Stochastic Composition Optimization
Algorithms and Sample Complexities

Mengdi Wang
Joint works with Ethan X. Fang, Han Liu, and Ji Liu

ORFE@Princeton

ICCOPT, Tokyo, August 8-11, 2016
Collaborators

1. Background: Why is SGD a good method?

2. A New Problem: Stochastic Composition Optimization

3. Stochastic Composition Algorithms: Convergence and Sample Complexity

4. Acceleration via Smoothing-Extrapolation
Outline

1. Background: Why is SGD a good method?

2. A New Problem: Stochastic Composition Optimization

3. Stochastic Composition Algorithms: Convergence and Sample Complexity

4. Acceleration via Smoothing-Extrapolation
Background

- Machine learning is optimization

<table>
<thead>
<tr>
<th>Learning from batch data</th>
<th>Learning from online data</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\min_{x \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(x; A_i, b_i) + \rho(x)$</td>
<td>$\min_{x \in \mathbb{R}^d} E_{\mathcal{A}, b} [\ell(x; A, b)] + \rho(x)$</td>
</tr>
</tbody>
</table>

- Both problems can be formulated as Stochastic Convex Optimization

$$\min_{x} E [f(x, \xi)]$$

expectation over batch data set or unknown distribution

- A general framework encompasses likelihood estimation, online learning, empirical risk minimization, multi-arm bandit, online MDP

- Stochastic gradient descent (SGD) updates by taking sample gradients:

$$x_{k+1} = x_k - \alpha \nabla f(x_k, \xi_k)$$

A special case of stochastic approximation with a long history (Robbins and Monro, Kushner and Yin, Polyak and Juditsky, Benveniste et al., Ruszcyński, Borkar, Bertsekas and Tsitsiklis, and many)
Background: Stochastic first-order methods

- Stochastic gradient descent (SGD) updates by taking sample gradients:
  \[ x_{k+1} = x_k - \alpha \nabla f(x_k, \xi_k) \]

(1,410,000 results on Google Scholar Search and 24,400 since 2016!)

Why is SGD a good method in practice?

- When processing either batch or online data, a scalable algorithm needs to update using partial information (a small subset of all data)
- Answer: We have no other choice

Why is SGD a good method beyond practical reasons?

- SGD achieves optimal convergence after processing \( k \) samples:
  \[ \mathbb{E} [F(x_k) - F^*] = \mathcal{O}(1/\sqrt{k}) \] for convex minimization
  \[ \mathbb{E} [F(x_k) - F^*] = \mathcal{O}(1/k) \] for strongly convex minimization


- Beyond convexity: nearly optimal online PCA (Li, Wang, Liu, Zhang 2015)
- Answer: Strong theoretical guarantees for data-driven problems
Outline

1. Background: Why is SGD a good method?

2. A New Problem: Stochastic Composition Optimization

3. Stochastic Composition Algorithms: Convergence and Sample Complexity

4. Acceleration via Smoothing-Extrapolation
Consider the problem

$$\min_{x \in \mathcal{X}} \left\{ F(x) := (f \circ g)(x) = f(g(x)) \right\},$$

where the outer and inner functions are $f : \mathbb{R}^m \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$

$$f(y) = \mathbb{E}[f_v(y)], \quad g(x) = \mathbb{E}[g_w(x)],$$

and $\mathcal{X}$ is a closed and convex set in $\mathbb{R}^n$.

- We focus on the case where the overall problem is convex (for now)
- No structural assumptions on $f$, $g$ (nonconvex/nonmonotone/nondifferentiable)
- We may not know the distribution of $v$, $w$. 
Expectation Minimization vs. Stochastic Composition Optimization

Recall the classical problem:

$$\min_{x \in \mathcal{X}} \mathbb{E}[f(x, \xi)]$$

linear w.r.t the distribution of $\xi$

In stochastic composition optimization, the objective is no longer a linear functional of the $(v, w)$ distribution:

$$\min_{x \in \mathcal{X}} \mathbb{E}[f_v(\mathbb{E}[g_w(x)])]$$

nonlinear w.r.t the distribution of $(w, v)$

- In the classical problem, nice properties come from linearity w.r.t. data distribution
- In stochastic composition optimization, they are all lost

A little nonlinearity takes a long way to go.
Motivating Example: High-Dimensional Nonparametric Estimation

- Sparse Additive Model (SpAM):

\[ y_i = \sum_{j=1}^{d} h_j(x_{ij}) + \epsilon_i. \]

- High-dimensional feature space with relatively few data samples:

\[ \begin{array}{c|c|c}
\text{Features } d & \text{Sample size } n \\
\midrule
n \gg d & n \ll d
\end{array} \]

- Optimization model for SpAM\(^1\):

\[
\min_{\mathbf{x}} \mathbb{E}[f_{\mathbf{v}}(\mathbb{E}[\mathbf{g}_{\mathbf{w}}(\mathbf{x})])] \iff \min_{h_j \in \mathcal{H}_j} \mathbb{E}\left[ Y - \sum_{j=1}^{d} h_j(X_j) \right]^2 + \lambda \sum_{j=1}^{d} \sqrt{\mathbb{E}[h_j^2(X_j)]}
\]

- The term \( \lambda \sum_{j=1}^{d} \sqrt{\mathbb{E}[h_j^2(X_j)]} \) induces sparsity in the feature space.

---

Motivating Example: Risk-Averse Learning

Consider the mean-variance minimization problem

$$\min_x E_{a,b} \left[ \ell(x; a, b) \right] + \lambda \text{Var}_{a,b}[\ell(x; a, b)],$$

Its batch version is

$$\min_x \frac{1}{N} \sum_{i=1}^{N} \ell(x; a_i, b_i) + \frac{\lambda}{N} \sum_{i=1}^{N} \left( \ell(x; a_i, b_i) - \frac{1}{N} \sum_{i=1}^{N} \ell(x; a_i, b_i) \right)^2.$$

- The variance $\text{Var}[Z] = E \left[ (Z - E[Z])^2 \right]$ is a composition between two functions.
- Many other risk functions are equivalent to compositions of multiple expected-value functions (Shapiro, Dentcheva, Ruszcyński 2014).
- A central limit theorem for composite of multiple smooth functions has been established for risk metrics (Dentcheva, Penev, Ruszcyński 2016).
- No good way to optimize a risk-averse objective while learning from online data.
Motivating Example: Reinforcement Learning

On-policy reinforcement learning is to learn the value-per-state of a stochastic system.

- We want to solve a (huge) Bellman equations

\[ \gamma P^\pi V^\pi + r^\pi = V^\pi, \]

where \( P^\pi \) is transition prob. matrix and \( r^\pi \) are rewards, both unknown.

- On-policy learning aims to solve Bellman equation via blackbox simulation. It becomes a special stochastic composition optimization problem:

\[ \min_x E[f_v(E[g_w(x)])] \iff \min_{x \in \mathbb{R}^S} \| E[A]x - E[b] \|^2, \]

where \( E[A] = I - \gamma P^\pi \) and \( E[b] = r^\pi \).
Outline

1 Background: Why is SGD a good method?

2 A New Problem: Stochastic Composition Optimization

3 Stochastic Composition Algorithms: Convergence and Sample Complexity

4 Acceleration via Smoothing-Extrapolation
Problem Formulation

\[
\min_{x \in \mathcal{X}} \left\{ F(x) := \mathbb{E}[f_v(\mathbb{E}[g_w(x)])] \right\},
\]
nonlinear w.r.t the distribution of \((w,v)\)

Sampling Oracle (SO)

Upon query \((x, y)\), the oracle returns:

- Noisy inner sample \(g_w(x)\) and its noisy subgradient \(\tilde{\nabla} g_w(x)\);
- Noisy outer gradient \(\nabla f_v(y)\)

Challenges

- Stochastic gradient descent (SGD) method does not work since an “unbiased” sample of the gradient
  \[
  \tilde{\nabla} g(x_k) \nabla f(g(x_k))
  \]
  is not available.
- Fenchel dual does not work except for rare conditions
- Sample average approximation (SAA) subject to curse of dimensionality.
- Sample complexity unclear
Basic Idea

To approximate

\[ x_{k+1} = \Pi_X \{ x_k - \alpha_k \tilde{\nabla}g(x_k) \nabla f(g(x_k)) \} , \]

by a quasi-gradient iteration using estimates of \( g(x_k) \).

Algorithm 1: Stochastic Compositional Gradient Descent (SCGD)

Require: \( x_0, z_0 \in \mathbb{R}^n, y_0 \in \mathbb{R}^m, SO, K, \) stepsizes \( \{\alpha_k\}_{k=1}^K \), and \( \{\beta_k\}_{k=1}^K \).

Ensure: \( \{x_k\}_{k=1}^K \)

\hspace{1em} for \( k = 1, \cdots, K \) do

\hspace{1em} Query the \( SO \) and obtain \( \tilde{\nabla}g_{w_k}(x_k), g_{w_k}(x_k), f_{v_k}(y_{k+1}) \)

\hspace{1em} Update by

\[ y_{k+1} = (1 - \beta_k)y_k + \beta_k g_{w_k}(x_k), \]
\[ x_{k+1} = \Pi_X \{ x_k - \alpha_k \tilde{\nabla}g_{w_k}(x_k) \nabla f_{v_k}(y_{k+1}) \} , \]

end for

Remarks

- Each iteration makes simple updates by interacting with \( SO \)
- Scalable with large-scale batch data and can process streaming data points online
- Considered for the first time by (Ermoliev 1976) as a stochastic approximation method without rate analysis
Sample Complexity (Wang et al., 2016)
Under suitable conditions (inner function nonsmooth, outer function smooth), and \( \mathcal{X} \) is bounded, let the stepsizes be
\[
\alpha_k = k^{-3/4}, \quad \beta_k = k^{-1/2},
\]
we have that, if \( k \) is large enough,
\[
\mathbb{E} \left[ F \left( \frac{2}{k} \sum_{t=k/2+1}^{k} x_t \right) - F^* \right] = \mathcal{O} \left( \frac{1}{k^{1/4}} \right).
\]
(Optimal rate which matches the lowerbound for stochastic programming)

Sample Convexity in Strongly Convex Case (Wang et al., 2016)
Under suitable conditions (inner function nonsmooth, outer function smooth), suppose that the compositional function \( F(\cdot) \) is strongly convex, let the stepsizes be
\[
\alpha_k = \frac{1}{k}, \quad \text{and} \quad \beta_k = \frac{1}{k^{2/3}},
\]
we have, if \( k \) is sufficiently large,
\[
\mathbb{E} [\| x_k - x^* \|^2] = \mathcal{O} \left( \frac{1}{k^{2/3}} \right).
\]
Outline of Analysis

- The auxiliary variable $y_k$ is taking a running estimate of $g(x_k)$ at biased query points $x_0, \ldots, x_k$:

$$y_{k+1} = \sum_{t=0}^{k} (\prod_{t'=t}^{k} \beta_t') g_{w_t}(x_t).$$

- Two entangled stochastic sequences

$$\{\epsilon_k = \|y_{k+1} - g(x_k)\|^2\}, \quad \{\xi_k = \|x_k - x^*\|^2\}.$$

- Coupled supermartingale analysis

$$E[\epsilon_{k+1} | F_k] \leq (1 - \beta_k)\epsilon_k + O(\beta_k^2 + \frac{1}{\beta_k} \|x_{k+1} - x_k\|^2)$$

$$E[\xi_{k+1} | F_k] \leq (1 + \alpha_k^2)\xi_k - \alpha_k(F(x_k) - F^*) + O(\beta_k)\epsilon_k$$

- Almost sure convergence by using a Coupled Supermartingale Convergence Theorem (Wang and Bertsekas, 2013)

- Convergence rate analysis via optimizing over stepsizes and balancing noise-bias tradeoff
Outline

1. Background: Why is SGD a good method?
2. A New Problem: Stochastic Composition Optimization
3. Stochastic Composition Algorithms: Convergence and Sample Complexity
4. Acceleration via Smoothing-Extrapolation
**Acceleration**

When the function \( g(\cdot) \) is smooth, can the algorithms be accelerated? Yes!

---

**Algorithm 2: Accelerated SCGD**

**Require:** \( x_0, z_0 \in \mathbb{R}^n, y_0 \in \mathbb{R}^m, S\mathcal{O}, K, \) stepsizes \( \{\alpha_k\}_{k=1}^K \), and \( \{\beta_k\}_{k=1}^K \).

**Ensure:** \( \{x_k\}_{k=1}^K \)

for \( k = 1, \ldots, K \) do

Query the \( S\mathcal{O} \) and obtain \( \nabla f_{v_k}(y_k), \nabla g_{w_k}(z_k) \).

Update by

\[
x_{k+1} = x_k - \alpha_k \nabla g_{w_k}^\top(x_k) \nabla f_{v_k}(y_k).
\]

Update auxiliary variables via **extrapolation-smoothing**:

\[
z_{k+1} = \left(1 - \frac{1}{\beta_k}\right) x_k + \frac{1}{\beta_k} x_{k+1},
\]

\[
y_{k+1} = (1 - \beta_k)y_k + \beta_k g_{w_{k+1}}(z_{k+1}),
\]

where the sample \( g_{w_{k+1}}(z_{k+1}) \) is obtained via querying the \( S\mathcal{O} \).

end for

---

**Key to the Acceleration**

Bias reduction by averaging over extrapolated points (**extrapolation-smoothing**)

\[
y_k = \sum_{t=0}^{k} (\prod_{t'=t}^{k} \beta_{t'}) g_{w_t}(z_t) \approx g(x_k) = g\left(\sum_{t=0}^{k} (\prod_{t'=t}^{k} \beta_{t'}) z_t\right).
\]
Accelerated Sample Complexity (Wang et al. 2016)
Under suitable conditions (inner function smooth, outer function smooth), if the stepsizes are chosen as
\[
\alpha_k = k^{-5/7}, \quad \beta_k = k^{-4/7},
\]
we have,
\[
\mathbb{E} \left[ F(\hat{x}_k) - F^* \right] = \mathcal{O}\left( \frac{1}{k^{2/7}} \right),
\]
where \( \hat{x}_k = \frac{2}{k} \sum_{t=k/2+1}^{k} x_t \).

Strongly Convex Case (Wang et al. 2016)
Under suitable conditions (inner function smooth, outer function smooth), and we assume that \( F \) is strongly-convex. If the stepsizes are chosen as
\[
\alpha_k = \frac{1}{k}, \quad \text{and} \quad \beta_k = \frac{1}{k^{4/5}},
\]
we have,
\[
\mathbb{E}[\|x_k - x^*\|^2] = \mathcal{O}\left( \frac{1}{k^{4/5}} \right).
\]
Regularized Stochastic Composition Optimization (Wang and Liu 2016)

\[
\min_{x \in \mathbb{R}^n} \left( E_v f_v \circ E_w g_w \right)(x) + R(x)
\]

The penalty term \( R(x) \) is convex and nonsmooth.

Algorithm 3: Accelerated Stochastic Compositional Proximal Gradient (ASC-PG)

Require: \( x_0, z_0 \in \mathbb{R}^n, y_0 \in \mathbb{R}^m, SO, K, \) stepsizes \( \{\alpha_k\}_{k=1}^K \), and \( \{\beta_k\}_{k=1}^K \).

Ensure: \( \{x_k\}_{k=1}^K \)

for \( k = 1, \ldots, K \) do

Query the \( SO \) and obtain \( \nabla f_{v_k}(y_k), \nabla g_{w_k}(z_k) \).

Update the main iterate by

\[
x_{k+1} = \text{prox}_{\alpha_k R} \left( x_k - \alpha_k \nabla g_{w_k}^T(x_k) \nabla f_{v_k}(y_k) \right).
\]

Update auxiliary iterates by an extrapolation-smoothing scheme:

\[
z_{k+1} = \left( 1 - \frac{1}{\beta_k} \right) x_k + \frac{1}{\beta_k} x_{k+1},
\]

\[
y_{k+1} = (1 - \beta_k) y_k + \beta_k g_{w_{k+1}}(z_{k+1}),
\]

where the sample \( g_{w_{k+1}}(z_{k+1}) \) is obtained via querying the \( SO \).

end for
Sample Complexity for Smooth Optimization (Wang and Liu 2016)

Under suitable conditions (inner function nonsmooth, outer function smooth), and $\mathcal{X}$ is bounded, let the stepsizes be chosen properly, we have that, if $k$ is large enough,

$$
E \left[ F \left( \frac{2}{k} \sum_{t=k/2+1}^{k} x_t \right) - F^* \right] = O \left( \frac{1}{k^{4/9}} \right),
$$

If either the outer or the inner function is linear, the rate improves to be optimal:

$$
E \left[ F \left( \frac{2}{k} \sum_{t=k/2+1}^{k} x_t \right) - F^* \right] = O \left( \frac{1}{k^{1/2}} \right),
$$

Sample Convexity in Strongly Convex Case (Wang and Liu 2016)

Suppose that the compositional function $F(\cdot)$ is strongly convex, let the stepsizes be chosen properly, we have, if $k$ is sufficiently large,

$$
E \left[ \|x_k - x^*\|^2 \right] = O \left( \frac{1}{k^{4/5}} \right).
$$

If either the outer or the inner function is linear, the rate improves to be optimal:

$$
E \left[ \|x_k - x^*\|^2 \right] = O \left( \frac{1}{k} \right).$$
When there are two nested uncertainties:

- A class of two-timescale algorithms that update using first-order samples
- Analyzing convergence becomes harder: two coupled stochastic processes and smoothness-noise interplay
- Convergence rates of stochastic algorithms establish sample complexity upper bounds for the new problem class

<table>
<thead>
<tr>
<th>Outer Nonsmooth, Inner Smooth</th>
<th>General Convex</th>
<th>Strongly Convex</th>
</tr>
</thead>
<tbody>
<tr>
<td>O(k⁻¹/₄)</td>
<td>O(k⁻²/₃)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outer and Inner Smooth</th>
<th>General Convex</th>
<th>Strongly Convex</th>
</tr>
</thead>
<tbody>
<tr>
<td>O(k⁻⁴/₉)</td>
<td>O(k⁻⁴/₅)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Special Case: minₓ E[f(x; ξ)]</th>
<th>General Convex</th>
<th>Strongly Convex</th>
</tr>
</thead>
<tbody>
<tr>
<td>O(k⁻¹/₂)</td>
<td>O(k⁻¹)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Special Case: minₓ E[f(E[A]x − E[b]; ξ)]</th>
<th>General Convex</th>
<th>Strongly Convex</th>
</tr>
</thead>
<tbody>
<tr>
<td>O(k⁻¹/₂)</td>
<td>O(k⁻¹)</td>
<td></td>
</tr>
</tbody>
</table>

**Table**: Summary of best known sample complexities

Applications and computations

- First scalable algorithm for sparse nonparametric estimation
- Optimal algorithm for on-policy reinforcement learning
Summary

• Stochastic Composition Optimization - a new and rich problem class

\[ \min_x \mathbb{E}[f_v(\mathbb{E}[g_w(x)])] \]

nonlinear w.r.t the distribution of \((w,v)\)

• Applications in risk, data analysis, machine learning, real-time intelligent systems

• A class of stochastic compositional gradient methods with convergence guarantees. Basic sample complexity being developed:

<table>
<thead>
<tr>
<th></th>
<th>General/Convex</th>
<th>Strongly Convex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outer Nonsmooth, Inner Smooth</td>
<td>(O\left(k^{-1/4}\right))</td>
<td>(O\left(k^{-2/3}\right))</td>
</tr>
<tr>
<td>Outer and Inner Smooth</td>
<td>(O\left(k^{-4/9}\right))</td>
<td>(O\left(k^{-4/5}\right))</td>
</tr>
<tr>
<td>Special Case: (\min_x \mathbb{E}[f(x; \xi)])</td>
<td>(O\left(k^{-1/2}\right))</td>
<td>(O\left(k^{-1}\right))</td>
</tr>
<tr>
<td>Special Case: (\min_x \mathbb{E}[f(\mathbb{E}[A</td>
<td>x] - E[b]; \xi)])</td>
<td>(O\left(k^{-1/2}\right))</td>
</tr>
</tbody>
</table>

• Many open questions remaining and more works needed!

Thank you very much!