Optimization for Machine Learning

Stochastic Gradient Methods

Lecturer: Robert M. Gower

Master IASD: AI Systems and Data Science, 2019
Core Info

• **Where:** ENS: 07/11 amphi Langevin, 03/12 U209, 05/12 amphi Langevin.

• **Online:** Teaching materials for these 3 classes:  
  https://gowerrobert.github.io/

• **Google docs with course info:** Can also be found on  
  https://gowerrobert.github.io/
Outline of my three classes

• 07/11/19  Foundations and the empirical risk problem, revision probability, SGD (Stochastic Gradient Descent) for ridge regression

• 03/12/19  (TODAY) SGD for convex optimization. Theory, variants including averaging, decreasing stepsizes and momentum.

• 05/12/19  Lab on SGD and variants  BRING LAPTOPS!
Solving the Finite Sum Training Problem
Recap

Training Problem

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( h_w(x^i), y^i \right) + \lambda R(w) =: f(w)
\]

\[L(w) = \text{loss}\]

General methods

\[\min f(w)\]

Two parts

\[\min L(w) + \lambda R(w)\]

- Gradient Descent
- Proximal gradient (ISTA)
- Fast proximal gradient (FISTA)
A Datum Function

$$f_i(w) := \ell \left( h_w(x^i), y^i \right) + \lambda R(w)$$

$$\frac{1}{n} \sum_{i=1}^{n} \ell \left( h_w(x^i), y^i \right) + \lambda R(w) = \frac{1}{n} \sum_{i=1}^{n} \left( \ell \left( h_w(x^i), y^i \right) + \lambda R(w) \right)$$

$$= \frac{1}{n} \sum_{i=1}^{n} f_i(w)$$

Finite Sum Training Problem

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f_i(w) =: f(w)$$

Can we use this sum structure?
The Training Problem

Solving the training problem:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f_i(w)$$

Reference method: Gradient descent

$$\nabla \left( \frac{1}{n} \sum_{i=1}^{n} f_i(w) \right) = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w)$$

Gradient Descent Algorithm

Set $w^0 = 0$, choose $\alpha > 0$.

for $t = 0, 1, 2, \ldots, T - 1$

$$w^{t+1} = w^t - \frac{\alpha}{n} \sum_{i=1}^{n} \nabla f_i(w^t)$$

Output $w^T$
The Training Problem

Solving the training problem:

\[
\min_{\mathbf{w} \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f_i(\mathbf{w})
\]

Problem with Gradient Descent:
Each iteration requires computing a gradient \( \nabla f_i(\mathbf{w}) \) for each data point. One gradient for each cat on the internet!

Gradient Descent Algorithm

Set \( \mathbf{w}^0 = 0 \), choose \( \alpha > 0 \).

for \( t = 0, 1, 2, \ldots, T \)

\[
\mathbf{w}^{t+1} = \mathbf{w}^t - \frac{\alpha}{n} \sum_{i=1}^{n} \nabla f_i(\mathbf{w}^t)
\]

Output \( \mathbf{w}^T \)
Stochastic Gradient Descent

Is it possible to design a method that uses only the gradient of a **single** data function $f_i(w)$ at each iteration?
Is it possible to design a method that uses only the gradient of a **single** data function $f_i(w)$ at each iteration?

**Unbiased Estimate**

Let $j$ be a random index sampled from \{1, ..., $n$\} selected uniformly at random. Then

\[
\mathbb{E}_j[\nabla f_j(w)] = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w) = \nabla f(w)
\]
Stochastic Gradient Descent

Is it possible to design a method that uses only the gradient of a single data function \( f_i(w) \) at each iteration?

**Unbiased Estimate**

Let \( j \) be a random index sampled from \( \{1, \ldots, n\} \) selected uniformly at random. Then

\[
\mathbb{E}_j[\nabla f_j(w)] = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w) = \nabla f(w)
\]

Use \( \nabla f_j(w) \approx \nabla f(w) \)
Stochastic Gradient Descent

Is it possible to design a method that uses only the gradient of a single data function $f_i(w)$ at each iteration?

**Unbiased Estimate**

Let $j$ be a random index sampled from $\{1, \ldots, n\}$ selected uniformly at random. Then

$$E_j[\nabla f_j(w)] = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w) = \nabla f(w)$$

Use $\nabla f_j(w) \approx \nabla f(w)$

**EXE:**

Let $\sum_{i=1}^{n} p_i = 1$ and $j \sim p_j$. Show $E[\nabla f_j(w)/(np_j)] = \nabla f(w)$
Stochastic Gradient Descent

**SGD 0.0 Constant stepsizes**

- Set $w^0 = 0$, choose $\alpha > 0$
- For $t = 0, 1, 2, \ldots, T - 1$
  - Sample $j \in \{1, \ldots, n\}$
  - $w^{t+1} = w^t - \alpha \nabla f_j(w^t)$
- Output $w^T$
Stochastic Gradient Descent
Assumptions for Convergence

**Strong Convexity**

\[ f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} \| y - w \|^2_2, \quad \forall w, y \]

\[ y = w^* \]

\[ 2\langle \nabla f(w), w - w^* \rangle \geq \lambda \| w - w^* \|^2_2 \]
Assumptions for Convergence

**Strong Convexity**

\[
f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} \|y - w\|_2^2, \quad \forall w, y
\]

\[
y = w^*
\]

\[
2\langle \nabla f(w), w - w^* \rangle \geq \lambda \|w - w^*\|_2^2
\]
Assumptions for Convergence

Strong Convexity

\[ f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} \|y - w\|_2^2, \quad \forall w, y \]

\[ y = w^* \]

\[ 2\langle \nabla f(w), w - w^* \rangle \geq \lambda \|w - w^*\|_2^2 \]

Expected Bounded Stochastic Gradients

\[ \mathbb{E}_j[\|\nabla f_j(w^t)\|_2^2] \leq B^2, \text{ for all iterates } w^t \text{ of SGD} \]
Assumptions for Convergence

**Strong Convexity**

\[ f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} ||y - w||^2_2, \quad \forall w, y \]

\[ y = w^* \]

\[ 2\langle \nabla f(w), w - w^* \rangle \geq \lambda ||w - w^*||^2_2 \]

**Expected Bounded Stochastic Gradients**

\[ E_j[||\nabla f_j(w^t)||^2_2] \leq B^2, \text{ for all iterates } w^t \text{ of SGD} \]
Complexity / Convergence

Theorem

If $0 < \alpha \leq \frac{1}{\lambda}$ then the iterates of the SGD 0.0 method satisfy

$$
\mathbb{E} \left[ ||w^t - w^*||_2^2 \right] \leq (1 - \alpha \lambda)^t ||w^0 - w^*||_2^2 + \frac{\alpha}{\lambda} B^2
$$

**EXE:** Do exercises on convergence of random sequences.
Complexity / Convergence

**Theorem**

If $0 < \alpha \leq \frac{1}{\lambda}$ then the iterates of the SGD 0.0 method satisfy

$$
\mathbb{E} \left[ \|w^t - w^*\|_2^2 \right] \leq (1 - \alpha \lambda)^t \|w^0 - w^*\|_2^2 + \frac{\alpha}{\lambda} B^2
$$

Shows that $\alpha \approx \frac{1}{\lambda}$

**EXE:** Do exercises on convergence of random sequences.
Complexity / Convergence

Theorem

If $0 < \alpha \leq \frac{1}{\lambda}$ then the iterates of the SGD 0.0 method satisfy

$$\mathbb{E} [||w^t - w^*||_2^2] \leq (1 - \alpha \lambda)^t ||w^0 - w^*||_2^2 + \frac{\alpha}{\lambda} B^2$$

Shows that $\alpha \approx \frac{1}{\lambda}$

Shows that $\alpha \approx 0$

**EXE:** Do exercises on convergence of random sequences.
Proof:

\[ \|w^{t+1} - w^*\|_2^2 = \|w^t - w^* - \alpha \nabla f_j(w^t)\|_2^2 \]

\[ = \|w^t - w^*\|_2^2 - 2\alpha \langle \nabla f_j(w^t), w^t - w^* \rangle + \alpha^2 \|\nabla f_j(w^t)\|_2^2. \]

Taking expectation with respect to \( j \)

\[ \mathbb{E}_j \left[ \|w^{t+1} - w^*\|_2^2 \right] = \|w^t - w^*\|_2^2 - 2\alpha \langle \nabla f(w^t), w^t - w^* \rangle + \alpha^2 \mathbb{E}_j \left[ \|\nabla f_j(w^t)\|_2^2 \right] \]

Strong conv.

\[ \leq \|w^t - w^*\|_2^2 - 2\alpha \langle \nabla f(w^t), w^t - w^* \rangle + \alpha^2 B^2 \]

Taking total expectation

\[ \mathbb{E} \left[ \|w^{t+1} - w^*\|_2^2 \right] \leq (1 - \alpha \lambda) \|w^t - w^*\|_2^2 + \alpha^2 B^2 \]

Using the geometric series sum

\[ \sum_{i=0}^{t} (1 - \alpha \lambda)^i = \frac{1 - (1 - \alpha \lambda)^{t+1}}{\alpha \lambda} \leq \frac{1}{\alpha \lambda} \]

\[ \mathbb{E} \left[ \|w^{t+1} - w^*\|_2^2 \right] \leq (1 - \alpha \lambda)^{t+1} \|w^0 - w^*\|_2^2 + \frac{\alpha}{\lambda} B^2 \]
Stochastic Gradient Descent
\( \alpha = 0.01 \)
Stochastic Gradient Descent
$\alpha = 0.1$
Stochastic Gradient Descent

$\alpha = 0.2$
Stochastic Gradient Descent

$\alpha = 0.5$
**Assumptions for Convergence**

**Strong Convexity**

\[ f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} \|y - w\|^2_2, \quad \forall w, y \]

\[ y = w^* \]

\[ 2\langle \nabla f(w), w - w^* \rangle \geq \lambda \|w - w^*\|^2_2 \]

**Expected Bounded Stochastic Gradients**

\[ \mathbb{E}_j[\|\nabla f_j(w^t)\|^2_2] \leq B^2, \text{ for all iterates } w^t \text{ of SGD} \]
Assumptions for Convergence

**Strong Convexity**

\[ f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} \| y - w \|_2^2, \quad \forall w, y \]

\[ y = w^* \]

\[ 2\langle \nabla f(w), w - w^* \rangle \geq \lambda \| w - w^* \|_2^2 \]

**Expected Bounded Stochastic Gradients**

\[ \mathbb{E}_j[\| \nabla f_j(w^t) \|_2^2] \leq B^2, \text{ for all iterates } w^t \text{ of SGD} \]
**Assumptions for Convergence**

**Strong Convexity**

\[
f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} ||y - w||^2_2, \quad \forall w, y
\]

\[
y = w^*
\]

\[
2\langle \nabla f(w), w - w^* \rangle \geq \lambda ||w - w^*||^2_2
\]

**Expected Bounded Stochastic Gradients**

\[
\mathbb{E}_j[||\nabla f_j(w^t)||^2_2] \leq B^2, \text{ for all iterates } w^t \text{ of SGD}
\]
Assumptions for Convergence

Strong Convexity

\[ f(y) \geq f(w) + \langle \nabla f(w), y - w \rangle + \frac{\lambda}{2} \| y - w \|^2_2, \quad \forall w, y \]

\[ y = w^* \]

\[ 2\langle \nabla f(w), w - w^* \rangle \geq \lambda \| w - w^* \|^2_2 \]

Expected Bounded Stochastic Gradients

\[ \mathbb{E}_{j}[\| \nabla f_j(w^t) \|_2^2] \leq B^2, \text{ for all iterates } w^t \text{ of SGD} \]

**EXE:**

Let \( A \in \mathbb{R}^{n \times d} \), \( f_j(w) = (A_j : w - b_j)^2 \). \[ \max_w \mathbb{E}_{j \sim \frac{1}{n}}[\| \nabla f_j(w) \|^2] = ? \]
**EXE:**
Let \( A \in \mathbb{R}^{n \times d} \), \( f_j(w) = (A_j:w - b_j)^2 \). \( \max_w \mathbb{E}_{j \sim \frac{1}{n}}[\|\nabla f_j(w)\|^2] = ? \)

**Proof:** \( \max_w \mathbb{E}_{j \sim \frac{1}{n}}[\|\nabla f_j(w)\|^2] = \infty \), indeed since

\[
\|\nabla f_j(w)\|^2 = 4\|A_j^\top (A_j:w - b_j)\|^2
\]

\[
= 4\|A_j:w\|^2 (A_j:w - b_j)^2
\]

\[
= 4(\hat{A}_j:w - \hat{b}_j)^2 \quad \text{where} \quad \hat{A}_j := A_j:\|A_j:\|, \quad \hat{b}_j := b_j:\|A_j:\|
\]

Taking expectation

\[
\mathbb{E}_{j \sim \frac{1}{n}} \|\nabla f_j(w)\|^2 = \frac{1}{n} \sum_{j=1}^{n} 4(\hat{A}_j:w - \hat{b}_j)^2 = \frac{1}{n}\|\hat{A}w - \hat{b}\|^2
\]

\[
\lim_{w \to \infty} \|\hat{A}w - b\|^2 = \infty
\]
Realistic assumptions for Convergence

Strongly quasi-convexity

\[ f(w^*) \geq f(w) + \langle \nabla f(w), w^* - w \rangle + \frac{\mu}{2} ||w^* - w||^2_2, \quad \forall w \]

Each \( f_i \) is convex and \( L_i \) smooth

\[ f_i(y) \leq f_i(w) + \langle \nabla f_i(w), y - w \rangle + \frac{L_i}{2} ||y - w||^2_2, \quad \forall w \]

\[ L_{\text{max}} := \max_{i=1,...,n} L_i \]

Definition: Gradient Noise

\[ \sigma^2 := \mathbb{E}_j[||\nabla f_j(w^*)||^2_2] \]
Assumptions for Convergence

**EXE:** Calculate the $L_i$'s and $L_{max}$ for

1. $f(w) = \frac{1}{2n} ||Aw - y||_2^2 + \frac{\lambda}{2} ||w||_2^2$

**HINT:** A twice differentiable $f_i$ is $L_i$-smooth if and only if

$$\nabla^2 f_i(w) \preceq L_i I \iff v^T \nabla^2 f_i(w)v \leq L_i ||v||^2, \forall v$$
Assumptions for Convergence

**EXE:** Calculate the $L_i$'s and $L_{\text{max}}$ for

1. $f(w) = \frac{1}{2n} ||Aw - y||_2^2 + \frac{\lambda}{2} ||w||_2^2$

**HINT:** A twice differentiable $f_i$ is $L_i$ - smooth if and only if

$$\nabla^2 f_i(w) \leq L_i I \iff v^T \nabla^2 f_i(w)v \leq L_i ||v||^2, \forall v$$

1. $f(w) = \frac{1}{2n} ||Aw - y||_2^2 + \frac{\lambda}{2} ||w||_2^2 = \frac{1}{n} \sum_{i=1}^{n} \left( \frac{1}{2} (A_i^T w - y_i)^2 + \frac{\lambda}{2} ||w||_2^2 \right) = \frac{1}{n} \sum_{i=1}^{n} f_i(w)$
Assumptions for Convergence

**EXE:** Calculate the $L_i$’s and $L_{\text{max}}$ for

$$f(w) = \frac{1}{2n} \| Aw - y \|_2^2 + \frac{\lambda}{2} \| w \|_2^2$$

**HINT:** A twice differentiable $f_i$ is $L_i$-smooth if and only if

$$\nabla^2 f_i(w) \preceq L_i I \iff v^\top \nabla^2 f_i(w) v \leq L_i \| v \|_2^2, \forall v$$

1. $f(w) = \frac{1}{2n} \| Aw - y \|_2^2 + \frac{\lambda}{2} \| w \|_2^2 = \frac{1}{n} \sum_{i=1}^n \left( \frac{1}{2} (A_i^\top w - y_i)^2 + \frac{\lambda}{2} \| w \|_2^2 \right)$

   $$= \frac{1}{n} \sum_{i=1}^n f_i(w)$$

   $$\nabla^2 f_i(w) = A_i A_i^\top + \lambda \preceq (\| A_i \|_2^2 + \lambda) I = L_i I$$
Assumptions for Convergence

**EXE:** Calculate the $L_i$’s and $L_{\text{max}}$ for

1. \[ f(w) = \frac{1}{2n} \|Aw - y\|_2^2 + \frac{\lambda}{2} \|w\|_2^2 \]

**HINT:** A twice differentiable $f_i$ is $L_i$-smooth if and only if

\[ \nabla^2 f_i(w) \preceq L_i I \iff v^\top \nabla^2 f_i(w) v \leq L_i \|v\|^2, \forall v \]

1. \[ f(w) = \frac{1}{2n} \|Aw - y\|_2^2 + \frac{\lambda}{2} \|w\|_2^2 = \frac{1}{n} \sum_{i=1}^{n} \left( \frac{1}{2}(A_i^\top w - y_i)^2 + \frac{\lambda}{2} \|w\|_2^2 \right) \]

\[ = \frac{1}{n} \sum_{i=1}^{n} f_i(w) \]

\[ \nabla^2 f_i(w) = A_i A_i^\top + \lambda \preceq (\|A_i\|_2^2 + \lambda)I = L_i I \]

\[ L_{\text{max}} = \max_{i=1, \ldots, n} (\|A_i\|_2^2 + \lambda) = \max_{i=1, \ldots, n} ||A_i||_2^2 + \lambda \]
Assumptions for Convergence

**EXE:** Calculate the $L_i$'s and $L_{\text{max}}$ for

$$f(w) = \frac{1}{n} \sum_{i=1}^{n} \ln(1 + e^{-y_i \langle w, a_i \rangle}) + \frac{\lambda}{2} \|w\|_2^2$$
Assumptions for Convergence

**EXE:** Calculate the $L_i$’s and $L_{\text{max}}$ for

\begin{align*}
2. \quad f(w) &= \frac{1}{n} \sum_{i=1}^{n} \ln(1 + e^{-y_i \langle w, a_i \rangle}) + \frac{\lambda}{2} \|w\|_2^2
\end{align*}

\begin{align*}
2. \quad f_i(w) &= \ln(1 + e^{-y_i \langle w, a_i \rangle}) + \frac{\lambda}{2} \|w\|_2^2,
\end{align*}
Assumptions for Convergence

**EXE:** Calculate the $L_i$’s and $L_{\text{max}}$ for

2. $f(w) = \frac{1}{n} \sum_{i=1}^{n} \ln(1 + e^{-y_i \langle w, a_i \rangle}) + \frac{\lambda}{2} \|w\|_2^2$

2. $f_i(w) = \ln(1 + e^{-y_i \langle w, a_i \rangle}) + \frac{\lambda}{2} \|w\|_2^2$

$$\nabla f_i(w) = \frac{-y_i a_i e^{-y_i \langle w, a_i \rangle}}{1 + e^{-y_i \langle w, a_i \rangle}} + \lambda w$$

$$\nabla^2 f_i(w) = a_i a_i^\top \left( \frac{(1 + e^{-y_i \langle w, a_i \rangle}) e^{-y_i \langle w, a_i \rangle}}{(1 + e^{-y_i \langle w, a_i \rangle})^2} \right) - \frac{e^{-2y_i \langle w, a_i \rangle}}{(1 + e^{-y_i \langle w, a_i \rangle})^2} + \lambda I$$

$$= a_i a_i^\top \frac{e^{-y_i \langle w, a_i \rangle}}{(1 + e^{-y_i \langle w, a_i \rangle})^2} + \lambda I \preceq \left( \frac{\|a_i\|_2^2}{4} + \lambda \right) I = L_i I$$
Relationship between smoothness constants

**EXE:** Let \( f \) be differentiable and convex. Show that \( f(w) \) is \( L \)-smooth with

\[
L = \max_{w \in \mathbb{R}^d} \lambda_{\max}(\nabla^2 f(w))
\]

Thus \( f_i(w) \) is \( L_i \)-smooth with \( L_i = \max_{w \in \mathbb{R}^d} \lambda_{\max}(\nabla^2 f_i(w)) \) show that

\[
L \leq \frac{1}{n} \sum_{i=1}^{n} L_i \leq L_{\max} := \max_{i=1,\ldots,n} L_i
\]
Relationship between smoothness constants

**EXE:** Let $f$ be differentiable and convex. Show that $f(w)$ is $L$–smooth with

$$L = \max_{w \in \mathbb{R}^d} \lambda_{\max}(\nabla^2 f(w))$$

Thus $f_i(w)$ is $L_i$–smooth with $L_i = \max_{w \in \mathbb{R}^d} \lambda_{\max}(\nabla^2 f_i(w))$ show that

$$L \leq \frac{1}{n} \sum_{i=1}^{n} L_i \leq L_{\max} := \max_{i=1,\ldots,n} L_i$$

**Proof:** From the Hessian definition of smoothness

$$\nabla^2 f(w) \preceq \lambda_{\max}(\nabla^2 f(w)) I \preceq \max_{w \in \mathbb{R}^d} \lambda_{\max}(\nabla^2 f(w)) I$$

Furthermore

$$\lambda_{\max}(\nabla^2 f(w)) = \lambda_{\max}\left(\frac{1}{n} \sum_{i=1}^{n} \nabla^2 f_i(w)\right) \leq \frac{1}{n} \sum_{i=1}^{n} \lambda_{\max}(\nabla^2 f_i(w)) \leq \frac{1}{n} \sum_{i=1}^{n} L_i$$

Which follows since the largest eigenvalue function is convex over psd matrices. Now take the max over $w$, then max over $i$. 
Complexity / Convergence

Theorem.
Let $f$ be $\mu$–strongly quasi-convex and $f_i$ be $L_i$–smooth. If $0 < \alpha \leq \frac{1}{2L_{\text{max}}}$ then the iterates of the SGD 0.0 satisfy

$$\mathbb{E} \left[||w^t - w^*||_2^2\right] \leq (1 - \alpha \mu)^t ||w^0 - w^*||_2^2 + \frac{2\alpha}{\mu} \sigma^2$$

**EXE:** The steps of the proof are given in the SGD_proof exercise list for homework!
Stochastic Gradient Descent
\[ \alpha = 0.5 \]
Stochastic Gradient Descent

$\alpha = 0.5$

1) Start with big steps and end with smaller steps
Stochastic Gradient Descent
\[ \alpha = 0.5 \]

1) Start with big steps and end with smaller steps

2) Try averaging the points
SGD 1.0: Decreasing stepsize

Set $w^0 = 0$

Choose $\alpha_t > 0$, $\alpha_t \to 0$, $\sum_{t=0}^{\infty} \alpha_t = \infty$

for $t = 0, 1, 2, \ldots, T - 1$

sample $j \in \{1, \ldots, n\}$

$w^{t+1} = w^t - \alpha_t \nabla f_j(w^t)$

Output $w^T$
SGD shrinking stepsizes

**SGD 1.0: Decreasing stepsize**

Set $w^0 = 0$

Choose $\alpha_t > 0$, $\alpha_t \to 0$, $\sum_{t=0}^{\infty} \alpha_t = \infty$

for $t = 0, 1, 2, \ldots, T - 1$

sample $j \in \{1, \ldots, n\}$

$$w^{t+1} = w^t - \alpha_t \nabla f_j(w^t)$$

Output $w^T$

---

- How should we sample $j$?
- How fast $\alpha_t \to 0$?
- Does this converge?
SGD with shrinking stepsize
Compared with Gradient Descent
SGD with shrinking stepsize
Compared with Gradient Descent

Convergence plot

- Gradient Descent
- SGD 1.0
Complexity / Convergence

Theorem for shrinking stepsizes
Let $f$ be $\mu$–strongly quasi-convex and $f_i$ be $L_i$–smooth. Let $\mathcal{K} := L_{\text{max}} / \mu$ and let

$$\alpha^t = \begin{cases} \frac{1}{2L_{\text{max}}} & \text{for } t \leq 4[\mathcal{K}] \\ \frac{2t+1}{(t+1)^2 \mu} & \text{for } t > 4[\mathcal{K}] \end{cases}$$

If $t \geq 4[\mathcal{K}]$, then SGD 1.0 satisfies

$$\mathbb{E}\|w^t - w^*\|^2 \leq \frac{\sigma^2}{\mu^2} \frac{8}{t} + \frac{16}{\epsilon^2} \frac{[\mathcal{K}]^2}{t^2} \|w^0 - w^*\|^2$$

Iteration complexity $O\left(\frac{1}{t}\right)$
**Theorem for shrinking stepsizes**

Let $f$ be $\mu$–strongly quasi-convex and $f_i$ be $L_i$–smooth. Let $\mathcal{K} := \frac{L_{\text{max}}}{\mu}$ and let

\[
\alpha^t = \begin{cases} 
\frac{1}{2L_{\text{max}}} & \text{for } t \leq 4[\mathcal{K}] \\
\frac{2t+1}{(t+1)^2 \mu} & \text{for } t > 4[\mathcal{K}].
\end{cases}
\]

If $t \geq 4[\mathcal{K}]$, then SGD 1.0 satisfies

\[
\mathbb{E}\|w^t - w^*\|^2 \leq \frac{\sigma^2}{\mu^2} \frac{8}{t} + \frac{16}{\epsilon^2} \frac{[\mathcal{K}]^2}{t^2} \|w^0 - w^*\|^2
\]

Iteration complexity $O\left(\frac{1}{\epsilon}\right)$
Complexity / Convergence

Theorem for shrinking stepsizes
Let $f$ be $\mu$–strongly quasi-convex and $f_i$ be $L_i$–smooth. Let $\mathcal{K} := L_{\max}/\mu$ and let

$$\alpha^t = \begin{cases} \frac{1}{2L_{\max}} & \text{for } t \leq 4[\mathcal{K}] \\ \frac{2t+1}{(t+1)^2\mu} & \text{for } t > 4[\mathcal{K}]. \end{cases}$$

If $t \geq 4[\mathcal{K}]$, then SGD 1.0 satisfies

$$\mathbb{E}||w^t - w^*||^2 \leq \frac{\sigma^2}{\mu^2} \frac{8}{t} + \frac{16 [\mathcal{K}]^2}{e^2 t^2} ||w^0 - w^*||^2$$

$$O\left(\frac{1}{t}\right)$$

Iteration complexity $O\left(\frac{1}{\epsilon}\right)$

In practice $\alpha^t = C/(t + 1)$ or $\alpha^t = C/\sqrt{t + 1}$ where $C$ is tuned
Stochastic Gradient Descent Compared with Gradient Descent

Noisy iterates. Take averages?

SGD 1.0

Gradient Descent

Convergence plot

Loss function

# SGD iterations
SGD with (late start) averaging

**SGDA 1.1**

Set $w^0 = 0$

Choose $\alpha_t > 0$, $\alpha_t \to 0$, $\sum_{t=0}^{\infty} \alpha_t = \infty$

Choose averaging start $s_0 \in \mathbb{N}$

for $t = 0, 1, 2, \ldots, T - 1$

sample $j \in \{1, \ldots, n\}$

$w^{t+1} = w^t - \alpha_t \nabla f_j(w^t)$

if $t > s_0$

$$\bar{w} = \frac{1}{t-s_0} \sum_{i=s_0}^{t} w^t$$

else: $\bar{w} = w$

Output $\bar{w}$


*Acceleration of stochastic approximation by averaging*
SGD with (late start) averaging

**SGDA 1.1**

Set $w^0 = 0$

Choose $\alpha_t > 0$, $\alpha_t \to 0$, $\sum_{t=0}^{\infty} \alpha_t = \infty$

Choose averaging start $s_0 \in \mathbb{N}$

for $t = 0, 1, 2, \ldots, T - 1$

- sample $j \in \{1, \ldots, n\}$
- $w^{t+1} = w^t - \alpha_t \nabla f_j(w^t)$

if $t > s_0$

$$\overline{w} = \frac{1}{t-s_0} \sum_{i=s_0}^{t} w^t$$

else: $\overline{w} = w$

Output $\overline{w}$

---


*Acceleration of stochastic approximation by averaging*
Stochastic Gradient Descent
With and without averaging

Convergence plot

Loss function

#iterations

Starts slow, but can reach higher accuracy
Stochastic Gradient Descent
With and without averaging

Starts slow, but can reach higher accuracy

Only use averaging towards the end?
Stochastic Gradient Descent
Averaging the last few iterates
Comparison GD and SGD for strongly convex

| Iteration complexity | SGD \( O \left( \frac{1}{\epsilon} \right) \) | GD \( O \left( \log \left( \frac{1}{\epsilon} \right) \right) \) |
Comparison GD and SGD for strongly convex

<table>
<thead>
<tr>
<th>Iteration complexity</th>
<th>SGD</th>
<th>GD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$O\left(\frac{1}{\varepsilon}\right)$</td>
<td>$O\left(\log\left(\frac{1}{\varepsilon}\right)\right)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cost of an iteration</th>
<th>SGD</th>
<th>GD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$O\left(1\right)$</td>
<td>$O\left(n\right)$</td>
</tr>
</tbody>
</table>
Comparison GD and SGD for strongly convex

<table>
<thead>
<tr>
<th></th>
<th>SGD</th>
<th>GD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration complexity</td>
<td>$O\left(\frac{1}{\epsilon}\right)$</td>
<td>$O\left(\log\left(\frac{1}{\epsilon}\right)\right)$</td>
</tr>
<tr>
<td>Cost of an iteration</td>
<td>$O(1)$</td>
<td>$O(n)$</td>
</tr>
<tr>
<td>Total complexity*</td>
<td>$O\left(\frac{1}{\epsilon}\right)$</td>
<td>$O\left(n\log\left(\frac{1}{\epsilon}\right)\right)$</td>
</tr>
</tbody>
</table>
Comparison GD and SGD for strongly convex

<table>
<thead>
<tr>
<th></th>
<th>SGD</th>
<th>GD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration complexity</td>
<td>$O\left(\frac{1}{\varepsilon}\right)$</td>
<td>$O\left(\log\left(\frac{1}{\varepsilon}\right)\right)$</td>
</tr>
<tr>
<td>Cost of an iteration</td>
<td>$O\left(1\right)$</td>
<td>$O\left(n\right)$</td>
</tr>
<tr>
<td>Total complexity*</td>
<td>$O\left(\frac{1}{\varepsilon}\right)$</td>
<td>$O\left(n \log\left(\frac{1}{\varepsilon}\right)\right)$</td>
</tr>
</tbody>
</table>

*Total complexity  = (Iteration complexity) × (Cost of an iteration)*
Comparison GD and SGD for strongly convex

<table>
<thead>
<tr>
<th></th>
<th>SGD</th>
<th>GD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration complexity</td>
<td>$O\left(\frac{1}{\epsilon}\right)$</td>
<td>$O\left(\log\left(\frac{1}{\epsilon}\right)\right)$</td>
</tr>
<tr>
<td>Cost of an iteration</td>
<td>$O\left(1\right)$</td>
<td>$O\left(n\right)$</td>
</tr>
<tr>
<td>Total complexity*</td>
<td>$O\left(\frac{1}{\epsilon}\right)$</td>
<td>$O\left(n \log\left(\frac{1}{\epsilon}\right)\right)$</td>
</tr>
</tbody>
</table>

What happens if $\epsilon$ is small? What happens if $n$ is big?

*Total complexity = (Iteration complexity) $\times$ (Cost of an iteration)
Comparison SGD vs GD

Modern variance reduced version of SGD

Mathematical Programming
Minimizing Finite Sums with the Stochastic Average Gradient.
Comparison SGD vs GD

Modern variance reduced version of SGD

Mathematical Programming
Minimizing Finite Sums with the Stochastic Average Gradient.
Comparison SGD vs GD

log(error)

Mathematical Programming
Minimizing Finite Sums with the Stochastic Average Gradient.
Comparison SGD vs GD

Modern variance reduced version of SGD

Mathematical Programming
Minimizing Finite Sums with the Stochastic Average Gradient.
20 min tea time break?
Practical SGD for Sparse Data
Lazy SGD updates for Sparse Data

Finite Sum Training Problem

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( \langle w, x^i \rangle, y^i \right) + \frac{\lambda}{2} ||w||_2^2$$

Assume each data point $x^i$ is $s$-sparse, how many operations does each SGD step cost?
Lazy SGD updates for Sparse Data

Finite Sum Training Problem

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( \langle w, x^i \rangle, y^i \right) + \frac{\lambda}{2} \|w\|_2^2$$

Assume each data point $x^i$ is $s$-sparse, how many operations does each SGD step cost?

$$w^{t+1} = w^t - \alpha_t \left( \ell' \left( \langle w^t, x^i \rangle, y^i \right) x^i + \lambda w^t \right)$$

$$= (1 - \lambda \alpha_t) w^t - \alpha_t \ell' \left( \langle w^t, x^i \rangle, y^i \right) x^i$$
Lazy SGD updates for Sparse Data

Finite Sum Training Problem

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( \langle w, x^i \rangle, y^i \right) + \frac{\lambda}{2} \|w\|_2^2
\]

Assume each data point \( x^i \) is \( s \)-sparse, how many operations does each SGD step cost?

\[
w^{t+1} = w^t - \alpha_t \left( \ell' \left( \langle w^t, x^i \rangle, y^i \right)x^i + \lambda w^t \right)
\]

\[
= (1 - \lambda \alpha_t)w^t - \alpha_t \ell' \left( \langle w^t, x^i \rangle, y^i \right)x^i
\]

- Rescaling \( O(d) \)
- Addition sparse vector \( O(s) \)

\[
= O(d)
\]
Lazy SGD updates for Sparse Data

**SGD step**

\[ w^{t+1} = (1 - \lambda\alpha_t)w^t - \alpha_t\ell'(\langle w^t, x^i \rangle, y^i)x^i \]

**EXE:** re-write the iterates using \( w^t = \beta_tz^t \) where \( \beta_t \in \mathbb{R}, \ z^t \in \mathbb{R}^d \)

Can you update \( \beta_t \) and \( z^t \) so that each iteration is \( O(s) \)?
SGD step

\[ w^{t+1} = (1 - \lambda \alpha_t)w^t - \alpha_t \ell'(\langle w^t, x^i \rangle, y^i)x^i \]

**EXE:** re-write the iterates using \( w^t = \beta_t z^t \) where \( \beta_t \in \mathbb{R}, \ z^t \in \mathbb{R}^d \)

Can you update \( \beta_t \) and \( z^t \) so that each iteration is \( O(s) \)?

\[ \beta_{t+1} z^{t+1} = (1 - \lambda \alpha_t)\beta_t z^t - \alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i)x^i \]

\[ = (1 - \lambda \alpha_t)\beta_t \left( z^t - \frac{\alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i)}{(1 - \lambda \alpha_t)\beta_t}x^i \right) \]
Lazy SGD updates for Sparse Data

**SGD step**

\[ w^{t+1} = (1 - \lambda \alpha_t)w^t - \alpha_t \ell'(\langle w^t, x^i \rangle, y^i)x^i \]

**EXE:** re-write the iterates using \( w^t = \beta_t z^t \) where \( \beta_t \in \mathbb{R}, \ z^t \in \mathbb{R}^d \)

Can you update \( \beta_t \) and \( z^t \) so that each iteration is \( O(s) \)?

\[
\beta_{t+1} z^{t+1} = (1 - \lambda \alpha_t) \beta_t z^t - \alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i)x^i \\
= (1 - \lambda \alpha_t) \beta_t \left( z^t - \frac{\alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i)}{(1 - \lambda \alpha_t) \beta_t} x^i \right) \\
\beta_{t+1} = (1 - \lambda \alpha_t) \beta_t, \quad z^{t+1} = z^t - \frac{\alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i)}{(1 - \lambda \alpha_t) \beta_t} x^i
\]
Lazy SGD updates for Sparse Data

**SGD step**

$$w^{t+1} = (1 - \lambda \alpha_t) w^t - \alpha_t \ell'(\langle w^t, x^i \rangle, y^i) x^i$$

**EXE:** re-write the iterates using $w^t = \beta_t z^t$ where $\beta_t \in \mathbb{R}$, $z^t \in \mathbb{R}^d$

Can you update $\beta_t$ and $z^t$ so that each iteration is $O(s)$?

$$\beta_{t+1} z^{t+1} = (1 - \lambda \alpha_t) \beta_t z^t - \alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i) x^i$$

$$= (1 - \lambda \alpha_t) \beta_t \left( z^t - \frac{\alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i)}{(1 - \lambda \alpha_t) \beta_t} x^i \right)$$

$O(1)$ scaling + $O(s)$ sparse add = $O(s)$ update

$$\beta_{t+1} = (1 - \lambda \alpha_t) \beta_t, \quad z^{t+1} = z^t - \frac{\alpha_t \ell'(\beta_t \langle z^t, x^i \rangle, y^i)}{(1 - \lambda \alpha_t) \beta_t} x^i$$
Momentum
Issue with Gradient Descent

Solving the training problem:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f_i(w) =: f(w)$$

Baseline method: Gradient Descent (GD)

$$w^{t+1} = w^t - \gamma \nabla f(w^t)$$

Step size / Learning rate
Why GD and the the Issues

Local rate of change

\[ \Delta(d) := \lim_{{s \to 0^+}} \frac{f(x + ds) - f(x)}{s} \]

Max local rate

\[ \frac{\nabla f(w^t)}{\|\nabla f(w^t)\|} := \max_{{w \in \mathbb{R}^d}} \Delta(d) \]
subject to  \[ \|d\| = 1 \]

GD is the “steepest descent”
Issue with Gradient Descent

\[ f(x_1, x_2) = 100(x_1 - x_2^2)^2 + (1 - x_2)^2 \]

Get’s stuck in “flat” valleys  
Give momentum to keep going
Adding some Momentum to GD

Heavey Ball Method:

\[ w^{t+1} = w^t - \gamma \nabla f(w^t) + \beta (w^t - w^{t-1}) \]

Adds “Inertia” to update
Adding some Momentum to GD

Heavey Ball Method:

\[ w^{t+1} = w^t - \gamma \nabla f(w^t) + \beta (w^t - w^{t-1}) \]

GD with momentum (GDm):

\[ m^t = \beta m^{t-1} + \nabla f(w^t) \]

\[ w^{t+1} = w^t - \gamma m^t \]
Issue with Gradient Descent

Get's stuck in “flat” valleys

Solution

Give momentum to keep going
GDm and Heavy Ball Equivalence

GD with momentum:

\[ m^t = \beta m^{t-1} + \nabla f(w^t) \]
\[ w^{t+1} = w^t - \gamma m^t \]
GDm and Heavy Ball Equivalence

**GD with momentum:**

\[ m^t = \beta m^{t-1} + \nabla f(w^t) \]
\[ w^{t+1} = w^t - \gamma m^t \]

\[
\begin{align*}
   w^{t+1} &= w^t - \gamma m^t \\
   &= w^t - \gamma (\beta m^{t-1} + \nabla f(w^t)) \\
   &= w^t - \gamma \nabla f(w^t) - \gamma \beta m^{t-1} \\
   &= w^t - \gamma \nabla f(w^t) + \frac{\gamma \beta}{\gamma} (w^t - w^{t-1})
\end{align*}
\]
GDm and Heavy Ball Equivalence

GD with momentum:

\[ m^t = \beta m^{t-1} + \nabla f(w^t) \]
\[ w^{t+1} = w^t - \gamma m^t \]

\[
\begin{align*}
  w^{t+1} &= w^t - \gamma m^t \\
         &= w^t - \gamma (\beta m^{t-1} + \nabla f(w^t)) \\
         &= w^t - \gamma \nabla f(w^t) - \gamma \beta m^{t-1} \\
         &= w^t - \gamma \nabla f(w^t) + \frac{\gamma \beta}{\gamma} (w^t - w^{t-1})
\end{align*}
\]

\[ m^{t-1} = -\frac{1}{\gamma} (w^t - w^{t-1}) \]
GDm and Heavy Ball Equivalence

GD with momentum:

\[ m^t = \beta m^{t-1} + \nabla f(w^t) \]
\[ w^{t+1} = w^t - \gamma m^t \]

\[ w^{t+1} = w^t - \gamma m^t \]
\[ = w^t - \gamma (\beta m^{t-1} + \nabla f(w^t)) \]
\[ = w^t - \gamma \nabla f(w^t) - \gamma \beta m^{t-1} \]
\[ = w^t - \gamma \nabla f(w^t) + \frac{\gamma \beta}{\gamma} (w^t - w^{t-1}) \]

\[ m^{t-1} = -\frac{1}{\gamma}(w^t - w^{t-1}) \]

\[ w^{t+1} = w^t - \gamma \nabla f(w^t) + \beta(w^t - w^{t-1}) \]
GDm and Heavy Ball Equivalence

GD with momentum:

\[ m^t = \beta m^{t-1} + \nabla f(w^t) \]
\[ w^{t+1} = w^t - \gamma m^t \]

\[
\begin{align*}
  w^{t+1} &= w^t - \gamma m^t \\
  &= w^t - \gamma (\beta m^{t-1} + \nabla f(w^t)) \\
  &= w^t - \gamma \nabla f(w^t) - \gamma \beta m^{t-1} \\
  &= w^t - \gamma \nabla f(w^t) + \frac{\gamma \beta}{\gamma} (w^t - w^{t-1})
\end{align*}
\]

Heavey Ball Method:

\[
  w^{t+1} = w^t - \gamma \nabla f(w^t) + \beta (w^t - w^{t-1})
\]
Theorem  Let $f$ be $\mu$–strongly convex and $L$–smooth, that is
\[
\mu I \preceq \nabla^2 f(w) \preceq LI, \quad \forall w \in \mathbb{R}^d
\]
If $\gamma = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}$ and $\beta = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$ then SGDm converges
\[
\|w^t - w^*\| \leq \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^t \|w^0 - w^*\|
\]
\[
\kappa := \frac{L}{\mu}
\]
Convergence of Gradient Descent with Momentum

**Theorem**  Let $f$ be $\mu$–strongly convex and $L$–smooth, that is

$$\mu I \preceq \nabla^2 f(w) \preceq LI, \quad \forall w \in \mathbb{R}^d$$

If $\gamma = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}$ and $\beta = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$ then SGDm converges

$$\|w^t - w^*\| \leq \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^t \|w^0 - w^*\|$$

where $\kappa := \frac{L}{\mu}$

**Corollary**  $t \geq \frac{\sqrt{\kappa} + 1}{2} \log \left(\frac{1}{\epsilon}\right)$

$$\frac{\|w^t - w^*\|}{\|w^0 - w^*\|} \leq \epsilon$$
Convergence of Gradient Descent with Momentum

**Theorem** Let $f$ be $\mu$–strongly convex and $L$–smooth, that is

$$\mu I \preceq \nabla^2 f(w) \preceq LI, \ \forall w \in \mathbb{R}^d$$

If $\gamma = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}$ and $\beta = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$ then SGDm converges

$$\|w^t - w^*\| \leq \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^t \|w^0 - w^*\|$$

Compare with $\kappa \log(1/\epsilon)$ of GD

**Corollary**

$$t \geq \frac{\sqrt{\kappa} + 1}{2} \log \left(\frac{1}{\epsilon}\right) \Rightarrow \frac{\|w^t - w^*\|}{\|w^0 - w^*\|} \leq \epsilon$$

$\kappa := L/\mu$
Fundamental Theorem of Calculus

\[ \int_{s=0}^{1} \nabla^2 f(w_s) ds (w^t - w^*) = \nabla f(w^t) - \nabla f(w^*) = \nabla f(w^t) \]

\[ w_s := w^* + s(w^t - w^*) \]
Proof sketch: GDm convergence

Fundamental Theorem of Calculus

\[
\int_{s=0}^{1} \nabla^2 f(w_s) ds (w^t - w^*) = \nabla f(w^t) - \nabla f(w^*) = \nabla f(w^t)
\]

\[w_s := w^* + s(w^t - w^*)\]

\[w^{t+1} - w^* = w^t - w^* - \gamma \nabla f(w^t) + \beta (w^t - w^{t-1}) + w^* - w^*\]

\[= (I - \gamma \int_{s=0}^{1} \nabla^2 f(w^s)) (w^t - w^*) + \beta (w^t - w^{t-1})\]

\[= ((1 + \beta)I - \gamma \int_{s=0}^{1} \nabla^2 f(w^s)) (w^t - w^*) - \beta (w^{t-1} - w^*)\]
Proof sketch: GDm convergence

\[ \int_{s=0}^{1} \nabla^2 f(w_s) ds (w^t - w^*) = \nabla f(w^t) - \nabla f(w^*) = \nabla f(w^t) \]

\[ w_s := w^* + s(w^t - w^*) \]

\[ w^{t+1} - w^* = w^t - w^* - \gamma \nabla f(w^t) + \beta (w^t - w^{t-1}) + w^* - w^* \]

\[ = \left( I - \gamma \int_{s=0}^{1} \nabla^2 f(w_s) \right) (w^t - w^*) + \beta (w^t - w^{t-1}) \]

\[ = \left( (1 + \beta) I - \gamma \int_{s=0}^{1} \nabla^2 f(w_s) \right) (w^t - w^*) - \beta (w^{t-1} - w^*) \]

\[ =: A_s \]
Proof sketch: GDm convergence

Fundamental Theorem of Calculus

\[ \int_{s=0}^{1} \nabla^2 f(w_s) ds (w^t - w^*) = \nabla f(w^t) - \nabla f(w^*) = \nabla f(w^t) \]

\[ w_s := w^* + s(w^t - w^*) \]

\[ w^{t+1} - w^* = w^t - w^* - \gamma \nabla f(w^t) + \beta (w^t - w^{t-1}) + w^* - w^* \]

\[ = \left( I - \gamma \int_{s=0}^{1} \nabla^2 f(w^s) \right) (w^t - w^*) + \beta (w^t - w^{t-1}) \]

\[ = \left( (1 + \beta)I - \gamma \int_{s=0}^{1} \nabla^2 f(w^s) \right) (w^t - w^*) - \beta (w^{t-1} - w^*) \]

\[ =: A_s \]

\[ = A_s (w^t - w^*) - \beta (w^{t-1} - w^*) \]
Fundamental Theorem of Calculus

Proof sketch: GDm convergence

\[
\int_{s=0}^{1} \nabla^2 f(w_s)ds(w^t - w^*) = \nabla f(w^t) - \nabla f(w^*) = \nabla f(w^t)
\]

\[w_s := w^* + s(w^t - w^*)\]

\[w^{t+1} - w^* = w^t - w^* - \gamma \nabla f(w^t) + \beta (w^t - w^{t-1}) + w^* - w^*\]

\[= \left( I - \gamma \int_{s=0}^{1} \nabla^2 f(w^s) \right) (w^t - w^*) + \beta (w^t - w^{t-1})\]

\[= \left( (1 + \beta)I - \gamma \int_{s=0}^{1} \nabla^2 f(w^s) \right) (w^t - w^*) - \beta (w^{t-1} - w^*)\]

\[=: A_s\]

\[= A_s(w^t - w^*) - \beta (w^{t-1} - w^*)\]

Depends on past. Difficult recurrence
Proof: Convergence of Heavy Ball

\[ z^{t+1} = \begin{bmatrix} w^{t+1} - w^* \\ w^t - w^* \end{bmatrix} \in \mathbb{R}^{2d} \]
Proof: Convergence of Heavy Ball

\[
\tilde{z}^{t+1} = \begin{bmatrix} w^{t+1} - w^* \\ w^t - w^* \end{bmatrix} \in \mathbb{R}^{2d}
\]

\[
\tilde{z}^{t+1} = \begin{bmatrix} w^{t+1} - w^* \\ w^t - w^* \end{bmatrix} = \begin{bmatrix} A_s(w^t - w^*) - \beta(w^{t-1} - w^*) \\ w^t - w^* \end{bmatrix}
\]
Proof: Convergence of Heavy Ball

\[
\begin{bmatrix}
  \hat{z}^{t+1} \\
  \hat{z}^{t+1}
\end{bmatrix} = \begin{bmatrix}
  w^{t+1} - w^* \\
  w^t - w^*
\end{bmatrix} \in \mathbb{R}^{2d}
\]

\[
\hat{z}^{t+1} = \begin{bmatrix}
  w^{t+1} - w^* \\
  w^t - w^*
\end{bmatrix} = \begin{bmatrix}
  A_s(w^t - w^*) - \beta(w^{t-1} - w^*) \\
  w^t - w^*
\end{bmatrix}
\]

\[
= \begin{bmatrix}
  A_s & -I\beta \\
  I & 0
\end{bmatrix} \begin{bmatrix}
  w^t - w^* \\
  w^{t-1} - w^*
\end{bmatrix}
\]
Proof: Convergence of Heavy Ball

\[
\begin{bmatrix}
  \omega^{t+1} \\
  \omega^t - \omega^*
\end{bmatrix} = \begin{bmatrix}
  w^{t+1} - w^* \\
  w^t - w^*
\end{bmatrix} \in \mathbb{R}^{2d}
\]

\[
\begin{bmatrix}
  \omega^{t+1} \\
  \omega^t - \omega^*
\end{bmatrix} = \begin{bmatrix}
  A_s (w^t - w^*) - \beta (w^{t-1} - w^*) \\
  w^t - w^*
\end{bmatrix}
\]

\[
= \begin{bmatrix}
  A_s & -I \beta \\
  I & 0
\end{bmatrix}
\begin{bmatrix}
  w^t - w^* \\
  w^{t-1} - w^*
\end{bmatrix}
\]

\[
= \begin{bmatrix}
  A_s & -I \beta \\
  I & 0
\end{bmatrix} \begin{bmatrix}
  \omega^t \\
  \omega^t
\end{bmatrix}
\]
Proof: Convergence of Heavy Ball

\[ \hat{z}^{t+1} = \begin{bmatrix} w^{t+1} - w^* \\ w^t - w^* \end{bmatrix} \in \mathbb{R}^{2d} \]

\[ \hat{z}^{t+1} = \begin{bmatrix} w^{t+1} - w^* \\ w^t - w^* \end{bmatrix} = \begin{bmatrix} A_s(w^t - w^*) - \beta(w^{t-1} - w^*) \\ w^t - w^* \end{bmatrix} \]

\[ = \begin{bmatrix} A_s & -I\beta \\ I & 0 \end{bmatrix} \begin{bmatrix} w^t - w^* \\ w^{t-1} - w^* \end{bmatrix} \]

\[ = \begin{bmatrix} A_s & -I\beta \\ I & 0 \end{bmatrix} \hat{z}^t \]

Simple recurrence!
Proof: Convergence of Heavy Ball

\[ \mathbf{z}^{t+1} = \begin{bmatrix} \mathbf{w}^{t+1} - \mathbf{w}^* \\ \mathbf{w}^t - \mathbf{w}^* \end{bmatrix} \in \mathbb{R}^{2d} \]

\[ \mathbf{z}^{t+1} = \begin{bmatrix} \mathbf{w}^{t+1} - \mathbf{w}^* \\ \mathbf{w}^t - \mathbf{w}^* \end{bmatrix} = \begin{bmatrix} \mathbf{A}_s (\mathbf{w}^t - \mathbf{w}^*) - \mathbf{\beta} (\mathbf{w}^{t-1} - \mathbf{w}^*) \\ \mathbf{w}^t - \mathbf{w}^* \end{bmatrix} \]

\[ = \begin{bmatrix} \mathbf{A}_s & -\mathbf{I} \mathbf{\beta} \\ \mathbf{I} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w}^t - \mathbf{w}^* \\ \mathbf{w}^{t-1} - \mathbf{w}^* \end{bmatrix} \]

\[ = \begin{bmatrix} \mathbf{A}_s & -\mathbf{I} \mathbf{\beta} \\ \mathbf{I} & 0 \end{bmatrix} \mathbf{z}^t \]

Simple recurrence!

\[ \| \mathbf{z}^{t+1} \| \leq \| \begin{bmatrix} \mathbf{A}_s & -\mathbf{I} \mathbf{\beta} \\ \mathbf{I} & 0 \end{bmatrix} \| \| \mathbf{z}^t \| \]
Proof: Convergence of Heavy Ball

\[ \|z^{t+1}\| \leq \left\| \begin{bmatrix} A_s & -I \beta \\ I & 0 \end{bmatrix} \right\| \|z^t\| \]

\[ \|A\| := \max_{i=1, \ldots, 2n} |\lambda_i(A)| \]
Proof: Convergence of Heavy Ball

\[ \|z^{t+1}\| \leq \| \begin{bmatrix} A_s & -I \beta \\ I & 0 \end{bmatrix} \| \|z^t\| \]

\[ \|A\| := \max_{i=1, \ldots, 2n} |\lambda_i(A)| \]

**EXE on Eigenvalues:**

If \( \gamma = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2} \) and \( \beta = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}} \) then

\[ \| \begin{bmatrix} A_s & -I \beta \\ I & 0 \end{bmatrix} \| = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \]
Proof: Convergence of Heavy Ball

\[ \|z^{t+1}\| \leq \left\| \begin{bmatrix} A_s & -I\beta \\ I & 0 \end{bmatrix} \right\| \|z^t\| \]

\[ \|A\| := \max_{i=1,\ldots,2n} |\lambda_i(A)| \]

\[ (1 + \beta)I - \gamma \int_{s=0}^{1} \nabla^2 f(w^s) \]

**EXE on Eigenvalues:**

If \( \gamma = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2} \) and \( \beta = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}} \) then

\[ \left\| \begin{bmatrix} A_s & -I\beta \\ I & 0 \end{bmatrix} \right\| = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \]
Adding Momentum to SGD

Stochastic Heavey Ball Method:

\[ w^{t+1} = w^t - \gamma \nabla f_{j_t}(w^t) + \beta(w^t - w^{t-1}) \]

- Sampled i.i.d
- \( j \in \{1, \ldots, n\} \)
- \( j \sim \frac{1}{n} \)
- Adds “Inertia” to update

Adding Momentum to SGD

Stochastic Heavy Ball Method:

\[ w^{t+1} = w^t - \gamma \nabla f_{j_t}(w^t) + \beta (w^t - w^{t-1}) \]

Sampled i.i.d \( j \in \{1, \ldots, n\} \)
\[ j \sim \frac{1}{n} \]

Adds “Inertia” to update

SGD with momentum (SGDm):

\[ m^t = \beta m^{t-1} + \nabla f_{j_t}(w^t) \]

\[ w^{t+1} = w^t - \gamma m^t \]
SGDm and Averaging

\[ m^t = \beta m^{t-1} + \nabla f_{j_t}(w^t) \]
\[ = \beta m^{t-2} + \nabla f_{j_t}(w^t) + \beta \nabla f_{j_{t-1}}(w^{t-1}) \]
\[ = \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \]
SGDm and Averaging

\[ m^t = \beta m^{t-1} + \nabla f_{j_t}(w^t) \]
\[ = \beta m^{t-2} + \nabla f_{j_t}(w^t) + \beta \nabla f_{j_{t-1}}(w^{t-1}) \]
\[ = \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \]

\[ m^0 = 0 \]
SGDm and Averaging

\[ m^t = \beta m^{t-1} + \nabla f_{j_t}(w^t) \]
\[ = \beta m^{t-2} + \nabla f_{j_t}(w^t) + \beta \nabla f_{j_{t-1}}(w^{t-1}) \]
\[ = \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \]

SGD with momentum (SGDm):

\[ w^{t+1} = w^t - \gamma \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \]

SGDm and Averaging

\[ m^t = \beta m^{t-1} + \nabla f_j^t(w^t) \]
\[ = \beta m^{t-2} + \nabla f_j^t(w^t) + \beta \nabla f_{j_{t-1}}(w^{t-1}) \]
\[ = \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \]

**SGD with momentum (SGDm):**

\[ w^{t+1} = w^t - \gamma \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \]

Acts like an approximate variance reduction since

SGDm and Averaging

\[ m^t = \beta m^{t-1} + \nabla f_{j_t}(w^t) \]
\[ = \beta m^{t-2} + \nabla f_{j_t}(w^t) + \beta \nabla f_{j_{t-1}}(w^{t-1}) \]
\[ = \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \quad \text{for} \quad m^0 = 0 \]

**SGD with momentum (SGDm):**

\[ w^{t+1} = w^t - \gamma \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \]

Acts like an approximate variance reduction since

\[ \sum_{i=1}^{t} \beta^i \nabla f_{j_{t-i}}(w^{t-i}) \approx \sum_{i=1}^{n} \frac{1}{n} \nabla f_i(w^t) = \nabla f(w^t) \]

Why Machine Learners Like SGD
Why Machine Learners like SGD

Though we solve:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( h_w(x^i), y^i \right) + \lambda R(w)$$

We want to solve:

The statistical learning problem:

Minimize the expected loss over an unknown expectation

$$\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y) \sim \mathcal{D}} \left[ \ell \left( h_w(x), y \right) \right]$$

SGD can solve the statistical learning problem!
Why Machine Learners like SGD

The statistical learning problem:
Minimize the expected loss over an unknown expectation

\[
\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y) \sim \mathcal{D}} [\ell (h_w(x), y)]
\]

**SGD approximate for learning**
Set \( w^0 = 0, \alpha > 0 \)
for \( t = 0, 1, 2, \ldots, T - 1 \)
sample \( (x, y) \sim \mathcal{D} \)
calculate \( v_t \in \partial \ell (h_{w^t}(x), y) \)
\[
w^{t+1} = w^t - \alpha v_t
\]
Output \( \overline{w}^T = \frac{1}{T} \sum_{t=1}^{T} w^t \)
Bring laptops for Thursday TD!
RMG, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin and Peter Richtárik (2019), ICML
**SGD: general analysis and improved rates**

RMG, P. Richtarik, F. Bach (2018), preprint online
**Stochastic quasi-gradient methods: Variance reduction via Jacobian sketching**

**Optimal mini-batch and step sizes for SAGA**

O. Sebbouh, N. Gazagnadou, S. Jelassi, F. Bach, RMG
Neurips 2019, preprint online. **Towards closing the gap between the theory and practice of SVRG**