Gradient Descent

Sravan Danda*

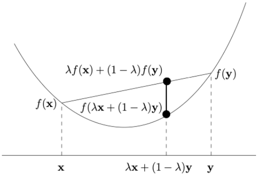
*CS&IS and APPCAIR, BITS-Pilani, Goa, India

Week 2

Outline

Characterizations of Convex Functions

Definition: Let $f : \mathbb{R}^d \to \mathbb{R}$ be a function. f is said to be **convex** when line segment joining any two points $(\boldsymbol{x}, f(\boldsymbol{x}))$ and $(\boldsymbol{y}, f(\boldsymbol{y}))$ on the graph lies on or above the graph of the function

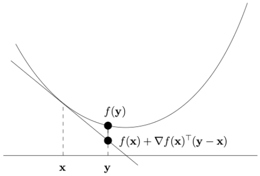


Screenshot Source [1]

[1] Optimization for Data Science, Lecture Notes, ETH, 2023

Characterizations of Convex Functions

First-order characterization: Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function. f is convex if and only if the tangent at any point on the graph lies below the graph.



Screenshot Source [2]

[2] Optimization for Data Science, Lecture Notes, ETH, 2023

Characterizations of Convex Functions[4]

Second-order characterization: Let $f : \mathbb{R}^d \to \mathbb{R}$ be a twice differentiable function. The *Hessian* (matrix of second partial derivatives) exists at every point and is symmetric. f is convex if and only if the Hessian is a positive semi-definite[3]

Example: If $f(x_1, x_2) = x_1^2 + x_2^2$ then the Hessian is given by

$$\nabla^2 f(x) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \tag{1}$$

which is a positive-semi-definite

[3] A matrix M is said to be a positive semi-definite if $\boldsymbol{x}^T M \boldsymbol{x} \geq 0$ for all $\boldsymbol{x}$

[4] characterizations help recognize convex functions in multiple ways. For a given function one way might be easier to check than the others.

Minimizing Convex Functions

Definition

A local minimum of $f : \mathbb{R}^d \to \mathbb{R}$ is a point $\boldsymbol{x}$ such that there exists $\epsilon > 0$ with

$$f(\boldsymbol{x}) \leq f(\boldsymbol{y}) \ \forall \boldsymbol{y} \ \textit{satisfying} \ ||\boldsymbol{y} - \boldsymbol{x}|| < \epsilon \tag{2}$$

Local and Global Minima of Convex Functions

Lemma

Let $\boldsymbol{x}^*$ be a local minima of a convex function $f : \mathbb{R}^d \to \mathbb{R}$ then $\boldsymbol{x}^*$ is a global minima of f i.e.

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{y}) \ \forall \boldsymbol{y} \in \mathbb{R}^d \tag{3}$$

If a convex function happens to have a local minima then it has to be a global minima. Note that there are convex functions that are unbounded below. These functions do not have any minima

Global Minima of Differentiable Convex Functions

Lemma

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable convex function. Let $\mathbf{x}^*$ be such that $\nabla f(\mathbf{x}^*) = 0$ then $\mathbf{x}^*$ is a global minima of f.

Goal: Iterative Approach to Approximate Global Minima

The set-up: Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex and differentiable function. Assume f has a global minimum $\boldsymbol{x}^*$. The goal is to find **a** x such that

$$f(\boldsymbol{x}) - f(\boldsymbol{x}^*) < \epsilon \tag{4}$$

for some given $\epsilon > 0$

Gradient Descent and Variants

Broadly, all the iterative methods start with an initialization $\boldsymbol{x}_0$ and obtain a sequence $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_T$ until

$$f(\boldsymbol{x}_T) - f(\boldsymbol{x}^*) < \epsilon \tag{5}$$

Convergence Rates[6]

Convention: Measure the relative error terms[5] as a function of the iteration and recursively track how fast it reduces. Let

$$\epsilon_t = \frac{f(\boldsymbol{x}_t) - f(\boldsymbol{x}^*)}{f(\boldsymbol{x}_0) - f(\boldsymbol{x}^*)} \tag{6}$$

Linear Convergence: $\exists t \geq T$ such that

$$\epsilon_{t+1} \leq c\epsilon_t \tag{7}$$

for some constant $0 < c < 1$

Practical implication: We require $\mathcal{O}(log\left(\frac{1}{\epsilon}\right))$ iterations to obtain an approximate solution

[5] this quantity may be hard to obtain if $f(\boldsymbol{x}^*)$ is unknown

[6] help determining how many iterations are needed to obtain an approximate solution if we know how good the initial estimate is

Convergence Rates

Superlinear Convergence[7]: $\exists t \geq T$ such that

$$\epsilon_{t+1} \leq c(\epsilon_t)^r \tag{8}$$

for some constant $0 < c < 1$ and for some $r > 1$

Practical implication: We require $\mathcal{O}(\frac{log(log(\frac{1}{\epsilon}))}{log r})$ iterations to obtain an approximate solution

[7] $r = 2$ corresponds to quadratic convergence

Convergence Rates: Loose Upper Bounds

	Lipschitz convex functions	smooth convex functions	strongly convex functions	smooth & strongly convex functions
gradient descent	Thm. 3.1 $\mathcal{O}(1/\varepsilon^2)$	Thm. 3.8 $\mathcal{O}(1/\varepsilon)$		Thm. 3.14 $\mathcal{O}(\log(1/\varepsilon))$
accelerated gradient descent		Thm. 3.9 $\mathcal{O}(1/\sqrt{\varepsilon})$		
projected gradient descent	Thm. 4.2 $\mathcal{O}(1/\varepsilon^2)$	Thm. 4.4 $\mathcal{O}(1/\varepsilon)$		Thm. 4.5 $\mathcal{O}(\log(1/\varepsilon))$
subgradient descent	Thm. 10.20 $\mathcal{O}(1/\varepsilon^2)$		Thm. 10.22 $\mathcal{O}(1/\varepsilon)$	
stochastic gradient descent	Thm. 12.4 $\mathcal{O}(1/\varepsilon^2)$		Thm. 12.4 $\mathcal{O}(1/\varepsilon)$	

Table 3.1: Results on gradient descent. Below each theorem, the number of steps is given which the respective variant needs on the respective function class to achieve additive approximation error at most ε.

Screenshot Source [8]

[8] Optimization for Data Science, Lecture Notes, ETH, 2023

Gradient Descent: Algorithm

Idea: Make a small step $\boldsymbol{v}_t$ in the opposite direction of the gradient to ensure descent on the value of the function i.e. $f(\boldsymbol{x}_t + \boldsymbol{v}_t) < f(\boldsymbol{x}_t)$

$$f(\boldsymbol{x}_t + \boldsymbol{v}_t) = f(\boldsymbol{x}_t) + \nabla f(\boldsymbol{x}_t)^T \boldsymbol{v}_t + o(||\boldsymbol{v}_t||) \tag{9}$$

With a fixed stepsize[9] $\gamma > 0$, we have

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \gamma \nabla f(\boldsymbol{x}_t) \tag{10}$$

[9] has to be small enough. If γ is too small the convergence will take forever and if it is large there will be fluctuations and the function may not strictly decrease as we iterate

Vanilla Analysis

Recall the first-order characterization: *the tangent at any point is below the graph of the function*

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) \tag{11}$$

Set $\boldsymbol{x} = \boldsymbol{x}_t$, $\boldsymbol{y} = \boldsymbol{x}^*$ and rearranging we can upper-bound the error in terms of the gradient!

$$f(\boldsymbol{x}_t) - f(\boldsymbol{x}^*) \leq \nabla f(\boldsymbol{x}_t)^T(\boldsymbol{x}_t - \boldsymbol{x}^*) \tag{12}$$

Vanilla Analysis Continued[11] . . .

For a fixed step-size i.e. when

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \gamma \nabla f(\mathbf{x}_t) \tag{13}$$

the sum of errors can be bounded by

$$\sum_{t=0}^{T-1} (f(\mathbf{x}_t) - f(\mathbf{x}^*)) \leq \frac{\gamma}{2} \sum_{t=0}^{T-1} ||\nabla f(\mathbf{x}_t)||^2 + \frac{1}{2\gamma} ||\mathbf{x}_0 - \mathbf{x}^*||^2 \tag{14}$$

smaller the squared gradients[10] and closer the initialization, the better!

---

[10] Lipschitz functions have bounded gradients. Can we provide some kind of guarantees for such subclass of convex functions?

[11] see ETH 2023 lecture notes for a complete proof

Analysis: Lipschitz Convex Functions

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable function with a global minimum $\mathbf{x}^*$. Furthermore suppose that $||\mathbf{x}_0 - \mathbf{x}^*|| < R$ and $||\nabla f(\mathbf{x})|| \leq B$ for all $\mathbf{x}$ then choosing the step size

$$\gamma = \frac{R}{B\sqrt{T}} \tag{15}$$

gradient descent yields

$$\frac{1}{T}\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}^*)) \leq \frac{RB}{\sqrt{T}} \tag{16}$$

Analysis: Lipschitz Convex Functions

Implications:

1. The average error is bounded as a function of the distance of initialization from the global minima, maximum norm of gradient and the number of iterations provided the step size is carefully chosen.
2. A loose upper bound[12] on T so that the minimum error within the first T iterations is less than ϵ is given by $\frac{R^2B^2}{\epsilon^2}$
3. The choice of step-size and the number of iterations for such guarantees depends on knowledge of initialization and an upper bound on the norm of the gradient

[12] this is a bad bound as the growth is quadratic in $\frac{1}{\epsilon}$ even if the initialization and Lipschitz constant are low

L-Smooth Functions[13]

Definition

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function and $L \in \mathbb{R}^+$. A function f is called L-**smooth** (i.e. parameter L) if

$$f(\boldsymbol{y}) \leq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) + \frac{L}{2}||\boldsymbol{x} - \boldsymbol{y}||^2 \ \forall \ \boldsymbol{x}, \boldsymbol{y} \tag{17}$$

This is intuitively imposing a restriction on how fast the function is allowed to grow!

[13] NOT to be confused with infinitely differentiable functions

Visualizing L-Smooth Convex Functions

If a function is both smooth and convex then the graph is lower bounded by the tangent and upper-bounded by the addition of quadratic term to the tangent. L controls the growth of the gradient norm

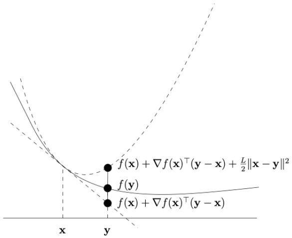


Figure 3.2: A smooth convex function

Screenshot Source [14]

[14] Optimization for Data Science, Lecture Notes, ETH, 2023

Characterizing L-Smooth Functions

Intuition: The curvature of a L-smooth function does not exceed L

Lemma

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function then the following statements are equivalent

1. *f is L-Smooth.*
2. *$g : \mathbb{R}^d \to \mathbb{R}$ defined by*

$$g(x) = \frac{L}{2}\boldsymbol{x}^T\boldsymbol{x} - f(\boldsymbol{x}) \tag{18}$$

is convex.

Quadratic Forms

Intuition: All quadratic forms are smooth as the highest degree term is a squared term and for large enough L, $\frac{L}{2}\boldsymbol{x}^T\boldsymbol{x}$ would have a higher curvature than that of the quadratic form

Lemma

Let $f : \mathbb{R}^d \to \mathbb{R}$ be defined as

$$f(\boldsymbol{x}) = \boldsymbol{x}^T Q\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{x} + c \tag{19}$$

where Q is a symmetric matrix, $\boldsymbol{b} \in \mathbb{R}^d$ and $c \in \mathbb{R}$. Then f is smooth with parameter $2||Q||$ where $||Q||$ refers to the spectral norm of Q.

L-Smooth Functions and Lipschitz

Intuition: Restricting the growth of the curvature is effectively imposing an upper bound on the norm of the second derivative!

Lemma

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable function. The following statements are equivalent

1. *f is L-Smooth.*
2. *The gradient of f is L-Lipschitz i.e.* $||\nabla f(\boldsymbol{y}) - \nabla f(\boldsymbol{x})|| \leq L||y - x||$ *for all* $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$

Choosing Step-Size for L-Smooth Functions

Intuition: With an appropriate step-size true descent is guaranteed[15]!

Lemma

Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable function and L-smooth. With the step-size

$$\gamma = \frac{1}{L} \tag{20}$$

the gradient descent satisfies

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \frac{1}{2L}||\nabla f(\mathbf{x}_t)||^2, \; t \geq 0 \tag{21}$$

[15] progress is guaranteed although the function may be flat. L-smooth only provides an upper bound on steepness and NOT a lower bound

Convergence Rate for L-Smooth Convex Functions

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex, differentiable function and has a global minimum $\boldsymbol{x}^*$. Furthermore assume f is L-smooth. With the step-size

$$\gamma = \frac{1}{L} \tag{22}$$

the gradient descent yields

$$f(\boldsymbol{x}_T) - f(\boldsymbol{x}^*) \leq \frac{L}{2T}||\boldsymbol{x}_0 - \boldsymbol{x}^*||^2, \;\; T > 0 \tag{23}$$

Implication: If $T > \frac{R^2 L}{2\epsilon}$, the approximation would be within ϵ

How to Find Parameter of a Smooth Function?

Intuition:

1. The norm of the Hessian i.e. the largest eigenvalue of the Hessian would be the smoothness parameter
2. How to compute the largest Eigenvalue efficiently[16]?
   1. Lanczos algorithm: $https://en.wikipedia.org/wiki/Lanczos\_algorithm$

16 memory is quadratic in parameters if the Hessian is computed explicitly

Acceleration for Smooth Convex Functions

Definition

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex, differentiable function and has a global minimum $\boldsymbol{x}^*$. Furthermore assume f is L-smooth. *Accelerated gradient descent* is the following algorithm: choose $\boldsymbol{z}_0 = \boldsymbol{y}_0 = \boldsymbol{x}_0$ arbitrary. For $t \geq 0$, set

$$\boldsymbol{y}_{t+1} = \boldsymbol{x}_t - \frac{1}{L}\nabla f(\boldsymbol{x}_t) \tag{24}$$

$$\boldsymbol{z}_{t+1} = \boldsymbol{z}_t - \frac{t+1}{2L}\nabla f(\boldsymbol{x}_t) \tag{25}$$

$$\boldsymbol{x}_{t+1} = \frac{t+1}{t+3}\boldsymbol{y}_{t+1} + \frac{2}{t+3}\boldsymbol{z}_{t+1} \tag{26}$$

Acceleration for Smooth Convex Functions

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex, differentiable function and has a global minimum $\boldsymbol{x}^*$. Furthermore assume f is L-smooth. Accelerated gradient descent yields

$$f(\boldsymbol{y}_T) - f(\boldsymbol{x}^*) \leq \frac{2L}{T(T+1)}||\boldsymbol{z}_0 - \boldsymbol{x}^*||^2, \ T > 0 \tag{27}$$

Implication: The rate of convergence is $\mathcal{O}\left(\frac{1}{\sqrt{\epsilon}}\right)$

Strongly Convex Functions

Intuition: Convex with a minimum level of the steepness allows for faster convergence

Definition

Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be a differentiable function and $\mu \in \mathbb{R}^+$. A function f is called strongly convex with parameter μ if

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) + \frac{\mu}{2}||\boldsymbol{x} - \boldsymbol{y}||^2 \ \forall\ \boldsymbol{x}, \boldsymbol{y} \tag{28}$$

Note: Strongly convex functions form a subclass of the convex functions as the first order characterization is automatically satisfied by obtaining a lower bound (removing the last term on RHS)

Visualizing Strongly Convex Functions

The steepness is at least as much as that of a quadratic scaled by μ

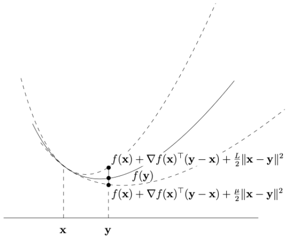


Figure 3.3: A smooth and strongly convex function

Screenshot Source [17]

[17] Optimization for Data Science, Lecture Notes, ETH, 2023

Characterizing Strongly Convex Functions

Intuition: The curvature of a Strongly convex function exceeds at least μ

Lemma

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function then the following statements are equivalent

1. *f is strongly convex with parameter μ.*
2. *$g : \mathbb{R}^d \to \mathbb{R}$ defined by*

$$g(x) = f(\boldsymbol{x}) - \frac{\mu}{2}\boldsymbol{x}^T\boldsymbol{x} \tag{29}$$

   is convex.

Convergence Rate for Smooth and Strongly Convex Functions

Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex, differentiable function. Furthermore assume f is L-smooth and strongly convex with parameter μ. Then f has a unique global minimum $\boldsymbol{x}^*$ and choosing the step-size $\gamma = \frac{1}{L}$ the gradient descent with arbitrary $\boldsymbol{x}_0$ satisfies the following:
1) Squared distances to $\boldsymbol{x}^*$ are geometrically decreasing:

$$||\boldsymbol{x}_{t+1} - \boldsymbol{x}^*||^2 \leq \left(1 - \frac{\mu}{L}\right) ||\boldsymbol{x}_t - \boldsymbol{x}^*||^2, \ \ t \geq 0 \tag{30}$$

2) The absolute error after T iterations is exponentially small in T:

$$f(\boldsymbol{x}_T) - f(\boldsymbol{x}^*) \leq \frac{L}{2}\left(1 - \frac{\mu}{L}\right)^T ||\boldsymbol{x}_0 - \boldsymbol{x}^*||^2, \ \ T > 0 \tag{31}$$

Convergence Rate for Smooth and Strongly Convex Functions

To approximate within ϵ it is enough to iterate for T steps where

$$T > \frac{L}{\mu} log \left(\frac{R^2 L}{2\epsilon} \right) \tag{32}$$

Intuitively, if it is known that the steepness is between L and μ, the minima can be reached very fast!

How to Find Parameter of a Strongly Convex Function?

Intuition:

1. The smallest eigenvalue of the Hessian would be the strong convexity parameter
2. How to compute the smallest Eigenvalue efficiently[18]?
    1. Lanczos algorithm: *https://en.wikipedia.org/wiki/Lanczos_algorithm*

[18] memory is quadratic in parameters if the Hessian is computed explicitly

Optimizing Neural Networks

Question: What are the factors that affect the smoothness and strong convexity parameters?

1. Loss function
2. Network architecture via
   1. larger depth - larger L and μ
   2. larger width - larger L and μ
   3. activation functions - ReLU, Sigmoid etc. How do they affect?
   4. special structures such as skip connections make L lower and μ larger
   5. BatchNorm reduces L

Summary

What Next?