Introduction to Machine Learning

Lecture 6: Stochastic Gradient Descent

Hao Wang
Email: wanghao1@shanghaitech.edu.cn

School of Information Science and Technology
ShanghaiTech University, Shanghai

March 14, 2023

Outline

Stochastic gradient descent (SGD)

Convergence Analysis

Noise Reduction

Stochastic gradient descent (SGD)

- Empirical loss:

$$J(w) = \frac{1}{2n} \sum_{j}^{n} J_j(w)$$

e.g. MSE: $J_j(w) = ({\mathbf{x}^j}^T w - y^j)^2$

- Batch gradient of empirical loss:

$$\nabla J(w) = \frac{1}{n} \sum_{j}^{n} \nabla J_j(w)$$

e.g. $\nabla J_j(w) = ({\mathbf{x}^j}^T w - y^j) \cdot \mathbf{x}^j$

- Stochastic (or "online") gradient descent:

$$w^{k+1} \leftarrow w^k - \alpha^k \nabla J_j(w^k)$$

- Use updates based on individual datum j, chosen at random

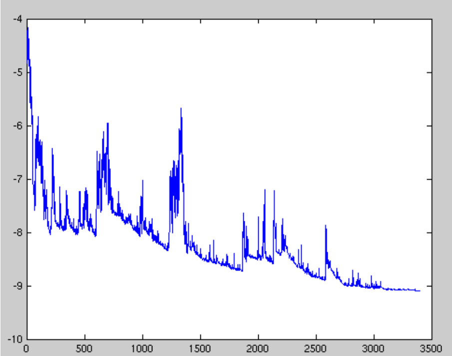
Stochastic gradient descent

- Batch GD is a monotone (for what?) algorithm (why?).
- SGD is not a monotone algorithm (why?).

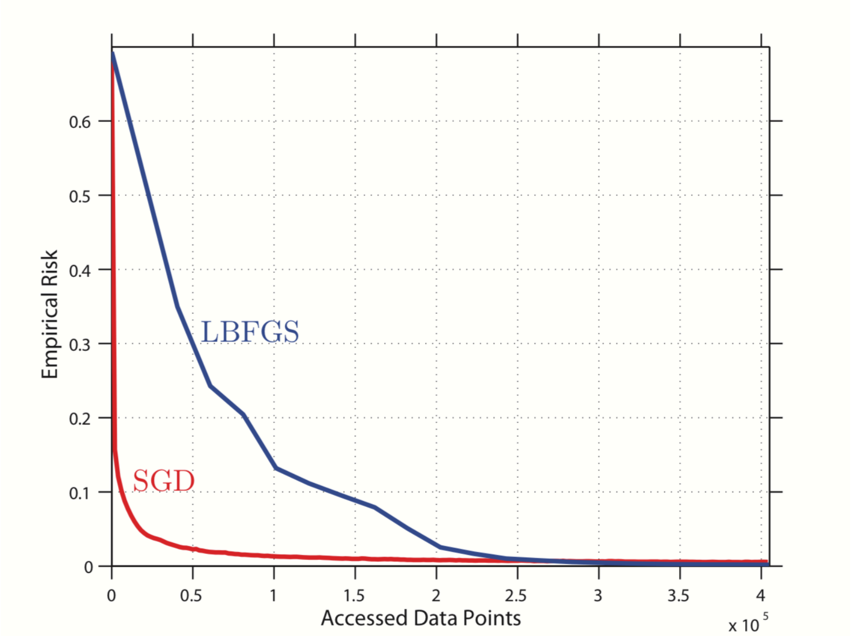
Definition

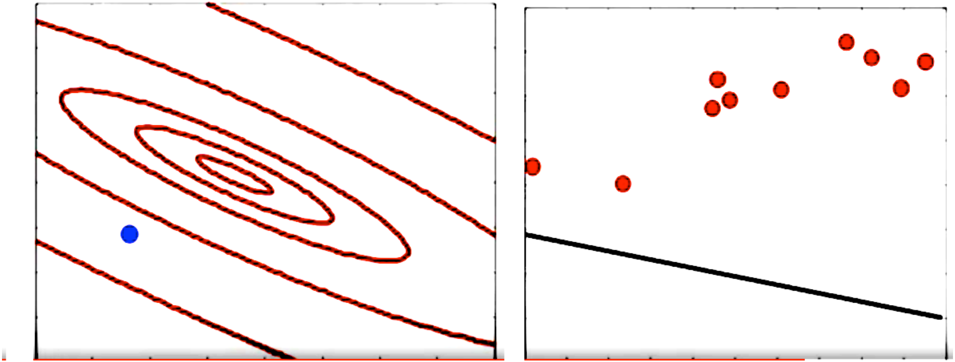
Each set of n consecutive accesses is called an epoch.

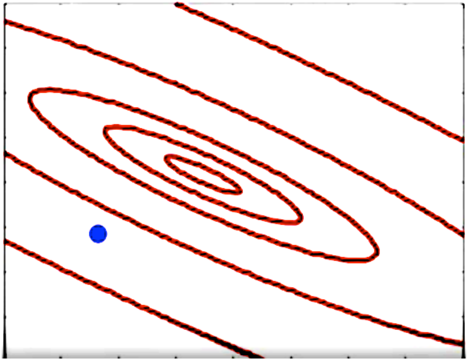
- The batch method performs only one step per epoch.
- SG performs n steps per epoch.

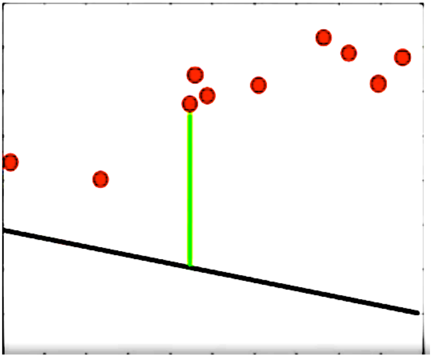


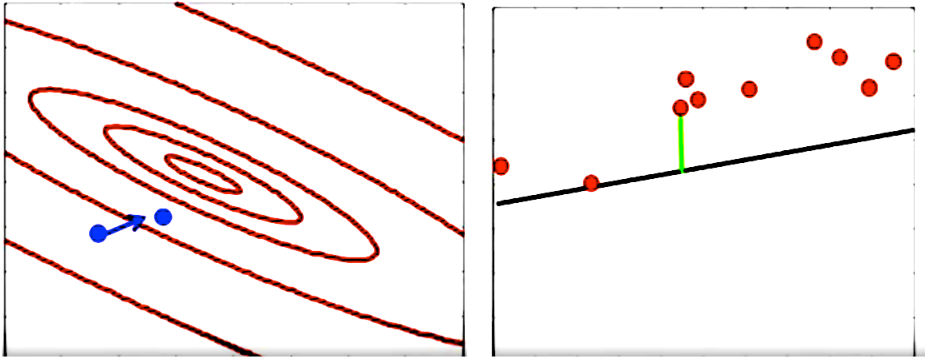
SGD and LBFGS

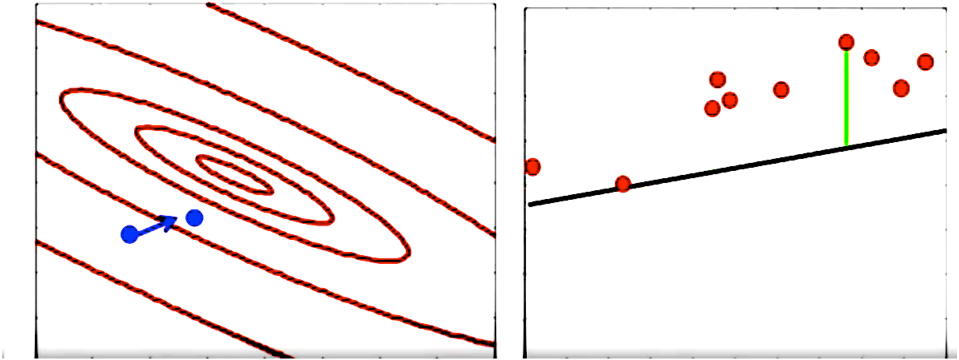


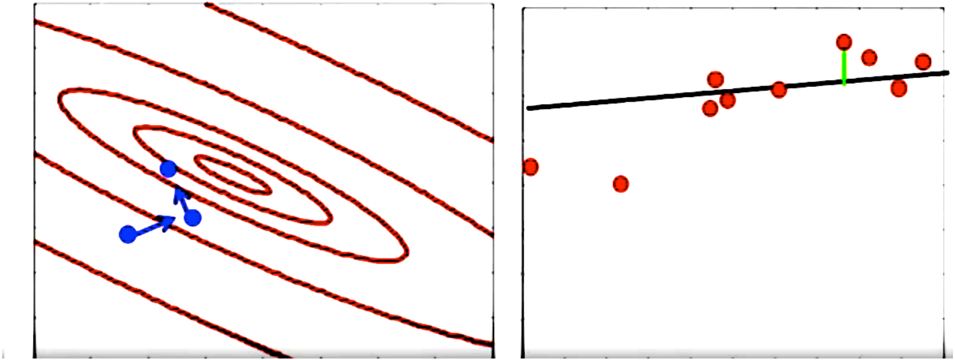


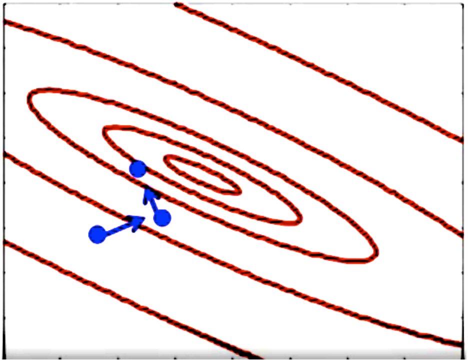


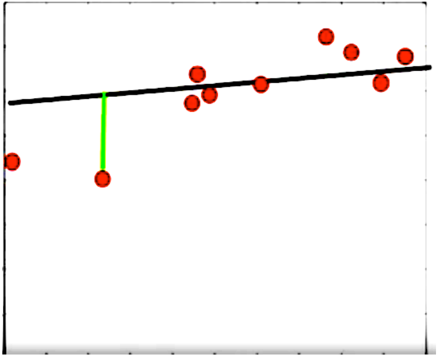


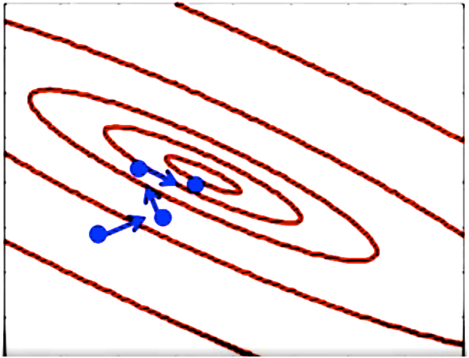


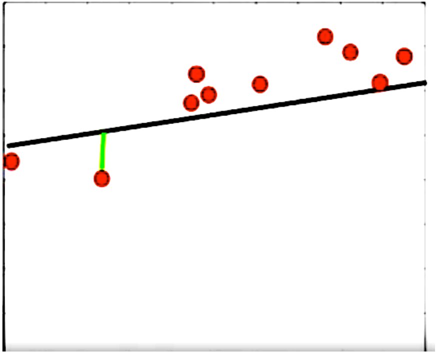




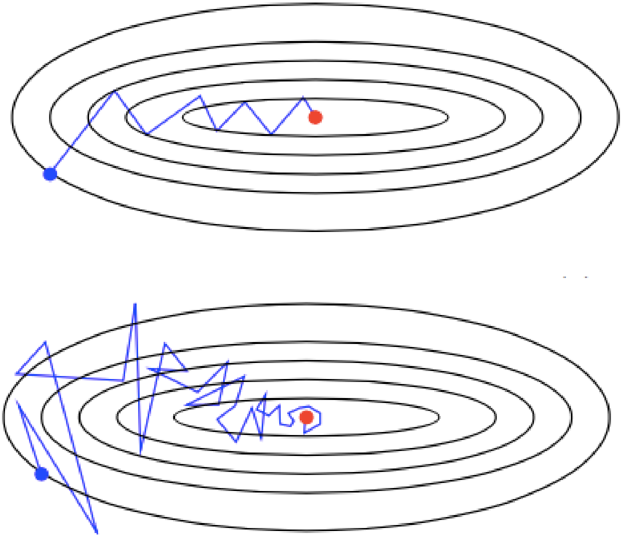






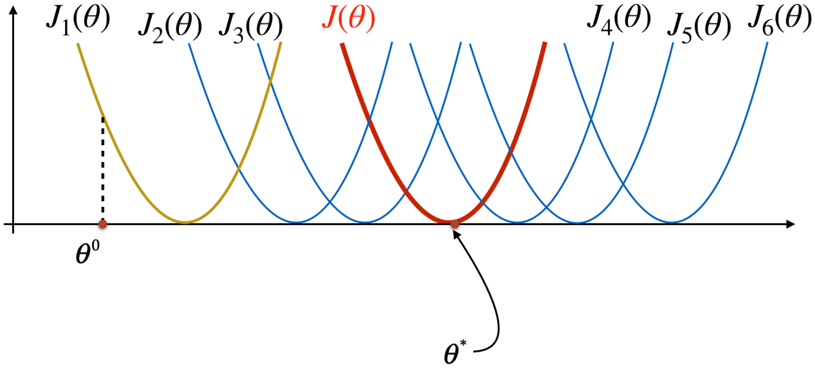


Stochastic vs deterministic



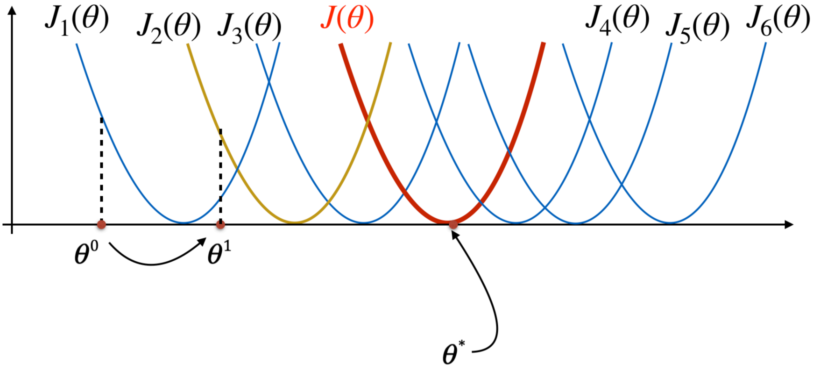
Fixed learning rate

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} J_i(\theta)$$



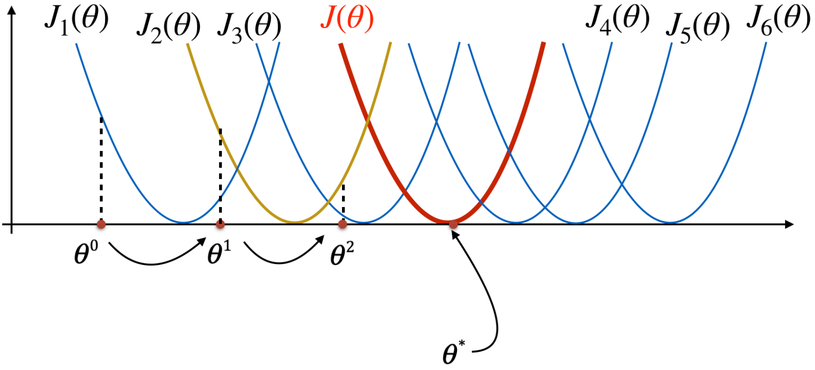
Fixed learning rate

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} J_i(\theta)$$



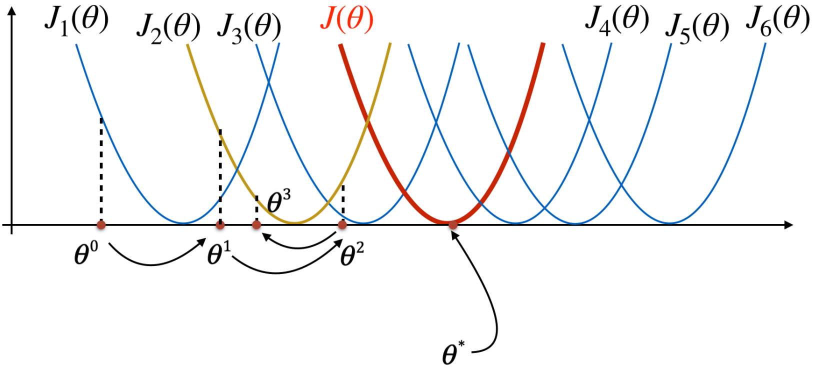
Fixed learning rate

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} J_i(\theta)$$



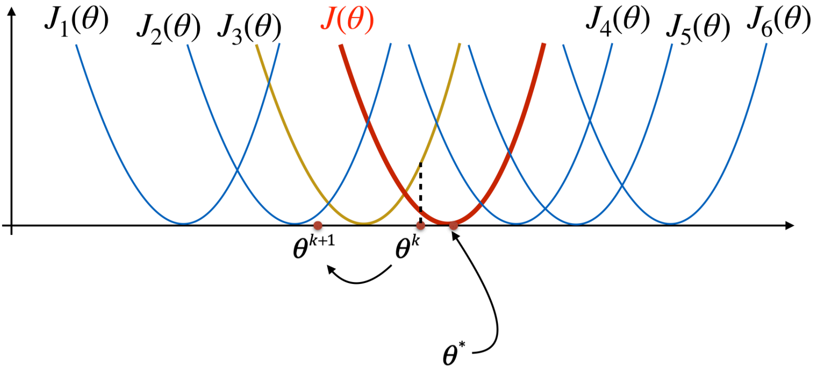
Fixed learning rate

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} J_i(\theta)$$



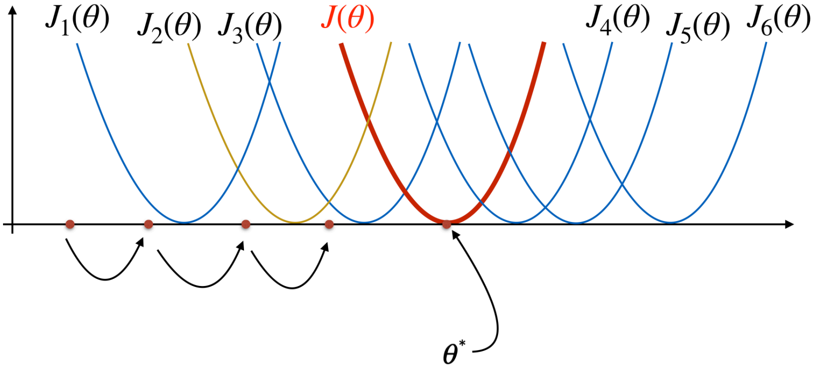
Fixed learning rate

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} J_i(\theta)$$



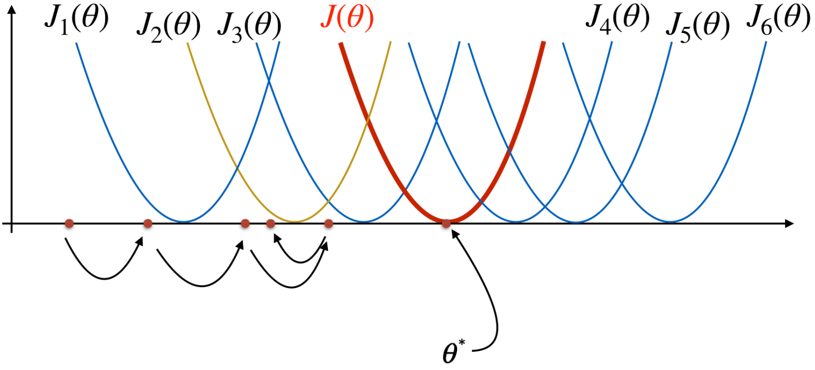
Diminishing learning rate

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} J_i(\theta)$$



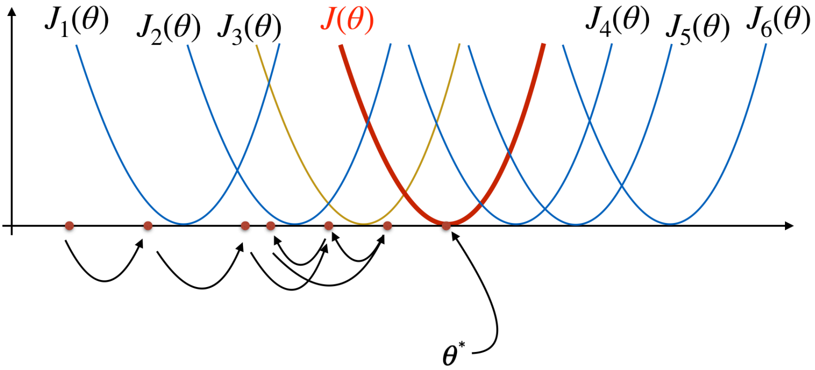
Diminishing learning rate

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} J_i(\theta)$$



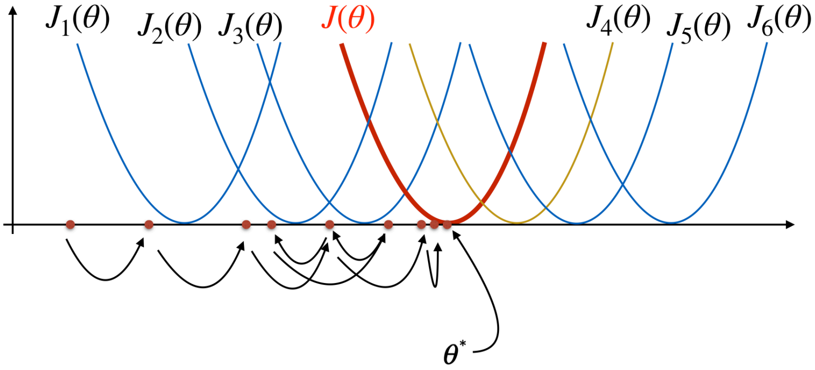
Diminishing learning rate

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} J_i(\theta)$$



Diminishing learning rate

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} J_i(\theta)$$



Is this what we want?

Mini-Batch stochastic gradient

- Empirical loss:

$$J(w) = \frac{1}{n} \sum_{j}^{n} J_j(w)$$

- Batch gradient of empirical loss:

$$\nabla J(w) = \frac{1}{n} \sum_{j}^{n} \nabla J_j(w)$$

- Stochastic (or "online") gradient descent: $\mathcal{S}_k \subset \{1, ..., n\}$

  - $w^{k+1} \leftarrow w^k - \alpha^k \frac{1}{|\mathcal{S}_k|} \sum_{j \in \mathcal{S}_k} \nabla J_j(w^k)$

  - $|\mathcal{S}_k|$ may also vary

Outline

Risk minimization

Minimizing the loss:

$$\min_{w} \quad F(w) = \begin{cases} R(w) = \mathbb{E}[f(w;\xi)] & \text{expected risk} \\ \text{or} & \\ R_n(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w) & \text{empirical risk} \end{cases}$$

For empirical risk: （**每个样本都是随机变量的一次采样**）

$$R_n(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w) = \frac{1}{n}\sum_{i=1}^{n} f(w;\xi_i)$$

$$f_i(w) = f(w;\xi_i)$$

Stochastic Gradient

The stochastic gradient is then defined as $g(w_k, \xi_k)$:

$$g(w_k, \xi_k) = \begin{cases} \nabla f(w_k; \xi_k), \text{ or} \\ \frac{1}{n_k} \sum_{i=1}^{n_k} \nabla f(w_k; \xi_{k,i}) \end{cases}$$

- ξ_k is a seed for generating a stochastic direction; e.g., a realization of it may represent the choice of a single training sample as in the simple SG method, or may represent a set of samples as in the minibatch SG method.
- $g(w_k, \xi_k)$ could represent a stochastic gradient—i.e., an unbiased estimator of $\nabla F(w_k)$

Algorithm

Algorithm 2.1 Stochastic Gradient (SG) Method

1: Choose an initial iterate w_1.
2: **for** $k = 1, 2, \dots$ **do**
3: Generate a realization of the random variable ξ_k
4: Compute a stochastic vector $g(w_k, \xi_k)$
5: Choose a stepwise $\alpha_k > 0$
6: Set the new iterate as $w_{k+1} \leftarrow w_k - \alpha_k g(w_k, \xi_k)$
7: **end for**

Convergence

Assumption

(Lipschitz-continuous objective gradients). The objective function $F : \mathbb{R}^d \to \mathbb{R}$ is continuously differentiable and the gradient function of F, namely, $\nabla F : \mathbb{R}^d \to \mathbb{R}^d$, is Lipschitz continuous with Lipschitz constant $L > 0$, i.e.,

$$\|\nabla F(w) - \nabla F(\bar{w})\|_2 \leq L\|w - \bar{w}\|_2 \quad \textit{for all } \{w, \bar{w}\} \subset \mathbb{R}^d.$$

This means

$$F(w) \leq F(\bar{w}) + \nabla F(\bar{w})^T(w - \bar{w}) + \tfrac{1}{2}L\|w - \bar{w}\|_2^2 \text{ for all } \{w, \bar{w}\} \subset \mathbb{R}^d.$$

Convergence

Lemma

The iterates of SG, satisfy the following inequality for all $k \in \mathbb{N}$:

$$\mathbb{E}_{\xi_k}[F(w_{k+1})] - F(w_k) \le -\alpha_k \nabla F(w_k)^T \mathbb{E}_{\xi_k}[g(w_k, \xi_k)] + \frac{1}{2}\alpha_k^2 L \mathbb{E}_{\xi_k}[\|g(w_k, \xi_k)\|_2^2].$$

Therefore, if $g(w_k, \xi_k)$ is an unbiased estimate of $\nabla F(w_k)$, then it follows from Lemma 4.2 that

$$\mathbb{E}_{\xi_k}[F(w_{k+1})] - F(w_k) \le -\alpha_k \|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha_k^2 L \mathbb{E}_{\xi_k}[\|g(w_k, \xi_k)\|_2^2]$$

In order to limit the second-order term of α, we need to restrict the variance of $g(w_k, \xi_k)$, i.e.,

$$\mathbb{V}_{\xi_k}[g(w_k, \xi_k)] := \mathbb{E}_{\xi_k}[\|g(w_k, \xi_k)\|_2^2] - \|\mathbb{E}_{\xi_k}[g(w_k, \xi_k)]\|_2^2$$

Convergence

Assumption

(First and second moment limits). The objective function and SG satisfy the following conditions:

1. *The sequence of iterates* $\{w_k\}$ *is contained in an open set over which* F *is bounded below by a scalar* F_{inf}.
2. *There exists scalar* $\mu_G \geq \mu > 0$ *such that, for all* $k \in \mathbb{N}$,

$$\nabla F(w_k)^T \mathbb{E}_{\xi_k}[g(w_k;\xi_k)] \geq \mu \|\nabla F(w_k)\|_2^2 \quad \text{and}$$
$$\|\mathbb{E}_{\xi_k}[g(w_k,\xi_k)]\|_2 \leq \mu_G \|\nabla F(w_k)\|_2$$

3. *There exist scalars* $M \geq 0$ *and* $M_V \geq 0$ *such that, for all* $k \in \mathbb{N}$

$$\mathbb{V}_{\xi_k}[g(w_k,\xi_k)] \leq M + M_V \|\nabla F(w_k)\|_2^2$$

Convergence

From the above assumption, we have that

$$\mathbb{E}_{\xi_k}[\|g(w_k,\xi_k)]\|_2^2] \leq M + M_G\|\nabla F(w_k)\|_2^2 \quad \text{with} \quad M_G := M_V + \mu_G^2 \geq \mu^2 > 0.$$

Lemma

The iterates of SG satisfy the following inequalities for all $k \in \mathbb{N}$:

$$\begin{aligned}\mathbb{E}_{\xi_k}[F(w_{k+1})] - F(w_k) &\leq -\mu\alpha_k\|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha_k^2 L\mathbb{E}_{\xi_k}[\|g(w_k,\xi_k)\|_2^2] \\ &\leq -(\mu - \frac{1}{2}\alpha_k L M_G)\alpha_k\|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha_k^2 LM.\end{aligned}$$

Convergence

Assumption

(strong convexity). The objective function $F : \mathbb{R}^d \to \mathbb{R}$ *is strongly convex in that there exits a constant* $c > 0$ *such that*

$$F(\bar{w}) \geq F(w) + \nabla F(w)^T(\bar{w} - w) + \frac{1}{2}c\|\bar{w} - w\|_2^2 \quad \forall (\bar{w}, w) \in \mathbb{R}^d \times \mathbb{R}^d.$$

Hence, F *has a unique minimizer, denoted as* $w_* \in \mathbb{R}^d$ *with* $F_* := F(w_*)$*.*

$$\implies F(w) - F(w_*) \leq \frac{1}{2c}\|\nabla F(w)\|_2^2 \quad \forall w \in \mathbb{R}^d$$

Since w_k is determined by the realization of the independent random variables $\{\xi_1, \xi_2, \ldots, \xi_{k-1}\}$, the *total expectation* of $F(w_k)$ for any k can be taken as

$$\mathbb{E}[F(w_k)] = \mathbb{E}_{\xi_1}\mathbb{E}_{\xi_2}\ldots\mathbb{E}_{\xi_{k-1}}[F(w_k)]$$

Convergence (strongly convex objective, fixed stepsize)

Theorem

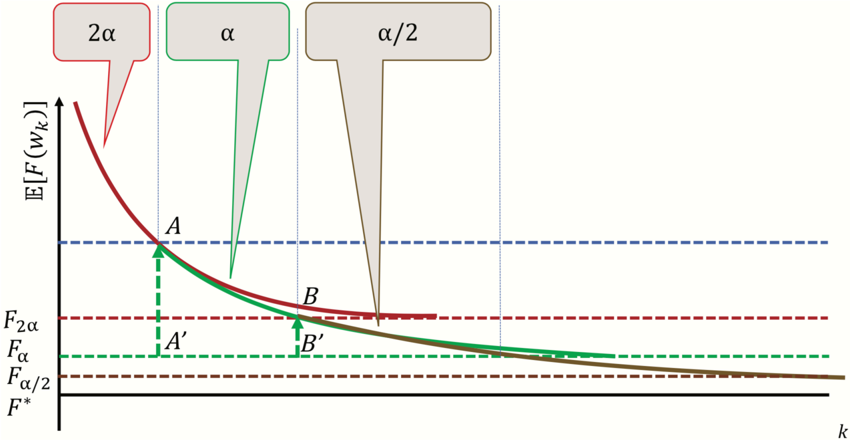
Suppose that the SG method is run with a fixed stepsize, $\alpha_k = \bar{\alpha}$*, satisfying*

$$0 < \bar{\alpha} \leq \frac{\mu}{LM_G}.$$

Then the expected optimality gap satisfies the following inequality for all k

$$\mathbb{E}[F(w_k) - F_*] \leq \frac{\bar{\alpha}LM}{2c\mu} + (1 - \bar{\alpha}\mu)^{k-1}\left(F(w_1) - F_* - \frac{\bar{\alpha}LM}{2c\mu}\right)$$
$$\xrightarrow{k\to\infty} \frac{\bar{\alpha}LM}{2c\mu}$$

Convergence (fixed learning rate)



Convergence (strongly convex objective, diminishing stepsizes)

Theorem

Suppose that the SG method is run with a fixed stepsize,

$$\alpha_k = \frac{\beta}{\gamma + k} \text{ for some } \beta > \frac{1}{c\mu} \text{ and } \gamma > 0 \text{ such that } \alpha_1 \leq \frac{\mu}{LM_G}.$$

Then, for all $k \in \mathbb{N}$, the expected optimality gap satisfies

$$\mathbb{E}[F(w_k) - F_*] \leq \frac{\nu}{\gamma + k}$$

where

$$\nu := \max\{\frac{\beta^2 LM}{2(\beta c\mu - 1)}, (\gamma + 1)(F(w_1) - F_*)\}$$

Convergence (nonconvex objective, fixed stepsize)

Now suppose F is not necessarily convex.

Theorem

Suppose that SG is run with a fixed stepsize $\alpha_k = \bar{\alpha}$ for all k, satisfying

$$0 < \bar{\alpha} \leq \frac{\mu}{LM_G}.$$

Then, the expected sum of squares and average-squared gradient of F corresponding to the SG iterates satisfy the following inequalities for all $K \in \mathbb{N}$:

$$\mathbb{E}\left[\sum_{k=1}^{K} \|\nabla F(w_k)\|_2^2\right] \leq \frac{K\bar{\alpha}LM}{\mu} + \frac{2(F(w_1) - F_{\inf})}{\mu\bar{\alpha}},$$

so that

$$\frac{1}{K}\mathbb{E}\left[\sum_{k=1}^{K} \|\nabla F(w_k)\|_2^2\right] \leq \frac{K\bar{\alpha}LM}{\mu} + \frac{2(F(w_1) - F_{\inf})}{K\mu\bar{\alpha}} \xrightarrow{K\to\infty} \frac{\bar{\alpha}LM}{\mu}.$$

Convergence (nonconvex objective, diminishing stepsize)

Theorem

Suppose that the SG method is run with a stepsize sequence satisfying. Then

$$\sum_{k=1}^{\infty} \alpha_k = \infty \quad \textit{and} \quad \sum_{k=1}^{\infty} \alpha_k^2 < \infty.$$

More precisely, let $A_K = \sum_{k=1}^{K} \alpha_k$, then

$$\mathbb{E}\left[\sum_{k=1}^{K} \alpha_k \|\nabla F(w_k)\|_2^2\right] < \infty,$$

so that

$$\mathbb{E}\left[\frac{1}{A_K}\sum_{k=1}^{K} \|\nabla F(w_k)\|_2^2\right] \overset{K\to\infty}{\longrightarrow} 0.$$

Convergence (nonconvex objective, diminishing stepsize)

Corollary

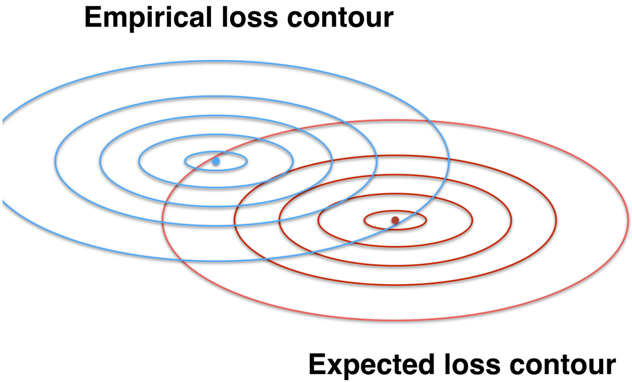
For any K, let $k(K) \in \{1, \ldots, K\}$ represents a random index chosen with probabilities proportional to $\{\alpha_k\}_{k=1}^{K}$. Then, $\|\nabla F(w_{k(K)})\|_2 \xrightarrow{K\to\infty} 0$ in probability.

Corollary

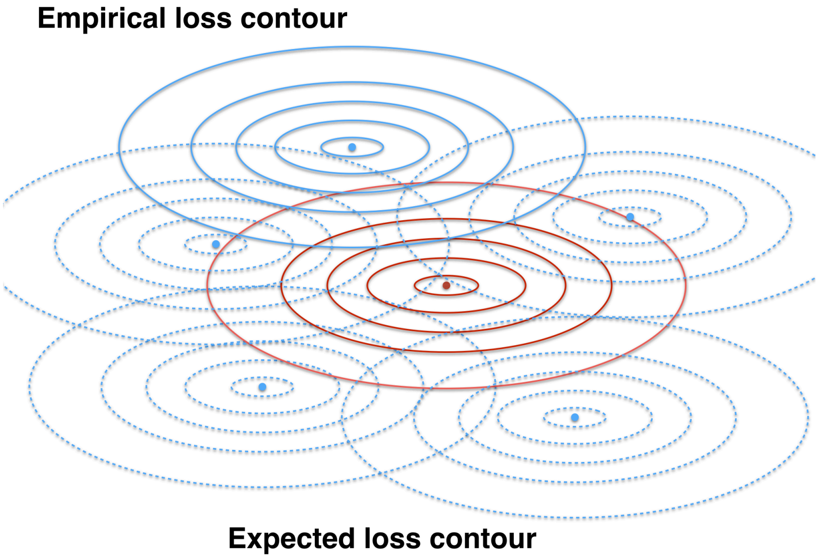
If F is twice differentiable, and that the mapping $w \to \|\nabla F(w)\|_2^2$ has Lipschitz-continuous derivatives, then

$$\lim_{k\to\infty} \mathbb{E}[\|\nabla F(w_k)\|_2^2] = 0.$$

Batch or Stochastic? Early termination



Early termination



Batch or Stochastic? Work complexity for large-scale learning

- In a big data scenario, let's compare GD and SGD.

- Suppose that both the expected risk R and the empirical risk R_n attain their minima with parameter vectors

$$w_* \in \arg\min R(w) \quad \text{and} \quad w_n \in \arg\min R_n(w).$$

- Let $\tilde{w}_n$ be the approximate empirical risk minimizer returned by a given optimization algorithm when the time budget $\mathcal{T}_{\max}$ is exhausted.

- Let $\epsilon := \mathbb{E}[R_n(\tilde{w}_n) - R_n(w_n)]$ you end up with your optimization tool, within time $\mathcal{T}_{\max}$.

Work complexity for large-scale learning

- The total error
$$\mathbb{E}[R(\tilde{w}_n)] = \underbrace{R(w_*)}_{\mathcal{E}_{app}(\mathcal{H})} + \underbrace{\mathbb{E}[R(w_n) - R(w_*)]}_{\mathcal{E}_{est}(\mathcal{H}, n)} + \underbrace{\mathbb{E}[R(\tilde{w}_n) - R(w_n)]}_{\mathcal{E}_{opt}(\mathcal{H}, n, \epsilon)}.$$

- The "quality" of your learning
$$\min_{n,\epsilon} \mathcal{E}(n, \epsilon) = \mathbb{E}[R(\tilde{w}_n) - R(w_*)] \text{ s.t. } \mathcal{T}(n, \epsilon) \leq \mathcal{T}_{\max}.$$

- For the error function, a direct application of the uniform laws of large numbers yields:
$$\begin{aligned}\mathcal{E}(n, \epsilon) = \mathbb{E}[R(\tilde{w}_n) - R(w_*)] = & \underbrace{\mathbb{E}[R(\tilde{w}_n) - R_n(\tilde{w}_n)]}_{= \mathcal{O}\left(\sqrt{\log(n)/n}\right)} + \underbrace{\mathbb{E}[R_n(\tilde{w}_n) - R_n(w_n)]}_{= \epsilon} \\ & + \underbrace{\mathbb{E}[R_n(w_n) - R_n(w_*)]}_{\leq 0} + \underbrace{\mathbb{E}[R_n(w_*) - R(w_*)]}_{= \mathcal{O}\left(\sqrt{\log(n)/n}\right)},\end{aligned}$$

Work complexity for large-scale learning

- We have the upper bound
$$\mathcal{E}(n, \epsilon) = \mathcal{O}\left(\sqrt{\frac{\log(n)}{n}} + \epsilon\right).$$
- For cases where loss function is strongly convex, or the data distribution satisfies certain assumptions, it is possible to show that
$$\mathcal{E}(n, \epsilon) = \mathcal{O}\left(\frac{\log(n)}{n} + \epsilon\right).$$
- To simplify further, let us work with the asymptotic equivalence (for large n, big data)
$$\mathcal{E}(n, \epsilon) \sim \frac{1}{n} + \epsilon$$

Work complexity for large-scale learning

$$\mathcal{E}(n,\epsilon) \sim \frac{1}{n} + \epsilon$$

- For SGD, achieve ϵ-optimality with a computing time of $\mathcal{T}_{stoch} \sim 1/\epsilon$.
- Within the time budget $\mathcal{T}_{\max}$, the accuracy achieved is proportional to $1/\mathcal{T}_{\max}$, regardless of n.
- To minimize the error $\mathcal{E}(n,\epsilon)$, simply choose n as large as possible.
- Since the max number of examples that can be processed by SG is proportional to $\mathcal{T}_{\max}$, so the optimal error is proportional $1/\mathcal{T}_{\max}$

- For GD, achieve ϵ-optimality with a computing time of $\mathcal{T}_{batch} \sim n\log(1/\epsilon)$.
- Within the time budget $\mathcal{T}_{\max}$, to achieve ϵ-accuracy, need to process $n \sim \mathcal{T}_{\max}/\log(1/\epsilon)$ examples.
- Optimal error is not necessarily achieved by choosing n as large as possible. But rather by choosing ϵ to minimize the $\mathcal{E}(n,\epsilon) = \log(1/\epsilon)/\mathcal{T}_{\max} + \epsilon$.
- Optimal $\epsilon \sim 1/\mathcal{T}_{\max}$, so that optimal error is

$$\log(\mathcal{T}_{\max})/\mathcal{T}_{\max} + 1/\mathcal{T}_{\max}$$

Batch or Stochastic?

	Batch	Stochastic
$\mathcal{T}(n, \epsilon)$	$\sim n \log\left(\frac{1}{\epsilon}\right)$	$\frac{1}{\epsilon}$
$\mathcal{E}^*$	$\sim \frac{\log(\mathcal{T}_{\max})}{\mathcal{T}_{\max}} + \frac{1}{\mathcal{T}_{\max}}$	$\frac{1}{\mathcal{T}_{\max}}$

Comments

- Fragility of the asymptotic performance of SG.
- SG and ill-conditioning.
- Opportunities for distributed computing.
- Alternatives with faster convergence.

Outline

Noise Reduction Methods I (optional)

What if choosing $\alpha^k = \alpha$, must reduce the noise in sampled gradient at a geometric rate

Dynamic Sample Size Methods

- $w^{k+1} \leftarrow w^k - \alpha \frac{1}{|\mathcal{S}_k|} \sum_{j \in \mathcal{S}_k} \nabla J_j(w^k)$
- $|\mathcal{S}_k| = \lceil \tau^{k-1} \rceil$ with $\tau > 1$.

Noise Reduction Methods II (optional)

Gradient Aggregation

- SVRG

$$\nabla J_j(\tilde{w}^k) \leftarrow \nabla J_j(\tilde{w}^k) - [\nabla J_j(w^k) - \nabla J(w^k)]$$
$$\tilde{w}^{k+1} \leftarrow \tilde{w}^k - \alpha \nabla J_j(\tilde{w}^k)$$

- SAGA

t chosen randomly $\in \{k-n, k-n+1, ..., k\}$

$$\nabla J_j(w^k) \leftarrow \nabla J_j(w^k) - \nabla J_j(w^{[t]}) + \frac{1}{n}\sum_{i=1}^{n} \nabla J_j(w^{[i]})$$
$$w^{k+1} \leftarrow w^k - \alpha \nabla J_j(w^k)$$

Noise Reduction Methods III (optional)

Iterate Averaging Methods

$$w^{k+1} \leftarrow w^k - \alpha^k \nabla J_j(w^k)$$
$$\tilde{w}^{k+1} \leftarrow \frac{1}{k+1} \sum_{i=1}^{k+1} w^i$$

- $\alpha^k \sim O(1/k)$ or slower
- $\tilde{w}^k$ is *not* used for iterate update

Learning Algorithms