Stochastic Optimization Algorithms Beyond SG

Frank E. Curtis, Lehigh University

involving joint work with

Léon Bottou, Facebook AI Research
Jorge Nocedal, Northwestern University
“Optimization Methods for Large-Scale Machine Learning”
http://arxiv.org/abs/1606.04838

ExxonMobil Research, Annandale, NJ

27 July 2017
Outline

GD and SG

GD vs. SG

Beyond SG

Stochastic Quasi-Newton

Self-Correcting Properties of BFGS

Proposed Algorithm: SC-BFGS

Summary
Stochastic optimization

Over a parameter vector $w \in \mathbb{R}^d$ and given

$$\ell(\cdot; y) \circ h(w; x) \quad \text{(loss w.r.t. “true label”} \circ \text{prediction w.r.t. “features”)},$$

consider the unconstrained optimization problem

$$\min_{w \in \mathbb{R}^d} f(w), \quad \text{where} \quad f(w) = \mathbb{E}_{(x, y)}[\ell(h(w; x), y)].$$
Stochastic optimization

Over a parameter vector \( w \in \mathbb{R}^d \) and given

\[
\ell(\cdot; y) \circ h(w; x) \quad \text{(loss w.r.t. “true label” \( \circ \) prediction w.r.t. “features”),}
\]

consider the unconstrained optimization problem

\[
\min_{w \in \mathbb{R}^d} f(w), \quad \text{where} \quad f(w) = \mathbb{E}_{(x, y)}[\ell(h(w; x), y)].
\]

Given training set \( \{(x_i, y_i)\}_{i=1}^n \), approximate problem given by

\[
\min_{w \in \mathbb{R}^d} f_n(w), \quad \text{where} \quad f_n(w) = \frac{1}{n} \sum_{i=1}^n \ell(h(w; x_i), y_i).
\]
Stochastic optimization

Over a parameter vector $w \in \mathbb{R}^d$ and given

$$\ell(\cdot; y) \circ h(w; x)$$ (loss w.r.t. “true label” \circ prediction w.r.t. “features”),

consider the unconstrained optimization problem

$$\min_{w \in \mathbb{R}^d} f(w), \quad \text{where} \quad f(w) = \mathbb{E}_{(x, y)} [\ell(h(w; x), y)].$$

Given training set $\{(x_i, y_i)\}_{i=1}^n$, approximate problem given by

$$\min_{w \in \mathbb{R}^d} f_n(w), \quad \text{where} \quad f_n(w) = \frac{1}{n} \sum_{i=1}^n \ell(h(w; x_i), y_i).$$

For this talk, let’s assume

- $f$ is continuously differentiable, bounded below, and potentially nonconvex;
- $\nabla f$ is $L$-Lipschitz continuous, i.e., $\|\nabla f(w) - \nabla f(\bar{w})\|_2 \leq L\|w - \bar{w}\|_2$.

Focus on optimization algorithms, not data fitting issues, regularization, etc.
Gradient descent

Algorithm GD : Gradient Descent

1: choose an initial point $w_0 \in \mathbb{R}^n$ and stepsize $\alpha > 0$
2: for $k \in \{0, 1, 2, \ldots \}$ do
3: set $w_{k+1} \leftarrow w_k - \alpha \nabla f(w_k)$
4: end for
**Algorithm GD : Gradient Descent**

1. choose an initial point $w_0 \in \mathbb{R}^n$ and stepsize $\alpha > 0$
2. for $k \in \{0, 1, 2, \ldots \}$ do
3. set $w_{k+1} \leftarrow w_k - \alpha \nabla f(w_k)$
4. end for

$$f(w_k) + \nabla f(w_k)^T(w - w_k) + \frac{1}{2}L\|w - w_k\|^2_2$$

$$f(w)\Leftarrow f(w)\Leftarrow f(w)\Leftarrow f(w)\Leftarrow f(w)$$

$w_k$
**Algorithm GD : Gradient Descent**

1: choose an initial point $w_0 \in \mathbb{R}^n$ and stepsize $\alpha > 0$
2: for $k \in \{0, 1, 2, \ldots\}$ do
3: set $w_{k+1} \leftarrow w_k - \alpha \nabla f(w_k)$
4: end for

\[ f(w_k) + \nabla f(w_k)^T (w - w_k) + \frac{1}{2} L \| w - w_k \|^2 \]
Algorithm GD: Gradient Descent

1: choose an initial point $w_0 \in \mathbb{R}^n$ and stepsize $\alpha > 0$
2: for $k \in \{0, 1, 2, \ldots\}$ do
3: set $w_{k+1} \leftarrow w_k - \alpha \nabla f(w_k)$
4: end for

$$f(w_k) + \nabla f(w_k)^T (w - w_k) + \frac{1}{2} L \|w - w_k\|^2$$

$$f(w_k) + \nabla f(w_k)^T (w - w_k) + \frac{1}{2} c \|w - w_k\|^2$$
GD theory

Theorem GD

If $\alpha \in (0, 1/L]$, then $\sum_{k=0}^{\infty} \|\nabla f(w_k)\|^2_2 < \infty$, which implies $\{\nabla f(w_k)\} \to 0$.

Proof.

$$f(w_{k+1}) \leq f(w_k) + \nabla f(w_k)^T(w_{k+1} - w_k) + \frac{1}{2} L \|w_{k+1} - w_k\|^2_2$$

$$\leq f(w_k) - \frac{1}{2} \alpha \|\nabla f(w_k)\|^2_2$$
**Theorem GD**

If \( \alpha \in (0, 1/L] \), then \( \sum_{k=0}^{\infty} \|\nabla f(w_k)\|^2 \leq \infty \), which implies \( \{\nabla f(w_k)\} \to 0 \).

If, in addition, \( f \) is \( c \)-strongly convex, then for all \( k \geq 1 \):

\[
f(w_k) - f_* \leq (1 - \alpha c)^k (f(x_0) - f_*).
\]

**Proof.**

\[
f(w_{k+1}) \leq f(w_k) + \nabla f(w_k)^T (w_{k+1} - w_k) + \frac{1}{2} L \|w_{k+1} - w_k\|^2
\]

\[
\leq f(w_k) - \frac{1}{2} \alpha \|\nabla f(w_k)\|^2
\]

\[
\leq f(w_k) - \alpha c (f(w_k) - f_*).
\]

\[
\implies f(w_{k+1}) - f_* \leq (1 - \alpha c)(f(w_k) - f_*).
\]
GD illustration

Figure: GD with fixed stepsize
Stochastic gradient descent

Approximate gradient only; e.g., random \( i_k \) and \( \nabla_w \ell(h(w; x_{i_k}), y_{i_k}) \approx \nabla f(w) \).

**Algorithm SG**: Stochastic Gradient

1: choose an initial point \( w_0 \in \mathbb{R}^n \) and stepsizes \( \{\alpha_k\} > 0 \)
2: for \( k \in \{0, 1, 2, \ldots\} \) do
3: \hspace{1em} set \( w_{k+1} \leftarrow w_k - \alpha_k g_k \), where \( g_k \approx \nabla f(w_k) \)
4: end for
Approximate gradient only; e.g., random $i_k$ and $\nabla_w \ell(h(w; x_{i_k}), y_{i_k}) \approx \nabla f(w)$.

**Algorithm SG** : Stochastic Gradient

1: choose an initial point $w_0 \in \mathbb{R}^n$ and stepsizes $\{\alpha_k\} > 0$
2: **for** $k \in \{0, 1, 2, \ldots \}$ **do**
3: set $w_{k+1} \leftarrow w_k - \alpha_k g_k$, where $g_k \approx \nabla f(w_k)$
4: **end for**

Not a descent method!

...but can guarantee *eventual descent in expectation* (with $\mathbb{E}_k [g_k] = \nabla f(w_k)$):

$$f(w_{k+1}) \leq f(w_k) + \nabla f(w_k)^T(w_{k+1} - w_k) + \frac{1}{2} L \|w_{k+1} - w_k\|^2_2$$

$$= f(w_k) - \alpha_k \nabla f(w_k)^T g_k + \frac{1}{2} \alpha_k^2 L \|g_k\|^2_2$$

$$\implies \mathbb{E}_k [f(w_{k+1})] \leq f(w_k) - \alpha_k \|\nabla f(w_k)\|^2_2 + \frac{1}{2} \alpha_k^2 L \mathbb{E}_k [\|g_k\|^2_2].$$

Markov process: $w_{k+1}$ depends only on $w_k$ and random choice at iteration $k$. 
SG theory

Theorem SG

If $\mathbb{E}_k[\|g_k\|^2_2] \leq M + \|\nabla f(w_k)\|^2_2$, then:

$$\alpha_k = \frac{1}{L} \quad \Rightarrow \quad \mathbb{E}\left[\frac{1}{k} \sum_{j=1}^{k} \|\nabla f(w_j)\|^2_2\right] \to M$$

$$\alpha_k = \mathcal{O}\left(\frac{1}{k}\right) \quad \Rightarrow \quad \mathbb{E}\left[\sum_{j=1}^{k} \alpha_j \|\nabla f(w_j)\|^2_2\right] < \infty.$$  

(*Assumed unbiased gradient estimates; see paper for more generality.*)
Theorem SG

If $\mathbb{E}_k[\|g_k\|_2^2] \leq M + \|\nabla f(w_k)\|_2^2$, then:

$$
\alpha_k = \frac{1}{L} \implies \mathbb{E}\left[ \frac{1}{k} \sum_{j=1}^{k} \|\nabla f(w_j)\|_2^2 \right] \to M
$$

$$
\alpha_k = \mathcal{O}\left(\frac{1}{k}\right) \implies \mathbb{E}\left[ \sum_{j=1}^{k} \alpha_j \|\nabla f(w_j)\|_2^2 \right] < \infty.
$$

If, in addition, $f$ is $c$-strongly convex, then:

$$
\alpha_k = \frac{1}{L} \implies \mathbb{E}[f(w_k) - f_*] \to \frac{(M/c)}{2}
$$

$$
\alpha_k = \mathcal{O}\left(\frac{1}{k}\right) \implies \mathbb{E}[f(w_k) - f_*] = \mathcal{O}\left(\frac{(L/c)(M/c)}{k}\right).
$$

(*Assumed unbiased gradient estimates; see paper for more generality.*)
SG illustration

Figure: SG with fixed stepsize (left) vs. diminishing stepsizes (right)
Outline

GD and SG

GD vs. SG

Beyond SG

Stochastic Quasi-Newton

Self-Correcting Properties of BFGS

Proposed Algorithm: SC-BFGS

Summary
Why SG over GD for large-scale machine learning?

We have seen:

GD: \[ \mathbb{E}[f_n(w_k) - f_{n,*}] = \mathcal{O}(\rho^k) \] linear convergence

SG: \[ \mathbb{E}[f_n(w_k) - f_{n,*}] = \mathcal{O}(1/k) \] sublinear convergence

So why SG?
Why SG over GD for large-scale machine learning?

We have seen:

GD: \[ \mathbb{E}[f_n(w_k) - f_n,\ast] = \mathcal{O}(\rho^k) \] linear convergence

SG: \[ \mathbb{E}[f_n(w_k) - f_n,\ast] = \mathcal{O}(1/k) \] sublinear convergence

So why SG?

<table>
<thead>
<tr>
<th>Motivation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intuitive</td>
<td>data “redundancy”</td>
</tr>
<tr>
<td>Empirical</td>
<td>SG vs. L-BFGS with batch gradient (below)</td>
</tr>
<tr>
<td>Theoretical</td>
<td>[ \mathbb{E}[f_n(w_k) - f_n,\ast] = \mathcal{O}(1/k) ] [ \mathbb{E}[f(w_k) - f_\ast] = \mathcal{O}(1/k) ]</td>
</tr>
</tbody>
</table>

![Graph showing comparison between SGD and LBFGS on accessed data points and empirical risk over iterations.](image-url)
Work complexity

Time, not data, as limiting factor; Bottou, Bousquet (2008) and Bottou (2010).

<table>
<thead>
<tr>
<th>Convergence rate</th>
<th>Cost per iteration</th>
<th>Cost for $\epsilon$-optimality</th>
</tr>
</thead>
<tbody>
<tr>
<td>GD: $E[f_n(w_k) - f_n,*] = O(\rho^k)$</td>
<td>$O(n)$</td>
<td>$n \log(1/\epsilon)$</td>
</tr>
<tr>
<td>SG: $E[f_n(w_k) - f_n,*] = O(1/k)$</td>
<td>$O(1)$</td>
<td>$1/\epsilon$</td>
</tr>
</tbody>
</table>
Work complexity

Time, not data, as limiting factor; Bottou, Bousquet (2008) and Bottou (2010).

<table>
<thead>
<tr>
<th>Convergence rate</th>
<th>Cost per iteration</th>
<th>Cost for $\epsilon$-optimality</th>
</tr>
</thead>
<tbody>
<tr>
<td>GD: $\mathbb{E}[f_n(w_k) - f_{n,*}] = O(\rho^k)$ + $O(n)$</td>
<td>$n \log(1/\epsilon)$</td>
<td></td>
</tr>
<tr>
<td>SG: $\mathbb{E}[f_n(w_k) - f_{n,*}] = O(1/k)$ + $O(1)$</td>
<td>$1/\epsilon$</td>
<td></td>
</tr>
</tbody>
</table>

Considering total (estimation + optimization) error as

$$\mathcal{E} = \mathbb{E}[f(w^n) - f(w^*)] + \mathbb{E}[f(\tilde{w}^n) - f(w^n)] \sim \frac{1}{n} + \epsilon$$

and a time budget $T$, one finds:

- **SG:** Process as many samples as possible ($n \sim T$), leading to
  $$\mathcal{E} \sim \frac{1}{T}.$$

- **GD:** With $n \sim T / \log(1/\epsilon)$, minimizing $\mathcal{E}$ yields $\epsilon \sim 1/T$ and
  $$\mathcal{E} \sim \frac{1}{T} + \frac{\log(T)}{T}.$$
Outline

GD and SG

GD vs. SG

Beyond SG

Stochastic Quasi-Newton

Self-Correcting Properties of BFGS

Proposed Algorithm: SC-BFGS

Summary
End of the story?

SG is great! Let’s keep proving how great it is!

- SG avoids steep minima; Keskar, Mudigere, Nocedal, Smelyanskiy (2016)
- ... (many more)
End of the story?

SG is great! Let’s keep proving how great it is!

- SG avoids steep minima; Keskar, Mudigere, Nocedal, Smelyanskiy (2016)
- … (many more)

No, we should want more…

- SG requires a lot of tuning
- Sublinear convergence is not satisfactory
- … “linearly” convergent method eventually wins
- … with higher budget, faster computation, parallel?, distributed?

Also, any “gradient”-based method is not scale invariant.
What can be improved?

- Stochastic gradient
- Better rate
- Better constant
What can be improved?

- Stochastic gradient
- Better rate
- Better constant

Better rate and better constant
Two-dimensional schematic of methods

- Stochastic gradient
- Batch gradient
- Noise reduction
- Second-order

Stochastic Newton

Batch Newton
2D schematic: Noise reduction methods

- stochastic gradient
- noise reduction
  - dynamic sampling
  - gradient aggregation
  - iterate averaging
- batch gradient
2D schematic: Second-order methods

- stochastic gradient
  - diagonal scaling
  - natural gradient
  - Gauss-Newton
  - quasi-Newton
  - Hessian-free Newton
Even more...

- momentum
- acceleration
- (dual) coordinate descent
- trust region / step normalization
- exploring negative curvature
Outline

GD and SG

GD vs. SG

Beyond SG

Stochastic Quasi-Newton

Self-Correcting Properties of BFGS

Proposed Algorithm: SC-BFGS

Summary
Scale invariance

Neither SG nor GD are invariant to linear transformations.

\[
\begin{align*}
\min_{w \in \mathbb{R}^d} f(w) & \quad \implies \quad w_{k+1} \leftarrow w_k - \alpha_k \nabla f(w_k) \\
\min_{\tilde{w} \in \mathbb{R}^d} f(B\tilde{w}) & \quad \implies \quad \tilde{w}_{k+1} \leftarrow \tilde{w}_k - \alpha_k B \nabla f(B\tilde{w}_k) \quad \text{(for given } B \succ 0)\
\end{align*}
\]
Neither SG nor GD are invariant to linear transformations.

\[
\min_{w \in \mathbb{R}^d} f(w) \quad \Rightarrow \quad w_{k+1} \leftarrow w_k - \alpha_k \nabla f(w_k)
\]

\[
\min_{\tilde{w} \in \mathbb{R}^d} f(B\tilde{w}) \quad \Rightarrow \quad \tilde{w}_{k+1} \leftarrow \tilde{w}_k - \alpha_k B \nabla f(B\tilde{w}_k) \quad \text{(for given } B \succ 0)\]

Scaling latter by \(B\) and defining \(\{w_k\} = \{B\tilde{w}_k\}\) yields

\[
w_{k+1} \leftarrow w_k - \alpha_k B^2 \nabla f(w_k)
\]

- Algorithm is clearly affected by choice of \(B\)
- Surely, some choices may be better than others
Consider the function below and suppose that $w_k = (0, 3)$:
Newton scaling

GD step along $-\nabla f(w_k)$ ignores curvature of the function:
Newton scaling

Newton scaling \((B = (\nabla^2 f(w_k))^{-1/2})\): gradient step moves to the minimizer:

\[
\begin{align*}
    w_{k+1} &\leftarrow w_k + \alpha_k s_k \\
    s_k &= -\nabla f(w_k)
\end{align*}
\]
Newton scaling

...corresponds to minimizing a quadratic model of $f$ in the original space:

$$w_{k+1} \leftarrow w_k + \alpha_k s_k \quad \text{where} \quad \nabla^2 f(w_k)s_k = -\nabla f(w_k)$$
What is known about Newton’s method for deterministic optimization?

- local rescaling based on inverse Hessian information
- unit steps are good near strong minimizer (no tuning!)
- ... locally quadratically convergent
- global convergence rate better than gradient method (when regularized)
Deterministic case to stochastic case

What is known about Newton’s method for deterministic optimization?
- local rescaling based on inverse Hessian information
- unit steps are good near strong minimizer (no tuning!)
- ... locally quadratically convergent
- global convergence rate better than gradient method (when regularized)

However, it is way too expensive.
- But all is not lost: scaling can be practical.
- Wide variety of scaling techniques improve performance.
- ...could hope to remove condition number \((L/c)\) from convergence rate!
- Added costs can be minimal when coupled with noise reduction.
Quasi-Newton

Only *approximate* second-order information with gradient displacements:

Secant equation $H_k v_k = s_k$ to match gradient of $f$ at $w_k$, where

$$s_k := w_{k+1} - w_k \quad \text{and} \quad v_k := \nabla f(w_{k+1}) - \nabla f(w_k)$$
Balance between extremes

For deterministic, smooth optimization, a nice balance achieved by quasi-Newton:

\[ w_{k+1} \leftarrow w_k - \alpha_k M_k g_k, \]

where

- \( \alpha_k > 0 \) is a stepsize;
- \( g_k \leftarrow \nabla f(w_k) \);
- \( \{M_k\} \) is updated dynamically.

Background on quasi-Newton:

- local rescaling of step (overcome ill-conditioning)
- only first-order derivatives required
- no linear system solves required
- global convergence guarantees (say, with line search)
- superlinear local convergence rate

How can the idea be carried over to a stochastic setting?
Previous work: BFGS-type methods

Much focus on the secant equation \((H_{k+1} \sim \text{Hessian approximation})\)

\[
H_{k+1} s_k = y_k \quad \text{where} \quad s_k := w_{k+1} - w_k \\
y_k := \nabla f(w_{k+1}) - \nabla f(w_k)
\]

and an appropriate replacement for the gradient displacement:

\[
y_k \leftarrow \frac{1}{|S|} \sum_{i \in S} \nabla^2 f(w_{k+1}, \xi, \xi_{k+1}, i) s_k
\]

\(\text{use action of step on subsampled Hessian}\)

\(\text{SQN, Byrd et al. (2015)}\)

\(\text{oLBFGS, Schraudolph et al. (2007)}\)

\(\text{SGD-QN, Bordes et al. (2009)}\)

\(\text{RES, Mokhtari & Ribeiro (2014)}\)

Is this the right focus? Is there a better way (especially for nonconvex \(f\))?
Proposal

Propose a quasi-Newton method for stochastic (nonconvex) optimization

- exploit self-correcting properties of BFGS-type updates
  - Powell (1976)
  - Ritter (1979, 1981)
  - Werner (1978)
  - Byrd, Nocedal (1989)
- properties of Hessians offer useful bounds for inverse Hessians
- motivating convergence theory for convex and nonconvex objectives
- dynamic noise reduction strategy
- limited memory variant

Observed stable behavior and overall good performance
Outline

GD and SG

GD vs. SG

Beyond SG

Stochastic Quasi-Newton

Self-Correcting Properties of BFGS

Proposed Algorithm: SC-BFGS

Summary
BFGS-type updates

Inverse Hessian and Hessian approximation updating formulas ($s_k^Tv_k > 0$):

\[
M_{k+1} \leftarrow \left( I - \frac{v_k s_k^T}{s_k^Tv_k} \right)^T M_k \left( I - \frac{v_k s_k^T}{s_k^Tv_k} \right) + \frac{s_k s_k^T}{s_k^Tv_k}
\]

\[
H_{k+1} \leftarrow \left( I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \right)^T H_k \left( I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \right) + \frac{v_k v_k^T}{s_k^Tv_k}
\]

- Satisfy secant-type equations

\[M_{k+1}v_k = s_k \quad \text{and} \quad H_{k+1}s_k = v_k,
\]

but these are not relevant for our purposes here.

- Choosing $v_k \leftarrow y_k := g_{k+1} - g_k$ yields standard BFGS, but in this talk

\[v_k \leftarrow \beta_k s_k + (1 - \beta_k) \alpha_k y_k \quad \text{for some} \quad \beta_k \in [0, 1].
\]

This scheme is important to preserve self-correcting properties.
Geometric properties of Hessian update

Consider the matrices (which only depend on $s_k$ and $H_k$, not $g_k$!)

$$P_k := \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \quad \text{and} \quad Q_k := I - P_k.$$ 

Both $H_k$-orthogonal projection matrices (i.e., idempotent and $H_k$-self-adjoint).

- $P_k$ yields $H_k$-orthogonal projection onto span($s_k$).
- $Q_k$ yields $H_k$-orthogonal projection onto span($s_k$)$^\perp H_k$. 

Geometric properties of Hessian update

Consider the matrices (which only depend on $s_k$ and $H_k$, not $g_k$!)

$$P_k := \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \quad \text{and} \quad Q_k := I - P_k.$$ 

Both $H_k$-orthogonal projection matrices (i.e., idempotent and $H_k$-self-adjoint).

- $P_k$ yields $H_k$-orthogonal projection onto $\text{span}(s_k)$.
- $Q_k$ yields $H_k$-orthogonal projection onto $\text{span}(s_k)_{\perp H_k}$.

Returning to the Hessian update:

$$H_{k+1} \leftarrow \left( I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \right)^T H_k \left( I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \right) + \frac{v_k v_k^T}{s_k^T v_k}$$

- Curvature projected out along $\text{span}(s_k)$
- Curvature corrected by $\frac{v_k v_k^T}{s_k^T v_k} = \left( \frac{v_k v_k^T}{\|v_k\|^2} \right) \left( \frac{\|v_k\|^2}{v_k^T M_{k+1} v_k} \right)$ (inverse Rayleigh).
Self-correcting properties of Hessian update

Since curvature is constantly projected out, what happens after many updates?

Theorem SC (Byrd, Nocedal (1989))

Suppose that, for all $k$, there exists $\{\eta, \theta\} \subseteq \mathbb{R}^{++}$ such that

\[
\eta \leq s_k^T v_k \|s_k\|_2^2 \quad \text{and} \quad \|v_k\|_2^2 s_k^T v_k \leq \theta.
\]

(\text{KEY})

Then, for any $p \in (0, 1)$, there exist constants $\{\iota, \kappa, \lambda\} \subseteq \mathbb{R}^{++}$ such that, for any $K \geq 2$, the following relations hold for at least $\lceil pK \rceil$ values of $k \in \{1, \ldots, K\}$:

\[
\iota \leq s_k^T H_k s_k \|s_k\|_2 \|H_k s_k\|_2 \quad \text{and} \quad \kappa \leq \|H_k s_k\|_2 \|s_k\|_2 \leq \lambda.
\]

Proof technique.

Building on work of Powell (1976), etc., involves bounding growth of $\gamma(H_k) = \text{tr}(H_k) - \ln(\det(H_k))$. 
Self-correcting properties of Hessian update

Since curvature is constantly projected out, what happens after many updates?

**Theorem SC (Byrd, Nocedal (1989))**

Suppose that, for all \( k \), there exists \( \{\eta, \theta\} \subset \mathbb{R}_{++} \) such that

\[
\eta \leq \frac{s_k^T v_k}{\|s_k\|_2^2} \quad \text{and} \quad \frac{\|v_k\|_2^2}{s_k^T v_k} \leq \theta.
\]  

(\text{KEY})

Then, for any \( p \in (0, 1) \), there exist constants \( \{\iota, \kappa, \lambda\} \subset \mathbb{R}_{++} \) such that, for any \( K \geq 2 \), the following relations hold for at least \( \lceil pK \rceil \) values of \( k \in \{1, \ldots, K\} \):

\[
\iota \leq \frac{s_k^T H_k s_k}{\|s_k\|_2 \|H_k s_k\|_2} \quad \text{and} \quad \kappa \leq \frac{\|H_k s_k\|_2}{\|s_k\|_2} \leq \lambda.
\]

**Proof technique.**

Building on work of Powell (1976), etc., involves bounding growth of

\[
\gamma(H_k) = \text{tr}(H_k) - \ln(\det(H_k)).
\]
Self-correcting properties of inverse Hessian update

Rather than focus on superlinear convergence results, we care about the following.

**Corollary SC**

Suppose the conditions of Theorem SC hold. Then, for any $p \in (0,1)$, there exist constants $\{\mu, \nu\} \subset \mathbb{R}_{++}$ such that, for any $K \geq 2$, the following relations hold for at least $\lceil pK \rceil$ values of $k \in \{1, \ldots, K\}$:

$$
\mu \|g_k\|_2^2 \leq g_k^T M_k g_k \quad \text{and} \quad \|M_k g_k\|_2^2 \leq \nu \|g_k\|_2^2
$$

**Proof sketch.**

Follows simply after algebraic manipulations from the result of Theorem SC, using the facts that $s_k = -\alpha_k M_k g_k$ and $M_k = H_k^{-1}$ for all $k$. 
Outline

GD and SG

GD vs. SG

Beyond SG

Stochastic Quasi-Newton

Self-Correcting Properties of BFGS

Proposed Algorithm: SC-BFGS

Summary
Algorithm SC: Self-Correcting BFGS Algorithm

1: Choose $w_1 \in \mathbb{R}^d$.
2: Set $g_1 \approx \nabla f(w_1)$.
3: Choose a symmetric positive definite $M_1 \in \mathbb{R}^{d \times d}$.
4: Choose a positive scalar sequence $\{\alpha_k\}$.
5: for $k = 1, 2, \ldots$ do
6: Set $s_k \leftarrow -\alpha_k M_k g_k$.
7: Set $w_{k+1} \leftarrow w_k + s_k$.
8: Set $g_{k+1} \approx \nabla f(w_{k+1})$.
9: Set $y_k \leftarrow g_{k+1} - g_k$.
10: Set $\beta_k \leftarrow \min\{\beta \in [0, 1] : v(\beta) := \beta s_k + (1 - \beta) \alpha_k y_k \text{ satisfies (KEY)}\}$.
11: Set $v_k \leftarrow v(\beta_k)$.
12: Set
   \[
   M_{k+1} \leftarrow \left( I - \frac{v_k s_k^T}{s_k^T v_k} \right)^T M_k \left( I - \frac{v_k s_k^T}{s_k^T v_k} \right) + \frac{s_k s_k^T}{s_k^T v_k} .
   \]
13: end for
Global convergence theorem

**Theorem (Bottou, Curtis, Nocedal (2016))**

Suppose that, for all $k$, there exists a scalar constant $\rho > 0$ such that

$$-\nabla f(w_k)^T \mathbb{E}_{\xi_k}[M_k g_k] \leq -\rho \|\nabla f(w_k)\|_2^2,$$

and there exist scalars $\sigma > 0$ and $\tau > 0$ such that

$$\mathbb{E}_{\xi_k}[\|M_k g_k\|_2^2] \leq \sigma + \tau \|\nabla f(w_k)\|_2^2.$$

Then, $\{\mathbb{E}[f(w_k)]\}$ converges to a finite limit and

$$\lim_{k \to \infty} \mathbb{E}[\nabla f(w_k)] = 0.$$

**Proof technique.**

Follows from the critical inequality

$$\mathbb{E}_{\xi_k}[f(w_{k+1})] - f(w_k) \leq -\alpha_k \nabla f(w_k)^T \mathbb{E}_{\xi_k}[M_k g_k] + \alpha_k^2 L \mathbb{E}_{\xi_k}[\|M_k g_k\|_2^2].$$
The conditions in this theorem cannot be verified in practice.

- They require knowing $\nabla f(w_k)$.
- They require knowing $\mathbb{E}_{\xi_k}[M_kg_k]$ and $\mathbb{E}_{\xi_k}[\|M_kg_k\|_2^2]$.
- ... but $M_k$ and $g_k$ are not independent!
- That said, Corollary SC ensures that they hold with $g_k = \nabla f(w_k)$; recall

$$
\mu \|g_k\|_2^2 \leq g_k^T M_k g_k \quad \text{and} \quad \|M_k g_k\|_2^2 \leq \nu \|g_k\|_2^2.
$$
Reality

The conditions in this theorem cannot be verified in practice.

- They require knowing $\nabla f(w_k)$.
- They require knowing $\mathbb{E}_{\xi_k}[M_k g_k]$ and $\mathbb{E}_{\xi_k}[\|M_k g_k\|^2]$.
- ... but $M_k$ and $g_k$ are not independent!
- That said, Corollary SC ensures that they hold with $g_k = \nabla f(w_k)$; recall

$$\mu \|g_k\|_2^2 \leq g_k^T M_k g_k \quad \text{and} \quad \|M_k g_k\|_2^2 \leq \nu \|g_k\|_2^2.$$ 

**Stabilized variant (SC-s):** Loop over (stochastic) gradient computation until

$$\rho \|\hat{g}_{k+1}\|_2^2 \leq \hat{g}_{k+1}^T M_{k+1} g_{k+1}$$

and

$$\|M_{k+1} g_{k+1}\|_2^2 \leq \sigma + \tau \|\hat{g}_{k+1}\|_2^2.$$ 

Recompute $g_{k+1}$, $\hat{g}_{k+1}$, and $M_{k+1}$ until these hold.
Numerical Experiments: \textit{a1a}

logistic regression, data \textit{a1a}, diminishing stepsizes
Numerical Experiments: rcv1

SC-L and SC-L-s: limited memory variants of SC and SC-s, respectively:

logistic regression, data rcv1, diminishing stepsizes
Numerical Experiments: mnist

deep neural network, data mnist, diminishing stepsizes
Outline

GD and SG

GD vs. SG

Beyond SG

Stochastic Quasi-Newton

Self-Correcting Properties of BFGS

Proposed Algorithm: SC-BFGS

Summary
Contributions

Proposed a quasi-Newton method for stochastic (nonconvex) optimization

- exploited self-correcting properties of BFGS-type updates
- properties of Hessians offer useful bounds for inverse Hessians
- motivating convergence theory for convex and nonconvex objectives
- dynamic noise reduction strategy
- limited memory variant

Observed stable behavior and overall good performance

☆ F. E. Curtis.
A Self-Correcting Variable-Metric Algorithm for Stochastic Optimization.