A Variance Reduced Stochastic Newton Method

Aurelien Lucchi
Department of Computer Science, ETH Zürich Universitätstrasse 6, 8092 Zürich Switzerland
AURELIEN.LUCCHI@INF.ETHZ.CH

Brian McWilliams
Department of Computer Science, ETH Zürich Universitätstrasse 6, 8092 Zürich Switzerland
BRIAN.MCWILLIAMS@INF.ETHZ.CH

Thomas Hofmann
Department of Computer Science, ETH Zürich Universitätstrasse 6, 8092 Zürich Switzerland
THOMAS.HOFMANN@INF.ETHZ.CH

Abstract
We present a new method to reduce the variance of stochastic versions of the BFGS optimization method, applied to the optimization of a class of smooth strongly convex functions. Although Stochastic Gradient Descent (SGD) is a popular method to solve this kind of problem, its convergence rate is sublinear as it is in fact limited by the noisy approximation of the true gradient. In order to recover a high convergence rate, one has to pick an appropriate step-size or explicitly reduce the variance of the approximate gradients. Another limiting factor of SGD is that it ignores the curvature of the objective function that can help greatly speed up convergence. Stochastic variants of BFGS that include curvature have shown good empirical performance but suffer from the same noise effects as SGD. We here propose a new algorithm VITE that uses an existing technique to reduce this variance while allowing a constant step-size to be used. We show that the expected objective value converges to the optimum at a geometric rate. We experimentally demonstrate improved convergence rate on diverse stochastic optimization problems.

1. Introduction
We consider the problem of optimizing a function expressed as an expectation over a set of data-dependent functions. Standard gradient descent techniques require the computation of a gradient over all datapoints and are thus inefficient for large datasets. Stochastic gradient descent (SGD) has become a popular alternative (Bottou, 2010; Shalev-Shwartz et al., 2011) as it only requires computing stochastic gradients over a much smaller subset of datapoints. Although SGD is attractive due to its simplicity, a severe limiting factor is that its convergence rate is sublinear as it is in fact limited by the noisy approximation of the true gradient. In an effort to deal with this randomness, two primary directions of focus have been developed. The first line of work focuses on choosing the appropriate SGD step-size (Bach et al., 2011; Lacoste-Julien et al., 2012; Rakhlin et al., 2011). As the step-size decreases, the variance is forced to zero asymptotically leading to convergence. However, small step-sizes slow down progress and severely limit the rate of convergence. Therefore, a careful choice of the step-size is needed, which can require extensive experimentation possibly negating the computational speedup of SGD. The approach for reducing the randomness of SGD is to use an improved estimate of the gradient with lower variance. As the variance of the new estimate goes to zero, asymptotically convergence is reached without having to tune the step-size allowing for a constant value to be used instead. This scheme is used in (Defazio et al., 2014; Roux et al., 2012) where the improved estimate of the gradient combines stochastic gradients computed at the current stage with others used at an earlier stage. A similar approach proposed in (Johnson & Zhang, 2013; Konečný & Richtárik, 2013) combines stochastic gradients with gradients periodically re-computed at a pivot point.

These methods show a clear speed-up over SGD, but are still theoretically limited by a linear convergence rate. In order to reach superlinear convergence, second order methods that require the computation and inversion of the Hessian matrix must be used, incurring high complexity for large-scale datasets. Approximate variants known as quasi-Newton methods (Dennis & Moré, 1977) have thus been developed, such as the popular BFGS or its limited memory...
A variance reduced stochastic Newton method

version known as LBFGS (Liu & Nocedal, 1989). Quasi-
Newton methods do not require computing the Hessian ma-
trix but instead constructs a quadratic model of the ob-
jective function by successive measurements of the gra-
dient. This yields a superlinear convergence when the
quadratic model is accurate. Stochastic variants of BFGS
have been proposed, for which stochastic gradients replace
their deterministic counterparts. This is for example the
case of the oBFGS method in (Schraudolph et al., 2007) or
a regularized version known as RES (Mokhtari & Ribeiro,
2014). By enforcing a bound on the eigenvalues of the
Hessian matrix, RES achieves a linear convergence rate.
Although quasi-Newton methods or their stochastic coun-
terparts have not be shown to achieve a superlinear conver-
sion, they empirically outperform SGD for problems with
a large condition number (Mokhtari & Ribeiro, 2014).

A clear drawback to second order stochastic methods is
that, similarly to their first-order counterparts, they suffer
from high variance in the approximation of the gradient.
One additional problem encountered when the stochastic
gradients are too noisy is that the estimate of the Hessian
can then magnify the effect of the noise. In this paper, we
propose and analyze a stochastic variant of BFGS that uses
a multi-stage scheme similar to (Johnson & Zhang, 2013;
Konečný & Richtárik, 2013) to progressively reduce the
variance of the stochastic gradients. We call this method
Variance-reduced Stochastic Newton (VITE) and show that
it reaches the optimum at a geometric rate with a constant
step-size when dealing with smooth strongly convex func-
tions. We also experimentally demonstrate the improved
convergence rate on diverse stochastic optimization prob-
lems.

We first briefly review the BFGS and LBFGS algorithms,
and discuss the changes required to make them work on-
line. We then introduce VITE, a new variance reduction
method for the online version of BFGS, and analyze
its convergence properties as well as present experimental
results on real-world datasets.

2. Preliminaries

We consider the optimization of a function that depends
on a set of parameters \( w \in \mathbb{R}^d \) and a set of data \( D = \{x_1, \ldots, x_n, y_1, \ldots, y_n\} \) where \( x_i \in \mathbb{R}^d \) is the feature vector
of datapoint \( i \) and \( y_i \in [0, C] \) is the target output.
The question we consider here is to find the set of parameters \( w \) that minimizes the expected loss \( f(w) = \mathbb{E}[f_i(w)] \) where
the function \( f_i(w) \) takes the form
\[
f_i(w) = \ell(h(w, x_i), y_i),
\]
where \( \ell \) is a loss function and \( h \) is a prediction model
parametrized by \( w \).
The minimum of this function is denoted as \( w^* = \arg \min_w f(w) \). This optimization problem can be solved
exactly for convex functions using gradient descent, where
the gradient of the loss function is expressed as
\[
\nabla_w f_i(w) = \mathbb{E}[\nabla_w f_i(w)].
\]

When the size of the dataset \( n \) is large, the computation of
the gradient is impractical and one has to resort to stochas-
tic gradients. Similar to gradient descent, stochastic gradi-
ent descent updates the parameter vector \( w_t \) by stepping in
the opposite direction of the stochastic gradient \( \nabla_w f_i(w_t) \)
by an amount specified by a step size \( \eta_t \) as follows:
\[
w_{t+1} = w_t - \eta_t \nabla_w f_i(w_t). \tag{1}
\]

In general, a stochastic gradient can also be computed
as an average over a sample of datapoints as
\[
\hat{f}(w_t) = \sum_{i=1}^{T} f_i(w_t).
\]
Given that the stochastic gradients are unbiased estimates
of the gradient, Robbins and Monro (Robbins & Monro,
1951) proved convergence of SGD to \( w^* \) assuming that
the step-size sequence satisfies the following conditions:
\[
\lim_{T \to +\infty} \sum_{t=1}^{T} \eta_t = \infty \quad \text{and} \quad \lim_{T \to +\infty} \sum_{t=1}^{T} (\eta_t)^2 < \infty. \tag{2}
\]
A common choice for the step size is (Shalev-Shwartz et al.,
2011; Mokhtari & Ribeiro, 2014)
\[
a) \ \eta_t = \frac{\eta_0}{t} \quad \text{or} \quad b) \ \eta_t = \frac{\eta_0 T_0}{T_0 + t} \tag{3}
\]
where \( \eta_0 \) is a constant initial step size and \( T_0 \) controls the
speed of decrease.

Although the cost per iteration of SGD is low, it suffers
from slow convergence for certain ill-conditioned prob-
lems (Mokhtari & Ribeiro, 2014). An alternative is to use
a second order method such as Newton’s method that es-
timates the curvature of the objective function and can
achieve extremely fast quadratic convergence. In the follow-
ing, we review Newton’s method and its approximations
known as quasi-Newton methods.

2.1. Newton’s method

Newton’s method is an iterative method that minimizes the
Taylor expansion of \( f(w) \) around \( w_t \):
\[
f(w) = f(w_t) + (w - w_t)^\top \nabla_w f(w_t) + \frac{1}{2} (w - w_t)^\top H(w - w_t), \tag{4}
\]
where \( H \) is the Hessian of the function \( f(w) \).
Minimizing Eq. 4 leads to the following update rule:
\[
w_{t+1} = w_t - \eta_t H_t^{-1} \cdot \nabla f(w_t), \tag{5}
\]
where $\eta_t$ is the step size chosen by backtracking line search.

Given that computing and inverting the Hessian matrix is an expensive operation, approximate variants of Newton’s method have emerged, where $H_t^{-1}$ is replaced by an approximate version $\tilde{H}_t^{-1}$ selected to be positive definite and as close to $H_t^{-1}$ as possible. The most popular member of this class of quasi-Newton methods is BFGS (Nocedal & Wright, 1999).

BFGS approximates Newton’s method by incrementally updating an estimate of the inverse Hessian, denoted $J_t = H_t^{-1}$. This estimate is computed by minimizing a weighted Frobenius norm $||J_{t+1} - J_t||_W$ subject to the secant condition defined as follows:

$$w_{t+1} - w_t = J_{t+1}(\nabla f(w_{t+1}) - \nabla f(w_t)). \quad (6)$$

These conditions can be resolved in closed form leading to the following explicit expression:

$$J_{t+1} = \left( I - \frac{s y^\top}{y^\top s} \right) J_t \left( I - \frac{y s^\top}{y^\top s} \right) + \frac{ss^\top}{y^\top s} \quad (7)$$

where $s = w_{t+1} - w_t$ and $y = \nabla f(w_{t+1}) - \nabla f(w_t)$.

Eq. 7 is known to be positive definitive assuming that $J_0$ is initialized to be a positive definite matrix.

2.2. Stochastic BFGS

A stochastic version of BFGS known as oBFGS was proposed in (Schraudolph et al., 2007) in which stochastic gradients are used for both the determination of the descent direction and the approximation of the inverse Hessian. The oBFGS approach described in Algorithm 1 uses the following update equation:

$$w_{t+1} = w_t - \eta_t \tilde{J}_t \cdot \nabla \tilde{f}(w_t), \quad (8)$$

where the matrix $\tilde{J}_t$ and the vector $\nabla \tilde{f}(w_t)$ are stochastic estimates computed as follows.

Let $A \subset \{1 \ldots n\}$ and $B \subset \{1 \ldots n\}$ be sets containing two independent samples of datapoints. The variables $y$ and $\nabla f(w)$ defined in Eq. 7 are replaced by sampled variables $\hat{y}$ and $\nabla \hat{f}(w)$ computed as

$$\hat{y} = \sum_{k \in A} \nabla f_k(w_{t+1}) - \nabla f_k(w_t), \quad (9)$$

and

$$\nabla \hat{f}(w_t) = \nabla f_B(w_t) = \sum_{k \in B} \nabla f_k(w_t). \quad (10)$$

The estimate of the inverse Hessian then becomes

$$\tilde{J}_{t+1} = \left( I - \frac{s y^\top}{y^\top s} \right) \tilde{J}_t \left( I - \frac{y s^\top}{y^\top s} \right) + \frac{ss^\top}{y^\top s} \quad (11)$$

Unlike Newton’s method, oBFGS uses a fixed step size sequence instead of a line search. A common choice is to use a step size similar to the one used for SGD in Eq. 3.

2.3. Regularized Stochastic BFGS

The regularized stochastic BFGS method developed in (Mokhtari & Ribeiro, 2014) was named RES and differs from stochastic BFGS in the use of a regularization to enforce a bound on the eigenvalues of $J_t$ such that

$$\gamma I \preceq \tilde{J}_t \preceq \rho I = \left( \gamma + \frac{1}{\delta} \right) I, \quad (12)$$

where $\gamma$ and $\delta$ are given positive constants and the notation $A \preceq B$ means that $B - A$ is a positive semi-definite matrix.

Note that (12) also implies an upper and lower bound on $E[\tilde{J}_t]$ (Mokhtari & Ribeiro, 2014).

The update of RES is modified to incorporate an identity bias term $\gamma I$ as follows:

$$w_{t+1} = w_t - \eta_t (\tilde{J}_t + \gamma I) \cdot \nabla \tilde{f}(w_t), \quad (13)$$

The convergence proof derived in (Mokhtari & Ribeiro, 2014) shows that lower and upper bounds on the Hessian eigenvalues of the sample functions are sufficient to guarantee convergence to the optimum.

3. Formulation

While reducing the size of the sets $A$ and $B$ is highly desirable for computational efficiency, it also increases the variance of the stochastic gradients. Here we propose a new method called VITE that explicitly reduces this variance. In order to simplify the analysis of VITE, we select the samples in the sets $A$ and $B$ independently. This makes the two sets $A$ and $B$ conditionally independent given the
past, which means that the expectation of the Newton descent direction $E_{\mathcal{A},\mathcal{B}}[(\hat{J}_t)_{\mathcal{A}} \cdot (\nabla f(w_t))_{\mathcal{B}}]$ simplifies as

$$E_{\mathcal{A},\mathcal{B}}[(\hat{J}_t)_{\mathcal{A}} \cdot (\nabla f(w_t))_{\mathcal{B}}] = E_{\mathcal{A}}[\hat{J}_t]E_{\mathcal{B}}[\nabla f(w_t)].$$ (14)

Similarly, the second moment of $(\hat{J}_t)_{\mathcal{A}} \cdot (\nabla f(w_t))_{\mathcal{B}}$ decomposes as

$$E_{\mathcal{A},\mathcal{B}} \left\| (\hat{J}_t)_{\mathcal{A}} \cdot (\nabla f(w_t))_{\mathcal{B}} \right\|^2 \leq E_{\mathcal{A}} \left\| \hat{J}_t \right\|^2 E_{\mathcal{B}} \left\| \nabla f(w_t) \right\|^2. \quad (15)$$

In order to reduce the variance of the estimate $(\hat{J}_t)_{\mathcal{A}} \cdot (\nabla f(w_t))_{\mathcal{B}}$, one can thus reduce the variance of $\hat{J}_t$ and the variance of $\nabla f(w_t)$ independently. Here we assume that the variance of $\hat{J}_t$ can be kept under control, for example using the regularization of the RES method. On the other hand, we propose to reduce the variance of $\nabla f(w_t)$ using a variance reduction technique similar to the one proposed in (Johnson & Zhang, 2013; Konečný & Richtárik, 2013).

Algorithm 2 shows that VITE differs from oBFGS in the use of a multi-stage scheme where a variable $\tilde{w}$ is introduced and whose gradient is periodically computed and inserted in the update equation to reduce the variance. For each inner loop the method produces a random number $t_j \in [1, m]$ of steps, following a geometric law, with parameter

$$\beta = \sum_{t=1}^{m} (1 - \nu \eta)^{m-t},$$

where $\nu \in [0, \mu]$. Stochastic gradients at $w_t$ and $\tilde{w}$ are computed in each iteration of the inner loop. The descent direction $\nabla f_{\mathcal{B}}(w)$ is then replaced by

$$v_t = \nabla f_{\mathcal{B}}(w_t) - \nabla f_{\mathcal{B}}(\tilde{w}) + \tilde{\mu}$$

such that $\tilde{\mu} = E[\nabla f_{\mathcal{B}}(\tilde{w})]$ and $E[v_t] = E[\nabla f_{\mathcal{B}}(w_t)]$. A constant step-size can be used to reach convergence as the variance of the new gradient $v_t$ goes to 0 when both $w$ and $w_t$ converge to the same parameter $w^*$. Further details appear in Section 4.

The update equation for VITE thus becomes

$$w_{t+1} = w_t - \eta \hat{J}_t \cdot v_t. \quad (16)$$

The complexity of this approach depends on the number of epochs $S$ and a constant $m$ limiting the number of stochastic gradients computed in a single epoch, as well as other parameters that will be introduced in more details in the next section.

### Algorithm 2 VITE

1: INPUTS:
2: \( D \): Training set of \( n \) examples.
3: \( w_0 \): Arbitrary initial values, e.g., 0.
4: \( \eta \): Constant step size
5: \( m \): Arbitrary constant
6: \( \nu \in [0, \mu] \)
7: OUTPUT: \( w_t \)
8: \( \hat{J}_0 \leftarrow \alpha I \)
9: for \( s = 0 \ldots S \)
10: \( \tilde{w} = \hat{w}_{s-1} \)
11: \( \tilde{\mu} = \frac{1}{\nu} \sum_{i=1}^{n} \nabla f_i(\tilde{w}) \)
12: \( w_{t+1} \leftarrow w_t - \eta \hat{J}_{t+1} \cdot v_t \)
13: end for
14: for \( t = 0 \ldots t_j - 1 \)
15: Randomly pick independent sets \( \mathcal{A}, \mathcal{B} \subset \{1 \ldots n\} \)
16: \( v_t = \nabla f_{\mathcal{B}}(w_t) - \nabla f_{\mathcal{B}}(\tilde{w}) + \tilde{\mu} \)
17: \( w_{t+1} \leftarrow w_t - \eta \hat{J}_{t+1} \cdot v_t \)
18: end for
19: \( w_s = w_t \),
20: end for

### 4. Analysis

We present a convergence proof for the VITE algorithm that builds upon and generalizes the analysis provided in (Johnson & Zhang, 2013; Konečný & Richtárik, 2013). Specifically, we show how variance reduction on the stochastic gradient direction is sufficient to establish geometric convergence rates, even when performing linear transformations with a stochastic matrix $J_t$ (e.g. an approximation of the inverse Hessian). Since we do not exploit the specific form of the stochastic evolution equations for $J_t$, this analysis will not allow us to argue in favor of the specific choice of Eq. 11, yet it shows that variance reduction on the gradient estimate is sufficient for fast convergence as long as $J_t$ is guaranteed to obey the condition in Eq. 12.

Our analysis relies on the following assumptions:

#### Assumptions

**A1** Each function $f_i$ is differentiable and has a Lipschitz continuous gradient with constant $L > 0$, i.e., $\forall w, v \in \mathbb{R}^n$, 

$$f_i(w) \leq f_i(v) + (w - v)^\top \nabla f_i(v) + \frac{L}{2} ||w - v||^2 \quad (17)$$

**A2** $f$ is $\mu$-strongly convex, i.e., $\forall w, v \in \mathbb{R}^n$, 

$$f(w) \geq f(v) + (w - v)^\top \nabla f(v) + \frac{\mu}{2} ||w - v||^2 \quad (18)$$
which also implies
\[ \|\nabla f(w)\|^2 \geq 2\mu(f(w) - f(w^*)) \quad \forall w \in \mathbb{R}^n \] (19)
for the minimizer \( w^* \) of \( f \).

We start by stating two lemmas required for the proof of convergence.

**Lemma 1.** For any vectors \( a, b \in \mathbb{R}^d \) and positive definite matrix \( G \in \mathbb{R}^{d \times d} \) with largest eigenvalue \( \lambda_{\text{max}} \) and smallest eigenvalue \( \lambda_{\text{min}} \) we have
\[ a^T G b \geq \frac{\lambda_{\text{min}} - \lambda_{\text{max}}}{2}(\|a\|^2 + \|b\|^2) + \lambda_{\text{max}} a^T b \]

**Proof.** See Appendix. \( \square \)

**Lemma 2.** The following identity holds:
\[ Ef(\tilde{w}_{s+1}) = \frac{1}{\beta} \sum_{t=1}^{m} \tau_t Ef(w_{t-1}) \]
where \( \tau_t := (1 - \nu\rho_t) \) and the weight vectors \( w_t \) belong to epoch \( s \).

**Proof.** See Lemma 3 in (Konečný & Richtárik, 2013). \( \square \)

**Theorem 1.** Let Assumptions A1 and A2 be satisfied. Choose \( 0 \leq \nu \leq \mu, 0 < \eta < \left( \frac{1}{2L\rho} - \frac{\delta^{-1}}{\rho^2} \right) \) and set \( m \) be sufficiently large so that \( \alpha = \frac{(1 + \eta\delta^{-1} - \nu\eta\rho_m^{m})}{\beta\nu\eta \rho^{2L\rho^2 - 2L\delta^{-1}}} + \frac{2L\rho^2 + \delta^{-1}}{\beta \nu \rho^{2L\rho^2 - 2L\delta^{-1}}} \leq 1. \)

Then the suboptimality of \( \tilde{w}_s \) is bounded in expectation as follows:
\[ Ef(\tilde{w}_s) - f(w^*) \leq \alpha^s Ef(f(w_0) - f(w^*)]. \]

**Proof.** Our starting point is the basic equality
\[ \|\Delta w_t\|^2 = \|\Delta w_{t-1}\|^2 + \eta^2 \left\langle \partial t, \tilde{J}_t \right\rangle \]
\[ -2\eta\langle \Delta w_{t-1}, \tilde{J}_t \rangle, \]
with \( \Delta w_t := w_t - w^* \). For the purpose of the analysis, we define \( \mathcal{F}_{t-1} \) to be the sigma-algebra measuring \( w_{t-1} \). By conditioning on \( \mathcal{F}_{t-1} \), the remaining randomness is in the choice of the index sets \( \mathcal{A} \) and \( \mathcal{B} \) in round \( t \), which are tied to the stochasticity of \( \tilde{J}_t \) and \( \tilde{v}_t \), respectively. Note that the outcome of \( \mathcal{A} \) affects only \( \tilde{J}_t \) and we can compute a conditional expectation of the relevant terms showing up on the RHS of Eq. 21. Setting \( \tilde{J}_t := E_{\mathcal{A}}[\tilde{J}_t] \) we get:
\[ Ef(\tilde{w}_{s+1}) = \frac{1}{\beta} \sum_{t=1}^{m} \tau_t Ef(w_{t-1}) \]
\[ E_{\mathcal{A}}\left[\|\Delta w_{t-1}, \tilde{J}_t \right\rangle \]
\[ \geq \frac{\gamma - \rho}{2} \left( ||\Delta w_{t-1}||^2 + ||\tilde{v}_t||^2 \right) + \rho \langle \Delta w_{t-1}, \tilde{v}_t \rangle \]
\[ (23) \]

The first inequality is based on Eqs. 12 and 15, whereas the second one is based on Lemma 1 and Eq. 12. Plugging these bounds back into Eq. 21 and taking the (conditional) expectation with regard to both, \( \mathcal{A} \) and \( \mathcal{B} \), results in
\[ Ef(\tilde{w}_{s+1}) - f(w^*) \leq Ef(f(w_0) - f(w^*) + \frac{\eta^2 \rho^2 Ef(||\tilde{v}_t||^2)}{2} \]
\[ + \eta(\rho - \gamma)(||\Delta w_{t-1}||^2 + Ef(||\tilde{v}_t||^2)) \]
\[ - 2\eta\rho \langle \Delta w_{t-1}, \tilde{v}_t \rangle \]
\[ \leq \left( 1 + \frac{\eta}{\delta} \right) ||\Delta w_{t-1}||^2 + \eta \left( \eta^2 + \frac{1}{\delta} \right) Ef(||\tilde{v}_t||^2) \]
\[ - 2\eta\rho \langle \Delta w_{t-1}, \tilde{v}_t \rangle. \]
\[ (24) \]

Note that if we had \( \tilde{J}_t = I \) (i.e. no quasi-Newton method), we get \( \tilde{J}_t = I \) and \( \rho = \lambda_{\text{max}}(\tilde{J}_t) = 1 \) and we would be back to the standard line of argument as presented in (Johnson & Zhang, 2013; Konečný & Richtárik, 2013). The main generalization necessary is thus to allow for a separate weighting of the linear and norm component on the RHS of Eq. 24. We can achieve this as stated in the following lemmata that bound the respective terms:

**Lemma 3.**
\[ Ef(||\tilde{v}_t||^2) \leq 4L(f(w_{t-1}) - f(w^*) + f(\tilde{w}) - f(w^*)) \]

**Proof.** (Johnson & Zhang, 2013; Konečný & Richtárik, 2013), see also Appendix for the full derivation. \( \square \)

**Lemma 4.**
\[ -\langle \Delta w_t, \tilde{v}_t \rangle \leq f(w^*) - f(w_t) - \frac{\nu}{2} ||\Delta w_t||^2 \]
\[ (25) \]

**Proof.** Follows directly from the \( \mu \)-strong convexity of \( f \) and from the fact that \( \nu \in [0, \mu]. \) \( \square \)

For a fixed epoch \( s \) taking expectation over \( \mathcal{F}_{t-1} \) thus yields
\[ Ef(\tilde{w}_{s+1}) - f(w^*) \leq \left( 1 + \eta \delta^{-1} - \nu\eta\rho \right) Ef(||\Delta w_{t-1}||^2) \]
\[ + 4L(\eta\rho^2 + \delta^{-1})Ef(f(w_s) - f(w^*)) \]
\[ + 2\eta(2L\eta\rho^2 + 2L\delta^{-1} - \rho)Ef(f(w_{t-1}) - f(w^*). \]
\[ (26) \]

Now we sum all these inequalities at iterations \( t = 1, \ldots, m \) performed in epoch \( s \) with weights \( \tau_t \). Note that we can apply Lemma 2 to the last summand to recover \( f(\tilde{w}_{s+1}) \) and arrive at
\[ \beta Ef[f(\tilde{w}_{s+1}) - f(w^*)] \]
\[ \leq \frac{2L(\eta\rho^2 + \delta^{-1})}{\rho - 2L\eta\rho^2 - 2L\delta^{-1}} \beta Ef[f(w_s) - f(w^*)] \]
\[ \sum_{t=1}^{m} \tau_t \left[ 1 + \eta \delta^{-1} - \nu\eta\rho \right] Ef(||\Delta w_{t-1}||^2) \]
\[ + \frac{2\eta(2L\eta\rho^2 + 2L\delta^{-1} - \rho)}{\rho - 2L\eta\rho^2 - 2L\delta^{-1}} . \]
\[ (27) \]
We now need to bound the remaining sum \((*)\) in the numerator, which can be accomplished by re-grouping summands
\[
(*) = (1 + \eta \delta^{-1} - \nu \eta \mu)^m \mathbb{E} \| \Delta \tilde{w}_s \|^2 (28) \\
- (1 + \eta \delta^{-1} - \nu \eta \mu) \mathbb{E} \| \Delta \tilde{w}_{s+1} \|^2.
\]
By ignoring the negative term and exploiting the strong convexity of \(f\), we get
\[
(*) \leq \frac{2(1 + \eta \delta^{-1} - \nu \eta \mu)^m}{\mu} \mathbb{E} [f(\tilde{w}_s) - f(w^*)].
\]
Finally, we can summarize all terms into a factor
\[
\alpha = \frac{(1 + \eta \delta^{-1} - \nu \eta \mu)^m}{\beta \mu \eta (\rho - 2 L \eta \delta^2 - 2 L \delta^{-1})} + \frac{2 L (\eta \rho^2 + \delta^{-1})}{\rho - 2 L \eta \rho^2 - 2 L \delta^{-1}}.
\]

Theorem 1 implies that \(\text{VITE}\) has a local geometric convergence rate with a constant learning rate. In order to satisfy \(\mathbb{E} [f(\tilde{w}_s) - f(w^*)] \leq \epsilon\), the number of stages \(s\) needs to satisfy
\[
s \geq - \log \alpha^{-1} \log \frac{\mathbb{E} (f(\tilde{w}_0) - f(w^*))}{\epsilon}.
\]
Since each stage requires \(n + m (|A| + 2|B|)\) component gradient evaluations, the overall complexity is
\[
O((n + 2m (|A| + |B|)) \log (1/\epsilon)).
\]
The experimental results presented in the next section show that \(\text{VITE}\) can drastically reduce the variance of stochastic BFGS algorithms thus speeding convergence.

5. Experimental Results

This section presents experimental results that demonstrate the performance of \(\text{VITE}\) compared to several baselines. We consider two commonly occurring problems in machine learning, namely least-square regression and regularized logistic regression.

Linear Least Squares Regression We apply least-square regression on the binary version of the \textit{cov} dataset (Collobert et al., 2002). This dataset contains \(n = 581,012\) datapoints, each described by \(d = 54\) input features.

Logistic Regression We apply logistic regression on the UCI adult and \textit{ijcnn1} datasets obtained from the \textit{libsvm} website \(^1\). The adult dataset contains \(n = 32,561\) datapoints, each described by \(d = 123\) input features. The \textit{ijcnn1} dataset contains \(n = 49,990\) datapoints, each described by \(d = 22\) input features.

We added an \(\ell_2\)-regularizer with parameter \(\lambda = 10^{-5}\) to ensure the objective is strongly convex.

The complexity of \(\text{VITE}\) given in Eq. 30 depends on three quantities, namely the approximate Hessian \(J\), the pair of stochastic gradients \((\nabla f_\mu(w), \nabla f_{\tilde{\mu}}(\tilde{w}))\) and \(\tilde{\mu}\), respectively computed over the sets \(A, B\) and \(D\). Similarly to (Mokhtari & Ribeiro, 2014), we consider different choices for the size of \(A\) of \(B\) and pick the best value in a limited interval \(\{1, \ldots, 0.05 \times n\}\). These results are also reported for the oBFGS method that also depends on both \(|A|\) and \(|B|\). For SGD, we use \(|B| = 1\) as we found this value to be the best performer on all datasets. For the \(\text{VITE}\) method, we also consider computing the average gradient \(\tilde{\mu}\) over a small subset \(C \subset D\). Although this introduces a bias in the computation of \(\tilde{\mu}\), it did not seem to practically affect convergence for sufficiently large sets \(C\). In practice, we randomly selected \(|C| = 0.1|D|\) samples for all datasets. Each experiment was averaged over 5 runs with different initializations of \(w_0\) and a random selection of the samples in the sets \(A, B\) and \(C\). Given that the complexity per iteration of each method is different, we compare them as a function of the number of gradient evaluations.

Figure 1 shows the empirical convergence properties of \(\text{VITE}\) against oBFGS for least-square regression and logistic regression. The horizontal axis corresponds to the number of gradient evaluations while the vertical axis corresponds to the objective function value. The vertical bars in each plot show the variance over 5 runs. We show plots for different initializations of \(w_0\) and a random selection of the samples in the sets \(A, B\), and \(C\). Given that the complexity per iteration of each method is different, we compare them as a function of the number of gradient evaluations.

We also show the effect of the set \(A\) on the convergence of oBFGS and \(\text{VITE}\) in Figure 2. At first increasing \(|A|\) results in a more accurate approximation to the Hessian and yields a better convergence rate. However, as we keep increasing \(|A|\), the penalty paid in terms of gradient evaluations outweighs the gain in terms of better curvature estimates.

We show a comparison of oBFGS and \(\text{VITE}\) against SGD and SVRG (Johnson & Zhang, 2013; Konečný & Richtárik, 2013) in Figure 3. A critical factor in the performance of SGD is the selection of the step-size. A relatively large value causes SGD to progress rapidly.

\(^1\)http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets
Figure 1. The red and green curves are the losses achieved by oBFGS and VITE respectively. Each experiment was averaged over 5 runs and the variance is shown on each plot with error bars. In the regime $|\mathcal{B}| \leq 0.1\%$, VITE has a much lower variance and reaches a lower optimum value. As we increase the size of $\mathcal{B}$, thus decreasing the variance of the stochastic gradients, the two methods start to behave similarly and both take many gradient evaluations to converge. Overall, we found VITE with $\mathcal{B} = 1$ and $\mathcal{B} = 0.1\%$ to perform the best.

Figure 2. Evolution of the objective value of oBFGS and VITE for different values of $|\mathcal{A}|$. We can see that the lowest value of $|\mathcal{A}|$ performs better, which indicates that there is no gain at increasing this value past a certain cut-off value.
at first but then leads to oscillations around the minimum without converging to it. On the other hand, with smaller step-sizes, SGD does eventually reach the minimum but can be very slow to do so. We here use the step-size given in Eq. 3b and pick the parameters $T_0$ and $\eta_0$ by performing cross-validation over $T_0 = \{1, 10, 10^2, \ldots, 10^4\}$ and $\eta_0 = \{10^{-1}, \ldots, 10^{-4}\}$. The plots shown in Figure 3 show that SGD exhibits a high variance at the beginning which is clearly magnified by the stochastic Hessian. As the step-size decreases, the variance of both SGD and oBFGS drops and these two methods reach convergence to a suboptimal value. On the other hand, VITE uses a constant step size and exhibits smaller variance, which leads to convergence to a lower objective value.

6. Conclusion

We have shown that stochastic variants of BFGS can be made more robust to the effects of randomness due to noisy stochastic gradients. The variance reduction technique we proposed is suitable for the stochastic optimization of smooth convex functions and we showed that it has a geometric convergence rate in expectation. Experimental results on three benchmark datasets showed that VITE affords important reduction in variance, which also translates into a reduction in terms of convergence time relative to stochastic gradient descent by the stochastic Hessian. The theoretical analysis we present is quite general and additionally only requires that the bound on the eigenvalues of the inverse Hessian matrix in (12) holds. Therefore, the variance reduced framework we propose can be extended to other quasi-Newton methods, including the widely used LBFGS and AdaGrad (Duchi et al., 2011) algorithms. Further investigations will explore the use of VITE to optimize generic convex or non-convex functions.

Acknowledgement We would like to thank Peter Richtarik for pointing out a mistake in the first version of Lemma 1.

7. Appendix

Proof of Lemma 1 Since $G$ is a symmetric matrix we have
\[
(a - b)^\top G(a - b) = a^\top G(a - b) - b^\top G(a - b) = a^\top Ga + b^\top Gb - 2a^\topGb,
\]
so
\[
a^\top Gb = \frac{a^\top Ga + b^\top Gb - (a - b)^\top G(a - b)}{2}. \tag{31}
\]
For a positive definite matrix $G$ with largest eigenvalue $\lambda_1$ and smallest eigenvalue $\lambda_d > 0$, the following holds for any vector $a$,
\[
a^\top Ga \leq \lambda_1 \|a\|^2, \quad a^\top Ga \geq \lambda_d \|a\|^2.
\]
Using these inequalities in Eq (31) yields
\[
a^\top Gb \geq \lambda_d (\|a\|^2 + \|b\|^2) - \lambda_1 \|a - b\|^2 \tag{32}
\]
\[
\geq \lambda_d \left(\|a\|^2 + \|b\|^2 - \lambda_1 (\|a\|^2 + \|b\|^2 - 2a^\top b)\right) \tag{33}
\]
\[
\geq \frac{\lambda_d - \lambda_1}{2}(\|a\|^2 + \|b\|^2) + \lambda_1 a^\top b.
\]

Proof of Lemma 3

\[
\mathbb{E}\|v_t\|^2 = \mathbb{E}\|\nabla f_i(w_{t-1}) - \nabla f_i(\hat{w}) + \nabla f(\hat{w})\|^2 \\
\leq 2\mathbb{E}\|\nabla f_i(w_{t-1}) - \nabla f_i(w^*)\|^2 + 2\mathbb{E}\|\nabla f_i(\hat{w}) - \nabla f_i(w^*)\|^2 \\
= 2\mathbb{E}\|\nabla f_i(w_{t-1}) - \nabla f_i(w^*)\|^2 + 2\mathbb{E}\|\nabla f_i(\hat{w}) - \nabla f_i(w^*)\|^2 \\
\leq 2\mathbb{E}\|\nabla f_i(w_{t-1}) - \nabla f_i(w^*)\|^2 + 2\mathbb{E}\|\nabla f_i(w_{t-1}) - \nabla f_i(w^*)\|^2 \\
\leq 4L(f(w_{t-1}) - f(w^*)) + f(\hat{w}) - f(w^*) \tag{32}
\]

The second inequality uses $\mathbb{E}\|\xi - \mathbb{E}\xi\|^2 = \mathbb{E}\|\xi\|^2 - \mathbb{E}\|\xi\|^2 \leq \mathbb{E}\|\xi\|^2$ for any random vector $\xi$.

The last inequality uses the following inequality derived from the fact that $f_i$ is a Lipschitz function:
\[
\mathbb{E}\|\nabla f_i(w^*) - \nabla f_i(w_t)\|^2 \leq 2L(f(w_t) - f(w^*)). 
\]
Figure 3. Comparison of oBFGS and VITE (trained with the best performing parameters) against SGD and SVRG. The reduction in variance for VITE is faster than SGD or oBFGS which typically lead to faster convergence.

References

Bach, Francis, Moulines, Eric, et al. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In Advances in Neural Information Processing Systems, pp. 451–459, 2011.

Bottou, Léon. Large-scale machine learning with stochastic gradient descent. In COMPSTAT, pp. 177–186. Springer, 2010.

Collobert, Ronan, Bengio, Samy, and Bengio, Yoshua. A parallel mixture of svms for very large scale problems. Neural computation, 14(5):1105–1114, 2002.

Defazio, Aaron, Bach, Francis, and Lacoste-Julien, Simon. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In Advances in Neural Information Processing Systems, pp. 1646–1654, 2014.

Dennis, Jr, John E and Moré, Jorge J. Quasi-newton methods, motivation and theory. SIAM review, 19(1):46–89, 1977.

Duchi, John, Hazan, Elad, and Singer, Yoram. Adaptive subgradient methods for online learning and stochastic optimization. The Journal of Machine Learning Research, 12:2121–2159, 2011.

Johnson, Rie and Zhang, Tong. Accelerating stochastic gradient descent using predictive variance reduction. In Advances in Neural Information Processing Systems, pp. 315–323, 2013.

Konečný, Jakub and Richtárik, Peter. Semi-stochastic gradient descent methods. arXiv preprint arXiv:1312.1666, 2013.

Lacoste-Julien, Simon, Schmidt, Mark, and Bach, Francis. A simpler approach to obtaining an o (1/t) convergence rate for the projected stochastic subgradient method. arXiv preprint arXiv:1212.2002, 2012.

Liu, Dong C and Nocedal, Jorge. On the limited memory bfgs method for large scale optimization. Mathematical programming, 45(1-3):503–528, 1989.

Mokhtari, Aryan and Ribeiro, Alejandro. Res: Regularized stochastic bfgs algorithm. arXiv preprint arXiv:1401.7625, 2014.

Nocedal, J and Wright, SJ. Numerical optimization, volume 2. Springer New York, 1999.

Rakhlin, Alexander, Shamir, Ohad, and Sridharan, Karthik. Making gradient descent optimal for strongly convex stochastic optimization. arXiv preprint arXiv:1109.5647, 2011.

Robbins, Herbert and Monro, Sutton. A stochastic approximation method. The annals of mathematical statistics, pp. 400–407, 1951.

Roux, Nicolas L, Schmidt, Mark, and Bach, Francis R. A stochastic gradient method with an exponential convergence rate for finite training sets. In Advances in Neural Information Processing Systems, pp. 2663–2671, 2012.

Schraudolph, Nicol, Yu, Jin, and Günter, Simon. A stochastic quasi-newton method for online convex optimization. In Intl. Conf. Artificial Intelligence and Statistics (Aistats), 2007.

Shalev-Shwartz, Shai, Singer, Yoram, Srebro, Nathan, and Cotter, Andrew. Pegasos: Primal estimated sub-gradient solver for svm. Mathematical programming, 127(1):3–30, 2011.