Regularized Stochastic BFGS algorithm

Aryan Mokhtari and Alejandro Ribeiro

***Abstract*—A regularized stochastic version of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton method is proposed to solve optimization problems with stochastic objectives that arise in large scale machine learning. Stochastic gradient descent is the currently preferred solution methodology but the number of iterations required to approximate optimal arguments can be prohibitive in high dimensional problems. BFGS modifies gradient descent by introducing a Hessian approximation matrix computed from finite gradient differences. This paper utilizes stochastic gradient differences and introduces a regularization to ensure that the Hessian approximation matrix remains well conditioned. The resulting regularized stochastic BFGS method is shown to converge to optimal arguments almost surely over realizations of the stochastic gradient sequence. Numerical experiments showcase reductions in convergence time relative to stochastic gradient descent algorithms and non-regularized stochastic versions of BFGS.**

I. Introduction

Many problems in machine learning involve minimizing an average cost written as a sum of individual costs associated with one out of a large number of data points [1]. E.g., in support vector machines (SVMs) we are given a training set $\mathcal{S} = \{(\boldsymbol{\theta}_i, y_i)\}_{i=1}^m$ containing m pairs $(\boldsymbol{\theta}_i, y_i)$ of feature vectors $\boldsymbol{\theta}_i \in \mathbb{R}^n$ and their corresponding class $y_i \in \{-1, 1\}$. The goal is to find a separating hyperplane supported by a vector $\mathbf{x}$ such that $\mathbf{x}^T\boldsymbol{\theta}_i \gtrless 0$ for points with $y_i = \pm 1$. Since this hyperplane may not exist or may not be unique we introduce a loss function $l((\boldsymbol{\theta}, y); \mathbf{x})$ and proceed to select as classifier the hyperplane with supporting vector

$$\mathbf{x}^* := \operatorname{argmin} \ \frac{\lambda}{2}\|\mathbf{x}\|^2 + \frac{1}{m}\sum_{i=1}^m l((\boldsymbol{\theta}_i, y_i); \mathbf{x}), \tag{1}$$

where $\lambda > 0$ is a regularization parameter. The vector $\mathbf{x}^*$ in (1) balances the minimization of the sum of distances to the separating hyperplane, as measured by the loss function $l((\boldsymbol{\theta}, y); \mathbf{x})$, with the minimization of the L_2 norm $\|\mathbf{x}\|_2$ to enforce desirable properties in $\mathbf{x}^*$ [1]. Common selections for the loss function are the hinge loss $l((\boldsymbol{\theta}, y); \mathbf{x}) = \max(0, 1 - y(\mathbf{x}^T\boldsymbol{\theta}))$ and the log loss $l((\boldsymbol{\theta}, y); \mathbf{x}) = \log(1 + \exp(-y(\mathbf{x}^T\boldsymbol{\theta})))$, e.g. [2].

The focus in recent years has shifted to large scale problems where the dimension of the vector $\mathbf{x}$ as well as the number of training samples m are very large. In such cases it is convenient to uncover the relationship with stochastic optimization problems by regarding m as infinite and invoking the law of large numbers to rewrite (1) as

$$\mathbf{x}^* := \operatorname*{argmin}_{\mathbf{x}} \mathbb{E}_{\boldsymbol{\theta}}[f(\mathbf{x}, \boldsymbol{\theta})] := \operatorname*{argmin}_{\mathbf{x}} F(\mathbf{x}). \tag{2}$$

In (2), we (re-)interpret $\boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^n$ as a random variable taking values in the convex set Θ according to an unknown probability distribution $m_{\boldsymbol{\theta}}(\boldsymbol{\theta})$. The feature vectors $\boldsymbol{\theta}_i$ in (1) are interpreted as samples of $\boldsymbol{\theta}$ and the loss functions $l((\boldsymbol{\theta}_i, y_i); \mathbf{x})$ as instantiations of the random function $f(\mathbf{x}, \boldsymbol{\theta})$. We refer to $f(\mathbf{x}, \boldsymbol{\theta})$ as the random or instantaneous functions and to $F(\mathbf{x}) := \mathbb{E}_{\boldsymbol{\theta}}[f(\mathbf{x}, \boldsymbol{\theta})]$ as the average function. Problems with the generic form in (2) are also common in optimal resource allocation problems in wireless systems [3], [4].

Descent algorithms can be used for the minimization of (2) when the objective function is convex. However, conventional descent methods require determination of the average gradient $\nabla_{\mathbf{x}} F(\mathbf{x}) = \mathbb{E}_{\boldsymbol{\theta}}[\nabla_{\mathbf{x}} f(\mathbf{x}, \boldsymbol{\theta})]$, which is intractable in general. Stochastic gradient descent algorithms overcome this issue by using unbiased gradient estimates based on small subsamples of data and are the workhorse methodology used to solve large-scale machine learning problems [2], [5], [6]. Useful though they are, gradient descent methods take a large number of iterations to converge. This problem is most acute when the variable dimension n is large as the condition number tends to increase with n. Developing stochastic Newton algorithms is not always possible because unbiased estimates of Newton steps are not easy to compute. Recourse to quasi-Newton methods then arises as a natural alternative. Indeed, quasi-Newton methods achieve superlinear convergence rates in deterministic settings while relying on gradients to compute curvature estimates [7], [8]. Since unbiased gradient estimates are computable at manageable cost, stochastic generalizations of quasi-Newton methods are realizable and expected to retain the convergence rate advantages of their deterministic counterparts [3], [9]. This expectation has been confirmed in numerical experiments for quadratic objectives [9].

The contribution of this paper is to develop a stochastic regularized version of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton method to solve (2). In BFGS the Hessian is approximated by a positive definite matrix that tracks the curvature of the last two iterates while being closest to the previous matrix in terms of differential entropy. It is well known that this approximation approaches a positive semidefinite matrix as the iteration index grows, but this is not a problem because the null eigenvector is perpendicular to the descent direction. In a stochastic setting, however, noise along the null eigenvector direction, which corresponds to an infinite in the inverse Hessian, is amplified by an arbitrary factor and results in a non convergent algorithm – see Section IV. Our regularization consists of modifying the differential entropy condition so that the approximant is a matrix with eigenvalues larger than a given lower bound. We show that this regularization guarantees almost sure convergence to the optimal argument $\mathbf{x}^*$ when the functions $f(\mathbf{x}, \boldsymbol{\theta})$ are strongly convex (Section III). Numerical experiments illustrate the improvement in convergence time relative to stochastic gradient descent algorithms and non-regularized stochastic versions of BFGS (Section IV).

II. Problem Formulation

Throughout the paper we assume that the functions $f(\mathbf{x}, \boldsymbol{\theta})$ are strongly convex. As a consequence, the objective function $F(\mathbf{x})$ in (2) is strongly convex and gradient descent algorithms can be used to find the optimal argument $\mathbf{x}^*$. To do so we need to compute gradients of the stochastic function $F(\mathbf{x})$ which according to (2) are given by

$$\mathbf{s}(\mathbf{x}) := \nabla_{\mathbf{x}} F(\mathbf{x}) = \mathbb{E}_{\theta}[\nabla_{\mathbf{x}} f(\mathbf{x}, \boldsymbol{\theta})]. \tag{3}$$

Since there are infinitely many functions $f(\mathbf{x}, \theta)$, exact evaluation of $\mathbf{s}(\mathbf{x})$ is not possible unless there is a closed form expression available for the expectation in (3). In practice, the number of functions is finite but very large and gradient computations are possible but impractical. Whether impossible or impractical we can avoid this problem by using stochastic gradients in lieu of actual gradients. For a given sample of L random variables $\tilde{\boldsymbol{\theta}} := [\boldsymbol{\theta}_1; ...; \boldsymbol{\theta}_L]$ drawn independently from the distribution of $\boldsymbol{\theta}$ we define the stochastic gradient at $\mathbf{x}$ given $\tilde{\boldsymbol{\theta}}$ as

$$\hat{\mathbf{s}}(\mathbf{x}, \tilde{\boldsymbol{\theta}}) := \frac{1}{L}\sum_{l=1}^L \nabla_{\mathbf{x}} f(\mathbf{x}, \boldsymbol{\theta}_l). \tag{4}$$

To compute the stochastic gradient $\hat{\mathbf{s}}(\mathbf{x}, \tilde{\boldsymbol{\theta}})$ we find the gradient of the random function $f(\mathbf{x}, \boldsymbol{\theta})$ for each $\boldsymbol{\theta}_l$ component of $\tilde{\boldsymbol{\theta}}$ and compute their average at manageable computational cost. Introducing now a time index t, an initial iterate $\mathbf{x}_1$, and a step size sequence ϵ_t the stochastic gradient descent algorithm is defined by the iteration

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \epsilon_t \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t). \tag{5}$$

Work supported by ARO W911NF-10-1-0388, NSF CAREER CCF-0952867, and ONR N00014-12-1-0997. The authors are with the Dept. of Electrical and Systems Eng., University of Pennsylvania, 200 S 33rd Street, Philadelphia, PA 19104. Email: {aryanm, aribeiro}@seas.upenn.edu.

Given that $\mathbb{E}_{\tilde{\boldsymbol{\theta}}}[\hat{\mathbf{s}}(\mathbf{x},\tilde{\boldsymbol{\theta}})] = \mathbf{s}(\mathbf{x})$, the stochastic gradient $\hat{\mathbf{s}}(\mathbf{x},\tilde{\boldsymbol{\theta}})$ in (4) is an unbiased estimate of the (average) gradient $\mathbf{s}(\mathbf{x})$ in (3). Thus, the iteration in (5) is such that, on average, iterates descend along a negative gradient direction. It is thus not surprising to learn that selecting the step size sequence to be nonsummable but square summable, i.e.,

$$\sum_{t=1}^{\infty} \epsilon_t = \infty, \quad \text{and} \quad \sum_{t=1}^{\infty} \epsilon_t^2 < \infty, \tag{6}$$

iterates $\mathbf{x}_t$ generated by (5) converge towards the optimal argument $\mathbf{x}^*$ if subsequent samples $\tilde{\boldsymbol{\theta}}$ are drawn independently. Selecting step sizes for which (6) holds is not difficult. A customary choice is to make $\epsilon_t = \epsilon_0\tau/(\tau+t)$, for given parameters ϵ_0 and τ that control the initial step size and its speed of decrease, respectively.

Convergence notwithstanding, the number of iterations required to approximate $\mathbf{x}^*$ can be prohibitive if the condition number of $F(\mathbf{x})$ is large as is common in large dimensional problems. To reduce the number of iterations required by (5) we resort to quasi-Newton methods whereby gradient descent directions are premultiplied by a matrix $\mathbf{B}_t^{-1}$,

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \epsilon_t \mathbf{B}_t^{-1}\mathbf{s}(\mathbf{x}_t). \tag{7}$$

The idea is to select matrices $\mathbf{B}_t$ close to the Hessian $\mathbf{H}(\mathbf{x}_t) := \nabla^2 F(\mathbf{x}_t)$ of the objective function. While various methods are known to select matrices $\mathbf{B}_t$ – see e.g., [7], [8] – those used in BFGS have been observed to work best in prior literature [7].

In BFGS the function's curvature is approximated by a finite difference. Specifically, define the variable and gradient variations at time t as

$$\mathbf{y}_t := \mathbf{x}_{t+1} - \mathbf{x}_t, \quad \mathbf{r}_t := \mathbf{s}(\mathbf{x}_{t+1}) - \mathbf{s}(\mathbf{x}_t), \tag{8}$$

respectively. We select the matrix $\mathbf{B}_{t+1}$ to be used in the next time step so that it satisfies the secant condition $\mathbf{B}_{t+1}\mathbf{y}_t = \mathbf{r}_t$. The rationale for this selection is that the Hessian $\mathbf{H}(\mathbf{x}_t)$ satisfies this condition for vanishing $\mathbf{y}_t$, i.e., for $\mathbf{x}_{t+1}$ tending to $\mathbf{x}_t$. Since there are many matrices that satisfy the secant condition $\mathbf{B}_{t+1}\mathbf{y}_t = \mathbf{r}_t$ we further notice that it is reasonable to expect the Hessians $\mathbf{H}(\mathbf{x}_{t+1})$ and $\mathbf{H}(\mathbf{x}_t)$ to be close to each other and therefore select $\mathbf{B}_{t+1}$ as the closest matrix to the previous Hessian approximation $\mathbf{B}_t$ among all those that satisfy the secant condition $\mathbf{B}_{t+1}\mathbf{y}_t = \mathbf{r}_t$. Closeness between $\mathbf{B}_t$ and $\mathbf{B}_{t+1}$ is specified in terms of the differential entropy between random variables with zero-mean Gaussian distributions $\mathcal{N}(0,\mathbf{B}_t)$ and $\mathcal{N}(0,\mathbf{Z})$ having covariance matrices $\mathbf{B}_t$ and $\mathbf{Z}$. Hence, the matrix $\mathbf{B}_{t+1}$ is defined as the solution of the semidefinite program

$$\begin{aligned} \mathbf{B}_{t+1} \;=\; & \operatorname{argmin} \;\; \operatorname{tr}(\mathbf{B}_t^{-1}\mathbf{Z}) - \log\det(\mathbf{B}_t^{-1}\mathbf{Z}) - n, \\ & \text{s.t.} \qquad \mathbf{Z}\mathbf{y}_t = \mathbf{r}_t, \quad \mathbf{Z} \succeq \mathbf{0}. \end{aligned} \tag{9}$$

It is not difficult to see that for convex functions the solution of (9) is positive definite when $\mathbf{B}_t$ is. It then follows that $\mathbf{B}_t \succ \mathbf{0}$ is positive definite for all iterations t as long as the initial matrix $\mathbf{B}_1 \succ \mathbf{0}$ is positive definite [8]. However, it is possible for the smallest eigenvalue of $\mathbf{B}_t$ to become arbitrarily close to zero which means that the largest eigenvalue of $\mathbf{B}_t^{-1}$ becomes very large. This has been proven not to be an issue in BFGS implementations but is a more significant challenge in the stochastic version proposed here motivating the regularization that we introduce in the following section.

A. Regularized BFGS

The most important property of Hessian approximations in quasi-Newton methods is satisfaction of the secant condition. Therefore, we introduce a regularization of (9) that keeps the constraint $\mathbf{Z}\mathbf{y}_t = \mathbf{r}_t$ but ensures the smallest eigenvalue of the solution $\mathbf{B}_{t+1}$ is larger than a positive constant δ,

$$\begin{aligned} \mathbf{B}_{t+1} = \operatorname{argmin} \; & \operatorname{tr}[\mathbf{B}_t^{-1}(\mathbf{Z}-\delta\mathbf{I})] - \log\det[\mathbf{B}_t^{-1}(\mathbf{Z}-\delta\mathbf{I})] - n, \\ & \text{s.t.} \qquad \mathbf{Z}\mathbf{y}_t = \mathbf{r}_t, \quad \mathbf{Z} \succeq \mathbf{0}. \end{aligned} \tag{10}$$

Since the negative logarithm determinant $-\log\det[\mathbf{B}_t^{-1}(\mathbf{Z}-\delta\mathbf{I})]$ diverges as the smallest eigenvalue of $\mathbf{Z}$ approaches δ, the smallest eigenvalue of the Hessian approximation matrices $\mathbf{B}_{t+1}$ computed as solutions of (10) exceed the lower bound δ. Subsequently, the largest eigenvalue of $\mathbf{B}_{t+1}^{-1}$ is bounded above by $1/\delta$ thereby limiting the effect of the noise inherent to the stochastic gradient. In the following lemma we show that the regularized approximations in (10) can be computed by an explicit formula[1].

Lemma 1 *Consider the semidefinite program in* (10) *where the matrix* $\mathbf{B}_t^{-1} \succ \mathbf{0}$ *is positive definite and the inner product* $(\mathbf{r}_t - \delta\mathbf{y}_t)^T\mathbf{y}_t > 0$. *Then, the solution* $\mathbf{B}_{t+1}$ *of* (10) *satisfies*

$$\mathbf{B}_{t+1} = \mathbf{B}_t + \frac{\tilde{\mathbf{r}}_t\tilde{\mathbf{r}}_t^T}{\mathbf{y}_t^T\tilde{\mathbf{r}}_t} - \frac{\mathbf{B}_t\mathbf{y}_t\mathbf{y}_t^T\mathbf{B}_t}{\mathbf{y}_t^T\mathbf{B}_t\mathbf{y}_t} + \delta\mathbf{I}, \tag{11}$$

where $\tilde{\mathbf{r}}_t := \mathbf{r}_t - \delta\mathbf{y}_t$ *is the corrected gradient variation.*

When $\delta = 0$ the update in (11) coincides with standard nonregularized BFGS [7], [8]. Therefore, the differences between BFGS and regularized BFGS are the replacement of the gradient variation $\mathbf{r}_t$ in (8) by the corrected variation $\tilde{\mathbf{r}}_t := \mathbf{r}_t - \delta\mathbf{y}_t$ and the addition of the regularization term $\delta\mathbf{I}$. We use (11) in the construction of the stochastic BFGS algorithm in the following section.

B. Stochastic BFGS

As can be seen from (11) the regularized BFGS curvature estimate $\mathbf{B}_{t+1}$ is obtained as a function of previous estimates $\mathbf{B}_t$, the iterates $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$, and corresponding gradients $\mathbf{s}(\mathbf{x}_t)$ and $\mathbf{s}(\mathbf{x}_{t+1})$. We can then think of a method in which gradients $\mathbf{s}(\mathbf{x}_t)$ are replaced by stochastic gradients $\hat{\mathbf{s}}(\mathbf{x}_t,\tilde{\boldsymbol{\theta}}_t)$ in both the curvature approximation update in (11) and the descent iteration in (7). These substitutions lead to the stochastic BFGS algorithm that we introduce in the following.

Start at time t with current iterate $\mathbf{x}_t$ and let $\hat{\mathbf{B}}_t$ stand for the Hessian approximation computed by stochastic BFGS in the previous iteration. Obtain a batch of channel samples $\tilde{\boldsymbol{\theta}}_t = [\boldsymbol{\theta}_{t,1}; ...; \boldsymbol{\theta}_{t,L}]$ and for each of the $\boldsymbol{\theta}_{t,l}$ samples determine the values of the stochastic gradient $\hat{\mathbf{s}}(\mathbf{x}_t,\tilde{\boldsymbol{\theta}}_t)$ as per (4). Add $\Gamma\mathbf{I}$ to the Hessian inverse approximation $\hat{\mathbf{B}}_t^{-1}$ to guarantee positive definiteness of pre-multiplier of stochastic gradient. Descend then along the direction $(\hat{\mathbf{B}}_t^{-1} + \Gamma\mathbf{I})\hat{\mathbf{s}}(\mathbf{x}_t,\tilde{\boldsymbol{\theta}}_t)$ moderated by the stepsize ϵ_t

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \epsilon_t(\hat{\mathbf{B}}_t^{-1} + \Gamma\mathbf{I})\hat{\mathbf{s}}(\mathbf{x}_t,\tilde{\boldsymbol{\theta}}_t). \tag{12}$$

For the iteration of $\mathbf{x}_{t+1}$ compute the stochastic gradient $\hat{\mathbf{s}}(\mathbf{x}_{t+1},\tilde{\boldsymbol{\theta}}_t)$ associated with the *same* set of radnom variable samples $\tilde{\boldsymbol{\theta}}_t$ used to compute the stochastic gradient $\hat{\mathbf{s}}(\mathbf{x}_t,\tilde{\boldsymbol{\theta}}_t)$. Define then stochastic gradient variation at time t as

$$\hat{\mathbf{r}}_t := \hat{\mathbf{s}}(\mathbf{x}_{t+1},\tilde{\boldsymbol{\theta}}_t) - \hat{\mathbf{s}}(\mathbf{x}_t,\tilde{\boldsymbol{\theta}}_t), \tag{13}$$

as well as the variable variation

$$\mathbf{y}_t := \mathbf{x}_{t+1} - \mathbf{x}_t. \tag{14}$$

Now based on the idea of regularized BFGS, we define the modified stochastic gradient variation as

$$\tilde{\mathbf{r}}_t := \hat{\mathbf{r}}_t - \delta\mathbf{y}_t. \tag{15}$$

The Hessian approximation $\hat{\mathbf{B}}_{t+1}$ for the next iteration is defined as the matrix that satisfies the stochastic secant condition $\hat{\mathbf{B}}_{t+1}\mathbf{y}_t = \hat{\mathbf{r}}_t$ and is closest to $\hat{\mathbf{B}}_t$ in the sense of (10). As per Lemma 1, when $(\hat{\mathbf{r}}_t - \delta\mathbf{y}_t)^T\mathbf{y}_t = \tilde{\mathbf{r}}^T\mathbf{y}_t > 0$ we can compute $\hat{\mathbf{B}}_{t+1}$ explicitly as the matrix

$$\hat{\mathbf{B}}_{t+1} = \hat{\mathbf{B}}_t + \frac{\tilde{\mathbf{r}}_t\tilde{\mathbf{r}}_t^T}{\mathbf{y}_t^T\tilde{\mathbf{r}}_t} - \frac{\hat{\mathbf{B}}_t\mathbf{y}_t\mathbf{y}_t^T\hat{\mathbf{B}}_t}{\mathbf{y}_t^T\hat{\mathbf{B}}_t\mathbf{y}_t} + \delta\mathbf{I}. \tag{16}$$

Conditions to guarantee that $\tilde{\mathbf{r}}_t^T\mathbf{y}_t > 0$ are introduced in Section III.

[1]Proofs are available in [10]

Algorithm 1 Stochastic BFGS

Require: Variable $\mathbf{x}_1$. Hessian approximation $\hat{\mathbf{B}}_1 \succ \delta \mathbf{I}$.

1: **for** $t = 1, 2, \ldots$ **do**

2: Acquire L independent channel samples $\tilde{\boldsymbol{\theta}}_t = [\boldsymbol{\theta}_{t,1}, \ldots, \boldsymbol{\theta}_{t,L}]$

3: Compute $\hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t) = \dfrac{1}{L}\displaystyle\sum_{l=1}^{L} \nabla_{\mathbf{x}} f(\mathbf{x}_t, \boldsymbol{\theta}_{t,l})$ [cf. (4)].

4: Descend along direction $(\hat{\mathbf{B}}_t^{-1} + \Gamma \mathbf{I})\ \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ [cf. (12)]

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \epsilon_t (\hat{\mathbf{B}}_t^{-1} + \Gamma \mathbf{I})\ \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t).$$

5: Compute $\hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_t) = \dfrac{1}{L}\displaystyle\sum_{l=1}^{L} \nabla_{\mathbf{x}} f(\mathbf{x}_{t+1}, \boldsymbol{\theta}_{t,l})$ [cf. (4)].

6: Compute variable variation $\mathbf{y}_t = \mathbf{x}_{t+1} - \mathbf{x}_t$ [cf. (14)].

7: Compute modified stochastic gradient variation [cf. (15)]

$$\tilde{\mathbf{r}}_t = \hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_t) - \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t) - \delta \mathbf{y}_t$$

8: Update approximation of Hessian [cf. (16)]

$$\hat{\mathbf{B}}_{t+1} = \hat{\mathbf{B}}_t + \frac{\tilde{\mathbf{r}}_t \tilde{\mathbf{r}}_t^T}{\mathbf{y}_t^T \tilde{\mathbf{r}}_t} - \frac{\hat{\mathbf{B}}_t \mathbf{y}_t \mathbf{y}_t^T \hat{\mathbf{B}}_t}{\mathbf{y}_t^T \hat{\mathbf{B}}_t \mathbf{y}_t} + \delta \mathbf{I}.$$

9: **end for**

The stochastic BFGS algorithm is summarized in Algorithm 1. Step 2 comprises the observation of L channel samples that are required to compute the stochastic gradients in steps 3 and 5. The stochastic gradient $\hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ in Step 3 is used in the descent iteration in Step 4. The stochastic gradient of Step 3 and the stochastic gradient $\hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_t)$ of Step 5 are used to compute the variations in steps 6 and 7 that permit carrying out the update of the Hessian approximation $\hat{\mathbf{B}}_t$ in Step 8. Iterations are initialized at arbitrary variable $\mathbf{x}_1$ and positive definite matrix $\hat{\mathbf{B}}_1$ with the smallest eigenvalue larger than δ.

Remark 1 One may think that the natural substitution of the gradient variation $\mathbf{r}_t = \mathbf{s}(\mathbf{x}_{t+1}) - \mathbf{s}(\mathbf{x}_t)$ is the stochastic gradient variation $\hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_{t+1}) - \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ instead of the one that we actually use $\hat{\mathbf{r}}_t = \hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_t) - \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$. This would have the advantage that $\hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_{t+1})$ is the stochastic gradient used to descend in iteration $t+1$ whereas $\hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_t)$ is not and is just computed for the purposes of updating $\mathbf{B}_t$. Therefore, using the variation $\hat{\mathbf{r}}_t = \hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_t) - \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ requires twice as many stochastic gradient computations as using the variation $\hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_{t+1}) - \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$. However, the use of the variation $\hat{\mathbf{r}}_t = \hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_t) - \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ is necessary to ensure that $(\hat{\mathbf{r}}_t - \delta \mathbf{y}_t)^T \mathbf{y}_t = \tilde{\mathbf{r}}_t^T \mathbf{y}_t > 0$. This is necessary for (16) to be true and cannot be guaranteed if we use the variation $\hat{\mathbf{s}}(\mathbf{x}_{t+1}, \tilde{\boldsymbol{\theta}}_{t+1}) - \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ – see Lemma (2) for details. The same observation holds true for the nonregularized version of stochastic BFGS introduced in [9].

III. Convergence

For the subsequent analysis it is convenient to define the instantaneous objective function associated with samples $\tilde{\boldsymbol{\theta}} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_L]$

$$\hat{f}(\mathbf{x}, \tilde{\boldsymbol{\theta}}) := \frac{1}{L}\sum_{l=1}^{L} f(\mathbf{x}, \boldsymbol{\theta}_l). \tag{17}$$

The definition of the instantaneous objective function $\hat{f}(\mathbf{x}, \tilde{\boldsymbol{\theta}})$ in association with the fact that $F(\mathbf{x}) := \mathbb{E}_{\boldsymbol{\theta}}[f(\mathbf{x}, \boldsymbol{\theta})]$ implies

$$F(\mathbf{x}) = \mathbb{E}_{\boldsymbol{\theta}}[\hat{f}(\mathbf{x}, \tilde{\boldsymbol{\theta}})]. \tag{18}$$

Our goal is to show as time progresses the sequence of $\mathbf{x}_t$ approaches the optimal value $\mathbf{x}^*$. To prove this result we make following assumptions.

Assumption 1 The instantaneous functions $\hat{f}(\mathbf{x}, \tilde{\boldsymbol{\theta}})$ are twice differentiable and the eigenvalues of the instantaneous Hessian $\hat{\mathbf{H}}(\mathbf{x}, \tilde{\boldsymbol{\theta}}) = \nabla_{\mathbf{x}}^2 \hat{f}(\mathbf{x}, \tilde{\boldsymbol{\theta}})$ are bounded between constants $\tilde{m} > 0$ and $\tilde{M} < \infty$ for all random variables $\tilde{\boldsymbol{\theta}}$,

$$\tilde{m}\mathbf{I} \preceq \hat{\mathbf{H}}(\mathbf{x}, \tilde{\boldsymbol{\theta}}) \preceq \tilde{M}\mathbf{I}. \tag{19}$$

The lower bound comes from the fact that we have assumed the random functions $\hat{f}(\mathbf{x}, \tilde{\boldsymbol{\theta}})$ are strongly convex. Having the upper bound for the eigenvalues of the instantaneous Hessian $\hat{\mathbf{H}}(\mathbf{x}, \tilde{\boldsymbol{\theta}})$ is equivalent to saying that each gradient $\hat{\mathbf{s}}(\mathbf{x}, \tilde{\boldsymbol{\theta}})$ is Lipschitz-countinuous with constant $\tilde{M}$.

Assumption 2 The second moment of the norm of the stochastic gradient is bounded for all $\mathbf{x}$. i.e., there exists a constant S^2 such that for all variables $\mathbf{x}$ it holds

$$\mathbb{E}\big[\|\hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)\|^2\big] \leq S^2, \tag{20}$$

As a consequence of Assumption 1 similar eigenvalue bounds hold for the function $F(\mathbf{x})$. It follows from the linearity of the expectation operator and the expression in (18) that the Hessian is $\nabla_{\mathbf{x}}^2 F(\mathbf{x}) = \mathbf{H}(\mathbf{x}) = \mathbb{E}[\hat{\mathbf{H}}(\mathbf{x}, \tilde{\boldsymbol{\theta}})]$. Combining this observation with the bounds in (19) implies that there are constants $m \geq \tilde{m}$ and $M \leq \tilde{M}$ such that

$$\tilde{m}\mathbf{I} \preceq m\mathbf{I} \preceq \mathbf{H}(\mathbf{x}) \preceq M\mathbf{I} \preceq \tilde{M}\mathbf{I}. \tag{21}$$

The bounds in (21) are customary in convergence proofs of descent methods. For the results here the stronger condition spelled in Assumption 1 is needed. The restriction imposed by Assumption 2 is typical of stochastic descent algorithms, its intent being to limit the random variation of stochastic gradients.

According to Lemma 1 the update in (16) is a solution to (10) – with the substitutions $\hat{\mathbf{B}}_t$ for $\mathbf{B}_t$ and $\mathbf{Z}\mathbf{y}_t = \hat{\mathbf{r}}_t$ for the secant condition $\mathbf{Z}\mathbf{y}_t = \mathbf{r}_t$ – as long as the inner product $(\hat{\mathbf{r}}_t - \delta \mathbf{y}_t)^T \mathbf{y}_t = \tilde{\mathbf{r}}_t^T \mathbf{y}_t > 0$ is positive. Our first result is to show that selecting $\delta < \tilde{m}$ guarantees that this inequality is satisfied for all times t.

Lemma 2 *Consider the modified stochastic gradient variation $\tilde{\mathbf{r}}_t$ defined in* (15) *and the variable variation $\mathbf{y}_t$ defined in* (14)*. Let Assumption 1 hold and recall the lower bound $\tilde{m}$ on the smallest eigenvalue of the instantaneous Hessians. Then, for all constants $\delta < \tilde{m}$ it holds*

$$\tilde{\mathbf{r}}_t^T \mathbf{y}_t \ = \ (\hat{\mathbf{r}}_t - \delta \mathbf{y}_t)^T \mathbf{y}_t \ > \ 0. \tag{22}$$

Initializing the curvature approximation matrix $\hat{\mathbf{B}}_1 \succ \delta \mathbf{I}$, which implies $\hat{\mathbf{B}}_1^{-1} \succ \mathbf{0}$, and setting $\delta < \tilde{m}$ it follows from Lemma (2) the hypotheses of Lemma (1) are satisfied for $t = 1$. Hence, the matrix $\hat{\mathbf{B}}_2$ computed from (16) is the solution of the semidefinite program in (10) and satisfies $\hat{\mathbf{B}}_2 \succ \delta \mathbf{I}$, which in turn implies $\hat{\mathbf{B}}_2^{-1} \succ \mathbf{0}$. Proceeding recursively we can conclude that $\hat{\mathbf{B}}_t^{-1} \succ \mathbf{0}$ and $\hat{\mathbf{B}}_t \succ \delta \mathbf{I}$; i.e., that the minimum eigenvalue of $\hat{\mathbf{B}}_t$ is at least δ for all times t. Equivalently we can conclude that $1/\delta \mathbf{I} \succ \hat{\mathbf{B}}_t^{-1}$ which implies the largest eigenvalue of $\hat{\mathbf{B}}_t^{-1}$ is at most $1/\delta$ for all times t. Observe that since the constant $\tilde{m}$ is not known in general we interpret the hypothesis $\delta < \tilde{m}$ as requiring δ to be suffieciently small. Adaptive selection of δ will be used in practice.

Having matrices $\hat{\mathbf{B}}_t^{-1}$ that are strictly positive definite and the eigenvalue are bounded by constant $1/\delta$ leads to the conclusion that if $\hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ is a descent direction, the same holds true of $\hat{\mathbf{B}}_t^{-1} \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$. The stochastic gradient $\hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ is not a descent direction in general, but we know that its conditional expectation $\mathbb{E}[\hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t) \,\big|\, \mathbf{x}_t] = \nabla_{\mathbf{x}} F(\mathbf{x}_t)$ is a descent direction. Therefore, we conclude that $\hat{\mathbf{B}}_t^{-1} \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t)$ is an average descent direction because $\mathbb{E}[\hat{\mathbf{B}}_t^{-1} \hat{\mathbf{s}}(\mathbf{x}_t, \tilde{\boldsymbol{\theta}}_t) \,\big|\, \mathbf{x}_t] = \hat{\mathbf{B}}_t^{-1} \nabla_{\mathbf{x}} F(\mathbf{x}_t)$. Stochastic optimization algorithms whose displacements $\mathbf{x}_{t+1} - \mathbf{x}_t$ are descent directions on average are expected to approach optimal arguments. This is indeed true as we claim in the following theorem.

Theorem 1 *Consider the stochastic BFGS algorithm as defined by* (12)-(16)*. If assumptions 1 and 2 hold true and the sequence of step sizes satisfies conditions* (6)*, then the limit infimum of the squared Euclidean distance to optimality $\|\mathbf{x}_t - \mathbf{x}^*\|^2$ satisfies*

$$\liminf_{t\to\infty} \|\mathbf{x}_t - \mathbf{x}^*\|^2 = 0 \tag{23}$$

with probability 1 over realizations of the random samples $\{\tilde{\boldsymbol{\theta}}_t\}_{t=1}^{\infty}$.

Theorem 1 shows the infimum of the squared distance between iterates $\mathbf{x}_t$ and the optimal vector $\mathbf{x}^*$ converges to zero almost surely. The

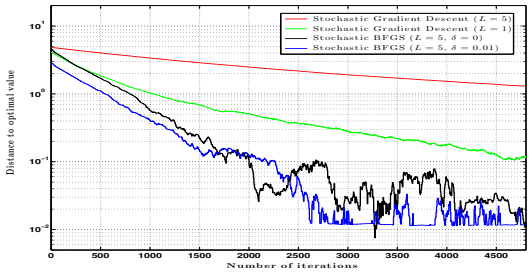

Fig. 1. Convergence of stochastic gradient descent, non-regularized stochastic BFGS, and regularized stochastic BFGS for the function in (24). For both versions of stochastic BFGS the number of iterations required to achieve a certain accuracy is smaller than the corresponding number for stochastic gradient descent. Further note the role of the regularization in providing more stability to stochastic BFGS. Condition number $10^{\xi} = 10^2$, step-size parameters $\epsilon_0 = 10^{-2}$ and $\tau = 10^3$.

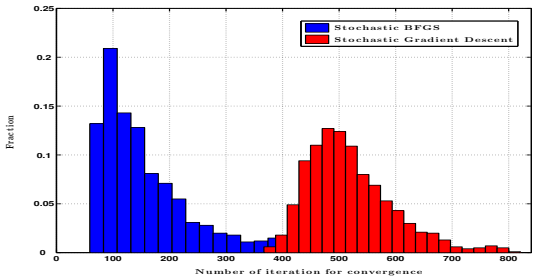

Fig. 2. Convergence of stochastic gradient descent and regularized stochastic BFGS for a well conditioned problem. Convergence for stochastic BFGS is better than stochastic gradient descent, but the number of iterations required for convergence is of the same order of magnitude. Condition number $10^{\xi} = 10^0$, regularization parameter $\delta = 10^{-2}$, batch size $L = 5$, step-size parameters $\epsilon_0 = 10^{-1}$ and $\tau = 10^3$, number of samples $= 10^3$, and accuracy $\|\mathbf{x}_t - \mathbf{x}^*\| \leq 10^{-1}$.

limit infimum convergence means there is a *subsequence* of iterates $\mathbf{x}_{t_j}$ converges to the optimal vector $\mathbf{x}^*$, rather than the whole sequence.

IV. Simulations

The goal of this section is to compare the convergence times of stochastic BFGS and stochastic gradient descent in problems with small and large condition numbers as well as to illustrate the advantage of regularizing stochastic BFGS. To explore these facts, we consider an optimization problem with a stochastic quadratic objective function. Let $\Theta = [-\theta_0, \theta_0]^n$ for some $\theta_0 < 1$ and $\boldsymbol{\theta}$ be uniformly drawn from Θ. Further consider given positive definite diagonal matrix $\mathbf{A} \in \mathbb{S}_n^{++}$ and vector $\mathbf{b} \in \mathbb{R}^n$ and define the quadratic stochastic function

$$\mathbb{E}_\theta\left[f(\mathbf{x},\theta)\right] := \mathbb{E}_\theta\left[\frac{1}{2}\mathbf{x}^T(\mathbf{A} + \mathbf{A}\text{diag}(\boldsymbol{\theta}))\mathbf{x} + \mathbf{b}^T\mathbf{x}\right]. \qquad (24)$$

The elements of $\mathbf{b}$ are fixed but chosen uniformly at random from $[0, 1]$ in different experiments. The elements $\mathbf{A}_{ii}$ of $\mathbf{A}$ are likewise fixed but chosen unfitly at random from the set $\{1, 10^{-1}, \ldots, 10^{-\xi}\}$ so that the condition number is 10^{ξ}. In the well conditioned problem we select $\xi = 0$ and in the ill conditioned problem $\xi = 2$. For the objective in (24) the average function can be computed in closed form as $(1/2)\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x}$ which permits determination of $\mathbf{x}^*$ for comparison against $\mathbf{x}_t$. Algorithm 1 is implemented for the function in (24) with $\theta_0 = 0.5$ and $n = 10$.

A representative run of stochastic gradient descent, non-regularized stochastic BFGS – corresponding to $\delta = 0$ in Algorithm 1 – and regularized stochastic BFGS with $\delta = 10^{-2}$ when $\xi = 2$ is shown in Fig. 1. Convergence for both versions of stochastic BFGS is faster than stochastic gradient descent. It takes gradient descent 4.8×10^2 iterations to reach a distance to optimality of 10^{-1} but 2.1×10^2 iterations for stochastic BFGS. This difference can be made arbitrarily large by modifying the condition number of $\mathbf{A}$. Further note the role of the regularization in providing more stability to stochastic BFGS.

A more comprehensive analysis of the relative advantages of BFGS appears in Figs. 2 and 3. Fig. 2 shows the histogram of the number of iterations needed to achieve distance $\|\mathbf{x}_t - \mathbf{x}^*\| \leq 10^{-1}$ for stochastic BFGS and stochastic gradient descent when $\xi = 0$. The step-size parameters are $\tau = 10^3$ and $\epsilon_0 = 10^{-1}$. As we can see in Fig. 2, the speed of convergence for stochastic BFGS is better than stochastic gradient descent, but the number of iterations required for convergence is of the same order of magnitude. Fig. 3 shows the number of iterations needed to achieve distance $\|\mathbf{x}_t - \mathbf{x}^*\| \leq 10^{-1}$ for stochastic BFGS and stochastic gradient descent when $\xi = 2$. The step-size parameters are $\tau = 10^3$ and $\epsilon_0 = 10^{-2}$ for both algorithms in this case. Fig. 3 shows that stochastic BFGS reduces the convergence time of stochastic gradient descent by an order of magnitude. It takes gradient descent an average

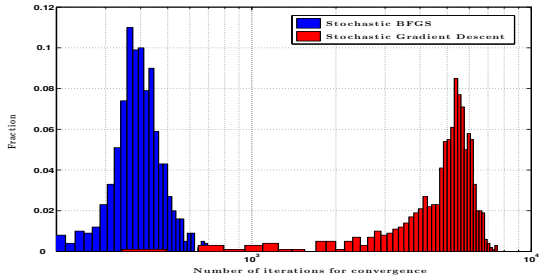

Fig. 3. Convergence of stochastic gradient descent and regularized stochastic BFGS for an ill conditioned problem. Stochastic BFGS reduces the convergence time of stochastic gradient descent by an order of magnitude. Condition number $10^{\xi} = 10^2$, regularization parameter $\delta = 10^{-2}$, batch size $L = 5$, step-size parameters $\epsilon_0 = 10^{-2}$ and $\tau = 10^3$, number of samples $= 10^3$, and accuracy $\|\mathbf{x}_t - \mathbf{x}^*\| \leq 10^{-1}$.

of 6×10^3 iterations to reach a distance to optimality of 10^{-1} whereas stochastic BFGS achieves the same in an average of 4×10^2 iterations.

References

[1] V. Vapnik, *The nature of statistical learning theory*, 2nd ed. springer, 1999.

[2] L. Bottou, "Large-scale machine learning with stochastic gradient descent," *In Proceedings of COMPSTAT'2010*, pp. 177–186, Physica-Verlag HD, 2010.

[3] A. Mokhtari and A. Ribeiro, "A dual stochastic dfp algorithm for optimal resource allocation in wireless systems," *In Proc. IEEE Workshop on Signal Process. Advances in Wireless Commun.*, (to appear), Darmstadt, Germany, June 16-19 2013.

[4] A. Ribeiro, "Optimal resource allocation in wireless communication and networking," *EURASIP J. Wireless commun.*, vol. 2012, no. 272, pp. 3727–3741, August 2012.

[5] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal estimated sub-gradient solver for svm," *In Proceedings of the 24th international conference on Machine learning*, pp. 807–814, ACM, 2007.

[6] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms," *In Proceedings of the twenty-first international conference on Machine learning*, p. 919926, ACM, 2004.

[7] R. H. Byrd, J. Nocedal, and Y. Yuan, "Global convergence of a class of quasi-newton methods on convex problems," *SIAM J. Numer. Anal.*, vol. 24, no. 5, pp. 1171–1190, October 1987.

[8] J. Nocedal and S. J. Wright, *Numerical optimization*, 2nd ed. New York, NY: Springer-Verlag, 1999.

[9] N. N. Schraudolph, J. Yu, and S. Gnter, "A stochastic quasi-newton method for online convex optimization," *In Proc. 11th Intl. Conf. on Artificial Intelligence and Statistics (AIstats)*, p. 433 440, Soc. for Artificial Intelligence and Statistics, 2007.

[10] A. Mokhtari and A. Ribeiro, "Regularized stochastic bfgs algorithm," *https://fling.seas.upenn.edu/~aryanm/wiki/index.php?n=Research.Publications*.