Optimization for Data Science Lecture 05: Convergence of Gradient Descent

Kimon Fountoulakis

School of Computer Science
University of Waterloo

24/09/2019 and 26/09/2019

Previous lecture

- We assumed that
  - The objective function f is differentiable
  - and its gradient $\nabla f(x)$ is Lipschitz continuous

$$\|\nabla f(z) - \nabla f(s)\|_2 \leq L\|z - s\|_2 \ \forall z, s$$

- Lipschitz continuity of the gradient implies that the gradient cannot change arbitrarily fast.
- Lipschitz continuity of the gradient is a common assumption in Machine Learning problems.
- For example, least-squares logistic regression, deep neural networks.

Previous lecture

- We defined gradient descent as the following iterative scheme:

$$x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)$$

- where k is the number of iteration and L is the Lipschitz constant of the gradient.

- We proved that at each iteration gradient descent decreases the objective function

$$f(x_{k+1}) < f(x_k)$$

Previous lecture

- More generally we defined the gradient descent using step-sizes α_k:

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$

- where α_k was chosen using line-search techniques.

- We proved that at each iteration gradient descent + line-search decreases the objective function

$$f(x_{k+1}) < f(x_k)$$

Previous lecture

- We also proved that if a function f is differentiable and its gradient $\nabla f(x)$ is Lipschitz continuous, then we can upper bound f:

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{L}{2} \|y - x\|_2^2 \quad \forall x, y \in \mathbb{R}^n$$

- **We are going to use this upper bound in this lecture a lot.**

Outline

- Convergence of gradient descent
- Convergence rate of gradient descent for non-convex and convex functions

A simplification

- In this lecture I will assume that we always work with the following version of gradient descent:

$$x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)$$

- which uses constant step-sizes $\alpha_k = 1/L \ \forall k$

- This simplifies the analysis, also, similar results can be shown for gradient descent + line-search.

Amount of decrease of the objective function

- If a function f is differentiable and its gradient $\nabla f(x)$ is Lipschitz continuous, then gradient descent satisfies

$$f(x_{k+1}) \leq f(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|_2^2$$

- This result shows that gradient descent is guaranteed to decreases the objective function

- The amount of decrease depends on the length of the gradient.

Asymptotic convergence

- We can show that as $k \to \infty$
- then $f(x_k) - f(x_{k+1}) \to 0$
- which implies that $\|\nabla f(x_k)\|_2 \to 0$

Asymptotic convergence: sketch of proof

- Assuming that the function f is bounded below:

$$f^* \leq f(x) \ \forall x \in \mathbb{R}^n$$

- (we have to assume this, otherwise we are minimizing unbounded functions)

- From the “amount of decrease inequality” we get

$$\|\nabla f(x_k)\|_2^2 \leq 2L(f(x_k) - f(x_{k+1}))$$

Asymptotic convergence: sketch of proof

- Because gradient descent monotonically decreases the objective function

$$f(x_{k+1}) < f(x_k)$$

- and the objective function is bounded below, then we must have that $f(x_k) - f(x_{k+1}) \to 0$ as $k \to \infty$

- which in combination with $\|\nabla f(x_k)\|_2^2 \leq 2L(f(x_k) - f(x_{k+1}))$

- implies that $\|\nabla f(x_k)\|_2 \to 0$ as $k \to \infty$.

Asymptotic convergence

- However, the asymptotic convergence results does not tell us about:
  - How fast the gradient goes to zero.
- Since the termination criterion of gradient descent is $\|\nabla f(x_k)\|_2 \leq \epsilon$, for some positive tolerance parameter ϵ, we would like to know how many iteration will be required by gradient descent to satisfy the termination criterion.

Asymptotic convergence

- In other words, given a tolerance parameter $\epsilon > 0$, we would like to know how many iterations does it take to get to $\|\nabla f(x_k)\|_2 \leq \epsilon$.

Convergence rate: assumptions

- Function f is differentiable and its gradient $\nabla f(x)$ is Lipschitz continuous.

- Function f is bounded below:

$$f^* \leq f(x) \ \forall x \in \mathbb{R}^n$$

Convergence rate

- After t iterations (start counting from zero), gradient descent satisfies

$$\min_{0 \le k \le t} \|\nabla f(x_k)\|_2^2 \le \frac{2L(f(x_0) - f^*)}{t+1}$$

- Thus after t iterations we have that gradient descent produces at least one x_k such that

$$\|\nabla f(x_k)\|_2^2 = \mathcal{O}\left(\frac{1}{t}\right)$$

Convergence rate

- After t iterations we have that gradient descent produces at least one x_k such that

$$\|\nabla f(x_k)\|_2^2 = \mathcal{O}\left(\frac{1}{t}\right)$$

- We say that $||\nabla f(x_k)||_2^2$ converges **sub-linearly**. Why? See next slide.

Convergence rate

- $-\log_{10} \|\nabla f(x_k)\|_2^2$ is a measure of the number of correct significant digits in $\|\nabla f(x_k)\|_2^2$.

- For example: $-\log_{10} 0.1 = 1$, $-\log_{10} 0.01 = 2$, $-\log_{10} 0.001 = 3$.

- We have that $-\log_{10} \|\nabla f(x_k)\|_2^2 \approx \log_{10} t$. Thus the number of correct digits scales logarithmically with t. The logarithm is a smaller function than the linear function, thus we call the $\mathcal{O}(1/t)$ rate sub-linear.

Iteration complexity

- How many iterations does it take to satisfy

$$\|\nabla f(x_k)\|_2^2 \leq \epsilon$$

- Gradient descent requires in **worst-case**

$$t \geq \frac{2L(f(x_0) - f^*)}{\epsilon}$$

- to produce an x_k that satisfies $\|\nabla f(x_k)\|_2^2 \leq \epsilon$.

Iteration complexity

- A similar result can be shown when using line-search techniques to compute the step-size α_k. Only some constants change.
- The rate $\mathcal{O}(1/t)$ is dimension independent (assuming that the Lipschitz constant L does not depend on the dimensions of the problem).

Iteration complexity

- We showed that

$$\min_{0 \leq k \leq t} \|\nabla f(x_k)\|_2^2 \leq \frac{2L(f(x_0) - f^*)}{t+1}$$

- But it is not necessary that the only the last iteration t satisfies the above bound.

- Since this is a **worst-case** result, earlier iterations might satisfy this bound too.

Iteration complexity

- For Machine Learning problems bounds like

$$\min_{0 \leq k \leq t} \|\nabla f(x_k)\|_2^2 \leq \frac{2L(f(x_0) - f^*)}{t+1}$$

- are often **very pessimistic**. In practice, gradient descent might converge faster.

- This reveals a practice and theory gap.

Iteration complexity

- Since our function f is not necessarily convex, gradient descent is only guaranteed to converge to a stationary point, i.e., $\nabla f(x) = 0$.

Convergence rate for convex functions: assumptions

- Function f is differentiable and its gradient $\nabla f(x)$ is Lipschitz continuous.

- Function f is bounded below:

$$f^* \leq f(x) \; \forall x \in \mathbb{R}^n$$

- where f^* represents the minimum of f.

- Function f is convex:

$$f(x) \geq f(y) + \nabla f(y)^T (x - y) \; \forall x \in \mathbb{R}^n, y \in \mathbb{R}^n$$

Convergence rate for convex functions

- After t iterations (start counting from zero), gradient descent satisfies

$$f(x_t) - f^* \leq \frac{2L\|x_0 - x^*\|_2^2}{t+1}$$

- Thus after t iterations we have that gradient descent produces an x_t such that

$$f(x_t) - f^* = \mathcal{O}\left(\frac{1}{t}\right)$$

Convergence rate

- After t iterations we have that gradient descent produces x_t such that

$$f(x_t) - f^* = \mathcal{O}\left(\frac{1}{t}\right)$$

- We say that $f(x_k) - f^*$ converges **sub-linearly.**

Iteration complexity for convex functions

- How many iterations does it take to satisfy

$$f(x_k) - f^* \leq \epsilon$$

- Gradient descent requires in **worst-case**

$$t \geq \frac{2L\|x_0 - x^*\|_2^2}{\epsilon}$$

- iterations to satisfy $f(x_t) - f^* \leq \epsilon$.

Convergence rate: non-convex vs convex

- Non-convex functions

$$\min_{0 \le k \le t} \|\nabla f(x_k)\|_2^2 \le \frac{2L(f(x_0) - f^*)}{t+1}$$

- Convex functions

$$f(x_t) - f^* \le \frac{2L\|x_0 - x^*\|_2^2}{t+1}$$

- We cannot bound the "distance" $f(x_t) - f^*$ for non-convex functions. That's because f^* represents the global minimum for non-convex functions and gradient descent is only guaranteed to converge to a stationary point.

Convergence rate: non-convex vs convex

- Non-convex functions

$$\min_{0\le k\le t} \|\nabla f(x_k)\|_2^2 \le \frac{2L(f(x_0) - f^*)}{t+1}$$

- Convex functions

$$f(x_t) - f^* \le \frac{2L\|x_0 - x^*\|_2^2}{t+1}$$

- The bound for non-convex function holds for some x_k that is produced during execution of gradient descent during the first t iterations.

- The bound for convex functions holds for the last iteration t.

Convergence rate: non-convex vs convex

- Non-convex functions

$$\min_{0\le k\le t} \|\nabla f(x_k)\|_2^2 \le \frac{2L(f(x_0)-f^*)}{t+1}$$

- Convex functions

$$f(x_t) - f^* \le \frac{2L\|x_0 - x^*\|_2^2}{t+1}$$

- For convex functions we can convert the bound on $f(x_t) - f^*$ to a bound on $\|\nabla f(x_t)\|_2^2$ by using the inequality $f(x) - f^* \ge \frac{1}{2L}\|\nabla f(x)\|_2^2 \; \forall x$.

Strong convexity

- We say that a differentiable function “f” is strongly convex if

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{\mu}{2} \|y - x\|_2^2$$

- for any x and y and some positive constant $\mu > 0$.

Strong convexity

- For twice differentiable functions strong convexity is equivalent to assuming that

$$y^T \nabla^2 f(x) y \geq \mu \|y\|_2^2 \ \forall x, y \in \mathbb{R}^n$$

Strong convexity: unique minimizer

- Strong convexity implies that function f has a unique minimum.

Convergence rate for strongly convex functions: assumptions

- Function f is differentiable and its gradient $\nabla f(x)$ is Lipschitz continuous.

- Function f is bounded below:

$$f^* \leq f(x) \ \forall x \in \mathbb{R}^n$$

- Function f is μ-strongly convex:

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{\mu}{2} \|y - x\|_2^2$$

Convergence rate for strongly convex functions: assumptions

- After t iterations (start counting from zero), gradient descent satisfies

$$f(x_t) - f^* \leq (1 - \mu/L)^t \left(f(x_0) - f^*\right)$$

Convergence rate

- After t iterations we have that gradient descent produces an x_t such that

$$f(x_t) - f^* \leq (1 - \mu/L)^t \left(f(x_0) - f^*\right)$$

- We say that $f(x_k) - f^*$ converges **linearly**. Why? See next slide.

Convergence rate

- $-\log_{10}(f(x_k) - f^*)$ is a measure of the number of correct significant digits in $f(x_k)$.

- We have that $-\log_{10}(f(x_k) - f^*) \approx -t\log_{10}(1 - \mu/L)$. Thus the number of correct digits scales **linearly** with t.

Iteration complexity for strongly convex functions

- How many iterations does it take to satisfy

$$f(x_k) - f^* \leq \epsilon$$

- Gradient descent requires in **worst-case**

$$t = \mathcal{O}\left(\log \frac{1}{\epsilon}\right)$$

- iterations to satisfy $f(x_t) - f^* \leq \epsilon$.

Convergence rate: non-convex vs convex vs strongly convex

- Non-convex functions

$$\min_{0 \le k \le t} \|\nabla f(x_k)\|_2^2 \le \frac{2L(f(x_0) - f^*)}{t+1}$$

- Convex functions

$$f(x_t) - f^* \le \frac{2L\|x_0 - x^*\|_2^2}{t+1}$$

- Strongly convex functions

$$f(x_t) - f^* \le (1 - \mu/L)^t \, (f(x_0) - f^*)$$

Iteration complexity: non-convex vs convex vs strongly convex

- Non-convex functions (converges to stationary point)

$$t \geq \frac{2L\|x_0 - x^*\|_2^2}{\epsilon}$$

- Convex functions (converges to global minimizer)

$$t \geq \frac{2L\|x_0 - x^*\|_2^2}{\epsilon}$$

- Strongly convex functions (converges to global minimizer)

$$t = \mathcal{O}\left(\log \frac{1}{\epsilon}\right)$$