

CS480/680: Introduction to Machine Learning

Lecture 8: Gradient Descent

Hongyang Zhang



UNIVERSITY OF
WATERLOO

June 5&7, 2023

Optimization in Machine Learning

Many ML methods can be formulated as an optimization problem. Examples:

- Perceptron (Lecture 2):

$$\min_{\mathbf{w}} -\frac{1}{n} \sum_{i=1}^n y_i \langle \mathbf{w}, \mathbf{x}_i \rangle \mathbb{I}[\text{mistake on } \mathbf{x}_i]$$

- Logistic regression (Lecture 4):

$$\min_{\mathbf{w}} \sum_{i=1}^n \log[1 + \exp(-y_i \langle \mathbf{x}_i, \mathbf{w} \rangle)]$$

- SVM (Lecture 6):

$$\min_{\mathbf{w}, b} \frac{1}{2\lambda} \|\mathbf{w}\|_2^2 + \frac{1}{n} \sum_{i=1}^n (1 - y_i(\langle \mathbf{x}_i, \mathbf{w} \rangle + b))^+$$

Gradient Descent

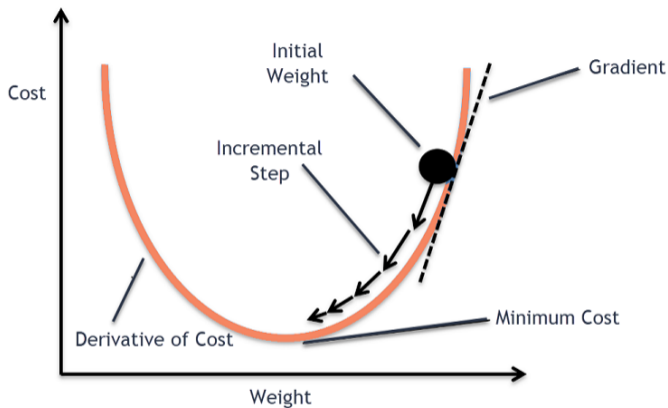
- Consider unconstrained optimization

$$\min_x f(x)$$

- ▶ Let's assume f is differentiable with gradient $\nabla f(x)$
- ▶ Denote optimal criterion value by $f^* = \min_x f(x)$, and a solution by x^*
- **Gradient descent** template: choose initial point $x^{(0)} \in \mathbb{R}^n$ and repeat

$$x^{(k)} = x^{(k-1)} - \underbrace{t}_{\text{step size}} \cdot \nabla f(x^{(k-1)}), \quad k = 1, 2, \dots$$

Gradient Descent



Intuition: Negative gradient is the steepest decreasing direction at that point. So if the step size is small and the function is convex, the algorithm will reach the minimizer.

An Example on Perceptron (Lecture 2)

$$\min_{\mathbf{w}} -\frac{1}{n} \sum_{i=1}^n y_i \langle \mathbf{w}, \mathbf{x}_i \rangle \mathbb{I}[\text{mistake on } \mathbf{x}_i]$$

- Gradient descent update:

$$\mathbf{w} \leftarrow \mathbf{w} + t \left[\frac{1}{n} \sum_{i=1}^n y_i \mathbf{x}_i \mathbb{I}[\text{mistake on } \mathbf{x}_i] \right]$$

- (Stochastic) Gradient descent update:

$$\mathbf{w} \leftarrow \mathbf{w} + t y_I \mathbf{x}_I \mathbb{I}[\text{mistake on } \mathbf{x}_I]$$

for a random index I

An Example on Soft-Margin SVM (Lecture 6)

$$\min_{\mathbf{w}, b} \frac{1}{2\lambda} \|\mathbf{w}\|_2^2 + \frac{1}{n} \sum_{i=1}^n \ell_{\text{hinge}}(y_i \hat{y}_i), \quad \text{s.t.} \quad \hat{y}_i = \langle \mathbf{x}_i, \mathbf{w} \rangle + b$$

- Gradient descent update:

$$\mathbf{w} \leftarrow \mathbf{w} - t \left[\frac{\mathbf{w}}{\lambda} + \frac{1}{n} \sum_{i=1}^n \ell'_{\text{hinge}}(y_i \hat{y}_i) y_i \mathbf{x}_i \right]$$

$$b \leftarrow b - t \left[\frac{1}{n} \sum_{i=1}^n \ell'_{\text{hinge}}(y_i \hat{y}_i) y_i \right]$$

Interpretation from Taylor Expansion

Consider the Taylor expansion of f **locally at x** , where x is the current iterate:

$$f(y) \approx f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t} \|y - x\|_2^2$$

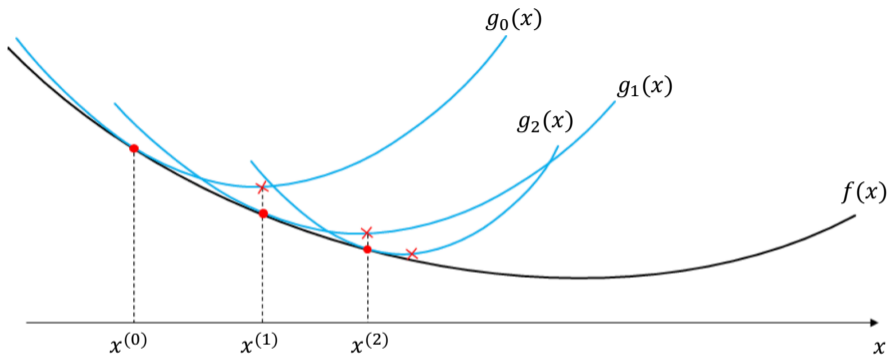
Taking the \min_y operation at both sides:

$$\min_y f(y) \approx \min_y \left[f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t} \|y - x\|_2^2 \right]$$

Choose next point $y = x^+$ to minimize the right hand side:

$$x^+ = x - t \nabla f(x)$$

Interpretation from Taylor Expansion

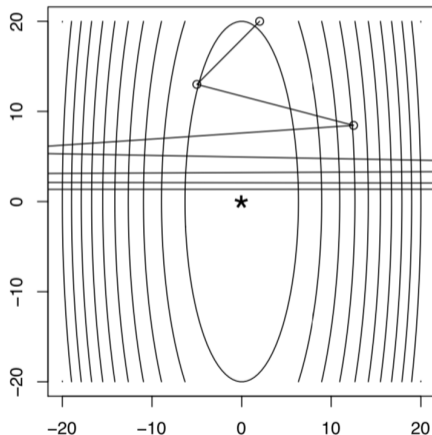


Circle point \bullet is x , cross point \times is

$$x^{(i+1)} = \operatorname{argmin}_y \underbrace{f(x^{(i)}) + \nabla f(x^{(i)})^T (y - x) + \frac{1}{2t} \|y - x^{(i)}\|_2^2}_{g_i(y)}$$

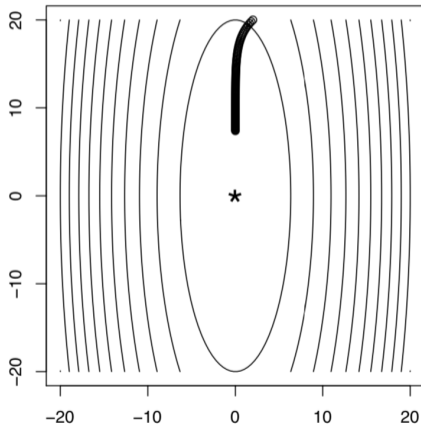
Step size cannot be too large

- **Diverge** if t is too large.
- Consider $f(x) = (10x_1^2 + x_2^2)/2$. Gradient descent after 8 steps:



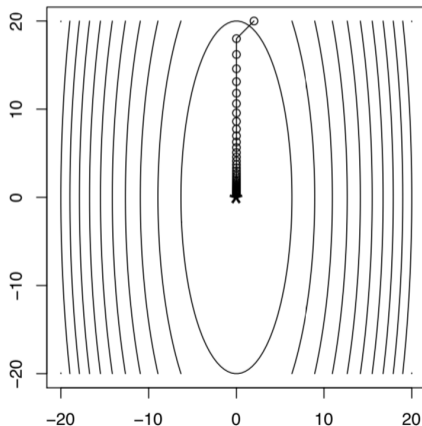
Step size cannot be too small

- Can be too **slow** if t is too small.
- Consider $f(x) = (10x_1^2 + x_2^2)/2$. Gradient descent:

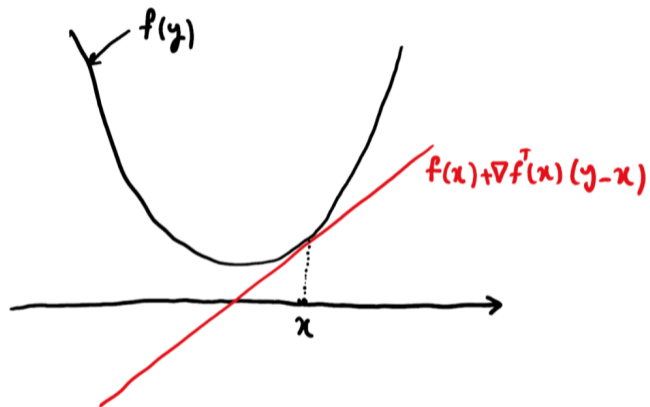


“Just right” step size

- Converge nicely when t is “just right”.
- Consider $f(x) = (10x_1^2 + x_2^2)/2$. Gradient descent after 40 steps:



Convex Function



Function f is convex: For any $x, y \in \mathbb{R}^n$,

$$f(y) \geq f(x) + \nabla f(x)^T (y - x)$$

Convergence Analysis for Convex Case

Assume that f is convex and differentiable, with $\text{dom}(f) = \mathbb{R}^n$, and additionally that ∇f is L -Lipschitz continuous (a.k.a. f is L -smooth):

$$L\mathbf{I} - \nabla^2 f(x)$$

is positive semi-definite (denoted by $L\mathbf{I} \succeq \nabla^2 f(x)$).

Theorem: Convergence rate for convex case

Gradient descent with fixed step size $t \leq 1/L$ satisfies

$$f(x^{(k)}) - f^* \leq \frac{\|x^{(0)} - x^*\|_2^2}{2tk}.$$

We say gradient descent has convergence rate $O(1/k)$. That is, it finds ϵ -suboptimal point in $\Omega(1/\epsilon)$ iterations.

Proof

For any y , perform a quadratic expansion and obtain:

$$\begin{aligned} f(y) &\leq f(x) + \nabla f(x)^T (y - x) + \frac{1}{2} (y - x)^T \nabla^2 f(x) (y - x) \\ &\leq f(x) + \nabla f(x)^T (y - x) + \frac{1}{2} L \|y - x\|_2^2 \quad (\text{because } L\mathbf{I} \succeq \nabla^2 f(x)) \end{aligned}$$

Plug in $y = x^+ = x - t\nabla f(x)$:

$$\begin{aligned} f(x^+) &\leq f(x) + \nabla f(x)^T (x^+ - x) + \frac{1}{2} L \|x^+ - x\|_2^2 \\ &= f(x) + \nabla f(x)^T (x - t\nabla f(x) - x) + \frac{1}{2} L \|x - t\nabla f(x) - x\|_2^2 \\ &= f(x) - \left(1 - \frac{1}{2} Lt\right) t \|\nabla f(x)\|_2^2 \\ &\leq f(x) - \frac{1}{2} t \|\nabla f(x)\|_2^2 \quad (\text{because } t \leq 1/L) \end{aligned} \tag{1}$$

That is, each update decreases the function value by at least $\frac{1}{2} t \|\nabla f(x)\|_2^2$!

Proof — Cont'

Function f is convex:

$$f(x^*) \geq f(x) + \nabla f(x)^T(x^* - x) \Rightarrow f(x) \leq f(x^*) + \nabla f(x)^T(x - x^*)$$

Plugging in (1), we obtain:

$$\begin{aligned} f(x^+) &\leq f(x^*) + \nabla f(x)^T(x - x^*) - \frac{t}{2} \|\nabla f(x)\|_2^2 \\ \Rightarrow f(x^+) - f(x^*) &\leq \frac{1}{2t} (2t \nabla f(x)^T(x - x^*) - t^2 \|\nabla f(x)\|_2^2) \\ \Rightarrow f(x^+) - f(x^*) &\leq \frac{1}{2t} (2t \nabla f(x)^T(x - x^*) - t^2 \|\nabla f(x)\|_2^2 - \|x - x^*\|_2^2 + \|x - x^*\|_2^2) \\ \Rightarrow f(x^+) - f(x^*) &\leq \frac{1}{2t} (\|x - x^*\|_2^2 - \|x - t \nabla f(x) - x^*\|_2^2) \\ \Rightarrow f(x^+) - f(x^*) &\leq \frac{1}{2t} (\|x - x^*\|_2^2 - \|x^+ - x^*\|_2^2) \end{aligned}$$

Proof — Cont'

Summing over iterations:

$$\begin{aligned}\sum_{i=1}^k (f(x^{(i)}) - f(x^*)) &\leq \sum_{i=1}^k \frac{1}{2t} (\|x^{(i-1)} - x^*\|_2^2 - \|x^{(i)} - x^*\|_2^2) \\ &= \frac{1}{2t} (\|x^{(0)} - x^*\|_2^2 - \|x^{(k)} - x^*\|_2^2) \\ &\leq \frac{1}{2t} \|x^{(0)} - x^*\|_2^2,\end{aligned}$$

which implies

$$f(x^{(k)}) \leq \frac{1}{k} \sum_{i=1}^k f(x^{(i)}) \leq f(x^*) + \frac{\|x^{(0)} - x^*\|_2^2}{2tk}.$$

The first inequality holds because $f(x^{(i)})$ is **decreasing** with the increase of i . Q.E.D.

Convergence Analysis for Strong Convexity

m-strong convexity of f means $f(x) - m\|x\|_2^2$ is convex: $L\mathbf{I} \succeq \nabla^2 f(x) \succeq m\mathbf{I}$.

Theorem: Convergence rate for strong convexity

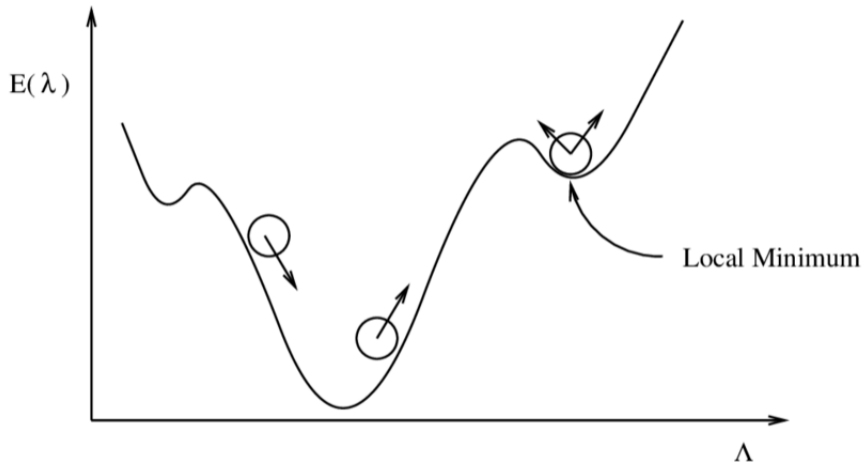
Let f be m -strongly convex and L -smooth. Gradient descent with fixed step size $t \leq 2/(m + L)$ satisfies

$$f(x^{(k)}) - f^* \leq \gamma^k \frac{L}{2} \|x^{(0)} - x^*\|_2^2,$$

where $0 < \gamma < 1$.

Rate under strong convexity is $O(\gamma^k)$, exponentially fast! That is, it finds ϵ -suboptimal point (i.e., $f(x^{(k)}) - f^* < \epsilon$) in $\Omega(\log(1/\epsilon))$ iterations.

Gradient descent for Nonconvex Case



Asking for optimality is too much. Let's focus on $\|\nabla f(x)\|_2 \leq \epsilon$.

Convergence Analysis for Nonconvex Case

Assume f is differentiable with L -Lipschitz gradient, now **nonconvex**.

Theorem: Convergence rate for nonconvex case

Gradient descent with fixed step size $t \leq 1/L$ satisfies

$$\min_{i=0,\dots,k} \|\nabla f(x^{(i)})\|_2 \leq \sqrt{\frac{2(f(x^{(0)}) - f^*)}{t(k+1)}}$$

Thus gradient descent has rate $O(1/\sqrt{k})$, even in the nonconvex case for finding stationary points.

This rate **cannot be improved** by any deterministic algorithm.

Stochastic Gradient Descent

- Consider decomposable optimization (n is very large)

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

- ▶ For example, $f_i(w) = \ell(w; x_i, y_i)$
- ▶ Let's assume f_i is differentiable with gradient $\nabla f_i(w)$

- Gradient descent:

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

- Stochastic gradient descent: (the same expectation)

$$w^+ = w - t \cdot \nabla f_I(w)$$

where I is a random index

Stochastic Gradient Descent — Convergence Rate

For convex and L smooth f_i (in the k -th iteration):

- Gradient descent:

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

- ▶ Step size $t \leq 1/L$
- ▶ Computational complexity: order of $\frac{n}{\epsilon}$

- Stochastic gradient descent:

$$w^+ = w - t \cdot \nabla f_I(w)$$

where I is a random index

- ▶ Step size $t = 1/k$ for $k = 1, 2, 3, \dots$
- ▶ Computational complexity: order of $\frac{1}{\epsilon^2}$
- ▶ Randomness leads to large variance of estimation of gradient. Thus SGD requires more iterations (though each iteration needs less computations)

Questions

?

?

Answers

?