Gao and D. Goldfarb, *Quasi-Newton methods: Superlinear convergence without line searches for self-concordant functions*, Optim. Methods Softw., 34 (2019), pp. 194–217.
[19] R. Gower, D. Koralev, F. Lieder, and P. Richtarik, *RSN: Randomized subspace Newton*, in Advances in Neural Information Processing Systems 32, 2019, pp. 614–623.
[20] R. M. Gower, D. Goldfarb, and P. Richtárik, *Stochastic block BFGS: Squeezing more curvature out of data*, in Proceedings of the 33rd International Conference on Machine Learning, 2016.
[21] R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik, *SGD: General analysis and improved rates*, in Proceedings of the 36th International Conference on Machine Learning, vol. 97, 2019, pp. 5200–5209.
[22] R. M. Gower and P. Richtárik, *Randomized iterative methods for linear systems*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 1660–1690.
[23] R. M. Gower and P. Richtárik, *Stochastic Dual Ascent for Solving Linear Systems*, arXiv:1512.06890, 2015.
[24] M. Gürbüzbalaban, A. Ozdaglar, and P. Parrilo, *A globally convergent incremental Newton method*, Math. Program., 151 (2015), 283313.
[25] O. Hinder, A. Sidford, and N. Sohoni, *Near-optimal methods for minimizing star-convex functions and beyond*, in Proceedings of the 33rd Conference on Learning Theory, 2020, pp. 1894–1938.
[26] R. Johnson and T. Zhang, *Accelerating stochastic gradient descent using predictive variance reduction*, in Advances in Neural Information Processing Systems 26, 2013, pp. 315–323.
[27] M. S. Kaczmarz, *Angenäherte auflösung von systemen linearer gleichungen*, Bulletin International de l'Académie Polonaise des Sciences et des Lettres. Série A, Sciences Mathématiques, 35 (1937), pp. 355–357.
[28] L. Kantorovitch, *The method of successive approximation for functional equations*, Acta Math., 71 (1939), pp. 63–97.
[29] C. T. Kelley, *Numerical methods for nonlinear equations*, Acta Numer., 27 (2018), 207287.
[30] R. Kohavi, *Scaling up the accuracy of naive-Bayes classifiers: A decision-tree hybrid*, in Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, 1996.
[31] J. M. Kohler and A. Lucchi, *Sub-sampled cubic regularization for non-convex optimization*, in Proceedings of the 34th International Conference on Machine Learning, vol. 70, 2017, pp. 1895–1904.
[32] D. Kovalev, K. Mishchenko, and P. Richtarik, *Stochastic Newton and Cubic Newton Methods with Simple Local Linear-Quadratic Rates*, arXiv:1912.01597, 2019.
[33] J. C. H. Lee and P. Valiant, *Optimizing star-convex functions*, in Proceedings of the IEEE 57th Annual Symposium on Foundations of Computer Science, I. Dinur, ed., 2016, pp. 603–614.
[34] A. Leitão and B. F. Svaiter, *On projective Landweber–Kaczmarz methods for solving systems of nonlinear ill-posed equations*, Inverse Problems, 32 (2016), 025004.

[35] S. Lu, Z. Wei, and L. Li, *A trust region algorithm with adaptive cubic regularization methods for nonsmooth convex minimization*, Comput. Optim. Appl., 51 (2010), pp. 551–573.
[36] H. Luo, A. Agarwal, N. Cesa-Bianchi, and J. Langford, *Efficient second order online learning by sketching*, in Advances in Neural Information Processing Systems 29, 2016, pp. 902–910.
[37] S. Ma, R. Bassily, and M. Belkin, *The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning*, in Proceedings of the 35th International Conference on Machine Learning, 2018.
[38] R. Mohammad, F. Thabtah, and T. Mccluskey, *An assessment of features related to phishing websites using an automated technique*, in Proceedings of the International Conference for Internet Technology and Secured Transactions, 2012.
[39] M. Mutny, M. Dereziński, and A. Krause, *Convergence analysis of block coordinate algorithms with determinantal sampling*, in Proceedings of the International Conference on Artificial Intelligence and Statistics, 2020.
[40] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, 2nd ed., Springer, New York, 2014.
[41] Y. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*, Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.
[42] Y. E. Nesterov and B. T. Polyak, *Cubic regularization of Newton method and its global performance*, Math. Program., 108 (2006), pp. 177–205.
[43] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer Ser. Oper. Res. 43, Springer, New York, 1999.
[44] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, SIAM, Philadelphia, 2000.
[45] M. Pilanci and M. J. Wainwright, *Newton sketch: A near linear-time optimization algorithm with linear-quadratic convergence*, SIAM J. Optim., 27 (2017), pp. 205–245.
[46] Z. Qu, P. Richtárik, and T. Zhang, *Quartz: Randomized dual coordinate ascent with arbitrary sampling*, in Advances in Neural Information Processing Systems 28, 2015.
[47] P. Richtárik and M. Takáč, *Stochastic reformulations of linear systems: Algorithms and convergence theory*, SIAM J. Matrix Anal. Appl., to appear.
[48] F. Roosta-Khorasani and M. W. Mahoney, *Sub-sampled Newton Methods* I*: Globally Convergent Algorithms*, arXiv:1601.04737, 2016.
[49] M. Schmidt, N. Le Roux, and F. Bach, *Minimizing finite sums with the stochastic average gradient*, Math. Program., 162 (2017), pp. 83–112.
[50] S. Shalev-Shwartz, *SDCA without duality, regularization, and individual convexity*, in Proceedings of the 33rd International Conference on Machine Learning, 48, 2016, pp. 747–754.
[51] S. Shalev-Shwartz and T. Zhang, *Stochastic dual coordinate ascent methods for regularized loss*, J. Mach. Learn. Res., 14 (2013), pp. 567–599.
[52] T. Strohmer and R. Vershynin, *A randomized Kaczmarz algorithm with exponential convergence*, J. Fourier Anal. Appl., 15 (2009), pp. 262–278.
[53] S. Vaswani, F. Bach, and M. W. Schmidt, *Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron*, in Proceedings of the AISTATS 2019, pp. 1195–1204.
[54] S. Vaswani, A. Mishkin, I. Laradji, M. Schmidt, G. Gidel, and S. Lacoste-Julien, *Painless stochastic gradient: Interpolation, line-search, and convergence rates*, in Advances in Neural Information Processing Systems 32, 2019, pp. 3732–3745.
[55] D. Wang, D. Irani, and C. Pu, *Evolutionary study of web spam: Webb Spam Corpus* 2011 *versus Webb Spam Corpus* 2006, in Proceedings of the 8th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, 2012.
[56] D. P. Woodruff, *Sketching as a Tool for Numerical Linear Algebra*, preprint, arXiv:1411.4357, 2014.
[57] S. Wright and J. Nocedal, *Interior-point methods for nonlinear programming*, in Numerical Optimization, Springer, New York, 2006, pp. 563–597.
[58] Y.-X. Yuan, *Recent advances in numerical methods for nonlinear equations and nonlinear least squares*, Numer. Algebra Control Optim., 1 (2011), pp. 15–34.
[59] R. Yuan, A. Lazaric, and R. M. Gower, *Sketched Newton–Raphson*, https://arxiv.org/pdf/2006.12120.pdf.
[60] D. Zhou, P. Xu, and Q. Gu, *Stochastic variance-reduced cubic regularized Newton methods*, in Proceedings of the 35th International Conference on Machine Learning, vol. 80, 2018.
[61] W. Zhou, *On the convergence of the modified Levenberg–Marquardt method with a nonmonotone second order Armijo type line search*, J. Comput. Appl. Math. 239, (2013), pp. 152–161.

[62] W. Zhou and X. Chen, *Global convergence of a new hybrid Gauss–Newton structured BFGS method for nonlinear least squares problems*, SIAM J. Optim., 20 (2010), pp. 2422–2441.
[63] Y. Zhou, J. Yang, H. Zhang, Y. Liang, and V. Tarokh, *SGD converges to global minimum in deep learning via star-convex path*, in Proceedings of the International Conference on Learning Representations, 2019.