CSED490Y: Optimization for Machine Learning

Week 04-1: Gradient descent

Namhoon Lee

POSTECH

Spring 2022

Admin

Warnings based on the current progress:

1. Students with no contact
2. Single member group
3. Group without project topic
4. Group with project out of context

- example topics? → see "S03-1"
- TA email address?
  - PLMs
  - OptML
  - slides "S01-1"

Admin

Warnings based on the current progress:

1. Students with no contact
2. Single member group
3. Group without project topic
4. Group with project out of context

Topic W6 Monday

W8

28 March 11:59PM

Team up (Due-extended: 11:59PM on Monday 21 March): W5

- Form a group of up to 3 members.
- Email TA about your group members by the due date.
- You may use the discussion board on PLMS to find your teammates.

Admin

New requirements:

- Avoid late submission: if you miss the due date, you will receive a penalty of 10% of the total marks. $\boxed{-4}$ / 40
- Submit a (self-)plagiarism statement: "I certify that this project is entirely my own work, and I have not previously worked, am currently working or planning to work on any aspect of this course project ..." – TA will soon send out a form to sign.

↳ prevent re-using previ materials

↳ ensure a fair assessment.

Admin

New requirements:

- Avoid late submission: if you miss the due date, you will receive a penalty of 10% of the total marks.
- Submit a (self-)plagiarism statement: "I certify that this project is entirely my own work, and I have not previously worked, am currently working or planning to work on any aspect of this course project ..." – TA will soon send out a form to sign.

Important:

- Make sure to check important dates for project.
- This course does not require any assignments other than project.
- If there is anything you are not sure of about the project, come talk to me.

Convex function

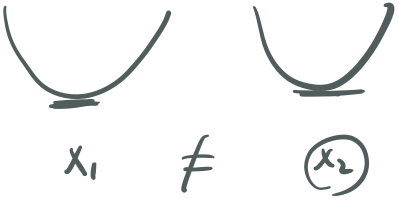
$f(y) \geq f(x) + \langle \nabla f(x), y-x \rangle$ differentiable fun.

opt. condition

More on convex functions..

- Notice from C^1 definition that $\boxed{\nabla f(x) = 0}$ implies $f(y) \geq f(x)$ for all y, so x is a global minimizer; this further explains why least squares can be solved by setting the derivative equal to zero.
- Strictly convex function have at most one global minimum; w and v can't both be global minima if $w \neq v$; it would imply convex combinations u of w and v would have $f(u)$ below the global minimum.

$\Rightarrow$ sol'n is unique.



Convex function

For strictly convex objective f there can be at most one global optimum.

Convex function

For strictly convex objective f there can be at most one global optimum.

Proof:

$x^* \neq x^\#$

1. Suppose x^* is a local minimum and also there exists another local minimum $x^\#$ ($\neq x^*$).

Convex function

For strictly convex objective f there can be at most one global optimum.

f convex $\rightarrow$ local $\rightarrow$ global min.

Proof:

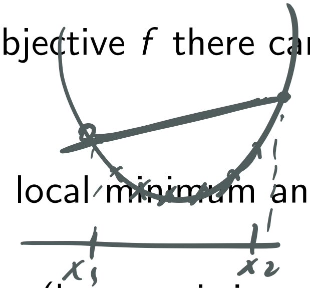
1. Suppose x^* is a local minimum and also there exists another local minimum $x^\#$ $(\neq x^*)$.
2. Since f is convex (because it is strictly convex), $\underline{f(x^*)}$ and $\underline{f(x^\#)}$ are both global minima, and $f(x^*) = f(x^\#)$.

Convex function

strictly convex

$$f(\underbrace{\theta x_1 + (1-\theta)x_2}_{y}) < \theta f(x_1) + (1-\theta) f(x_2)$$

For strictly convex objective f there can be at most one global optimum.



Proof:

1. Suppose x^* is a local minimum and also there exists another local minimum $x^\#$ ($\neq x^*$).
2. Since f is convex (because it is strictly convex), $f(x^*)$ and $f(x^\#)$ are both global minima, and $\boxed{f(x^*) = f(x^\#)}$.
3. The C^0 definition for $y = \theta x^* + (1-\theta)x^\#$, *i.e.*,

$$f(y) < \theta f(x^*) + (1-\theta)\boxed{f(x^\#)} = \theta f(x^*) + (1-\theta)\boxed{f(x^*)} = f(x^*)$$

contradicts that x^* is a global minimum.

$f(y) < f(x^*)$

Convex function

For strictly convex objective f there can be at most one global optimum.

Proof:

1. Suppose x^* is a local minimum and also there exists another local minimum $x^\#$ $(\neq x^*)$.
2. Since f is convex (because it is strictly convex), $f(x^*)$ and $f(x^\#)$ are both global minima, and $f(x^*) = f(x^\#)$.
3. The C^0 definition for $y = \theta x^* + (1-\theta)x^\#$, *i.e.*,

$$f(y) < \theta f(x^*) + (1-\theta)f(x^\#) = \theta f(x^*) + (1-\theta)f(x^*) = f(x^*)$$

   contradicts that x^* is a global minimum.
4. This means that for $x^\#$ to be a local minimum, it must be that $x^\# = x^*$.

Gradient descent

W2-2

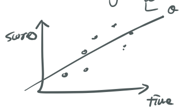
prob.

convex

Least squares : $\boxed{\min_{w \in \mathbb{R}^d} f(w) := \boxed{\frac{1}{2}} \|Xw - y\|^2}$

Scenario: predict scores for a ML course $\hat{y} = w^T x$ (θ)

$x \rightarrow \boxed{h} \rightarrow \hat{y}$



suppose. data

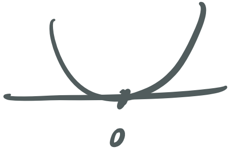
	time x_1	#course x_2	$\cdots$	x_d	y
1	3	5	$\cdots$		95
2					80
$\vdots$					90
n					$\vdots$

$\hat{y} = w_1 x_1 + w_2 x_2 + \cdots + w_d x_d$

$X = \begin{bmatrix} - x_1^T - \\ \vdots \\ - x_n^T - \end{bmatrix} \in \mathbb{R}^{n \times d}$ $\quad w \in \mathbb{R}^d$

$y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \in \mathbb{R}^n$

$f(x) = x^2$



$f'(x) = 2x = 0$

$x = 0$

$\nabla f(w) = X^T(Xw - y) = 0 \rightarrow w^* = (X^T X)^{-1} X y$

$\nabla^2 f(w) = X^T X \succeq 0$ p.s.d $\Rightarrow$ by C^2 def. $\Rightarrow$ f convex

$\Rightarrow$ w^* s.t. $\nabla f(w^*) = 0$ global min.

✱ Cross-validation

L2 regularized Least squares

$$\min_{w \in \mathbb{R}^d} \underbrace{\frac{1}{2}\|Xw - y\|^2}_{\text{error}} + \underbrace{\frac{\lambda}{2}\|w\|^2}_{\text{complexity of } w}$$

$$\begin{bmatrix} w_1 \\ \vdots \\ w_d \end{bmatrix} \qquad \|w\|^2$$

$$\boxed{\nabla f(w) = X^T(Xw - y) + \lambda w}$$

$$\nabla^2 f(w) = X^T X + \lambda I \succeq 0$$

unique solution,

$$v^T(X^T X + \lambda I)v = v^T X^T X v + \lambda v^T v$$

$$\forall v \in \mathbb{R}^d,\ v \neq 0 \qquad = \underbrace{\|Xv\|^2}_{\geq 0} + \underbrace{\lambda \|v\|^2}_{> 0} > 0$$

Gradient descent

linear system

$O(d^3)$

Cost of solving (regularized) least squares $\Rightarrow O(nd^2 + d^3)$

$\nabla f(w) = X^T(Xw - y) + \lambda w = 0$

$$\boxed{w^* = (X^TX + \lambda I)^{-1} X^T y}$$

if d is big

nd^2

pseudo inverse

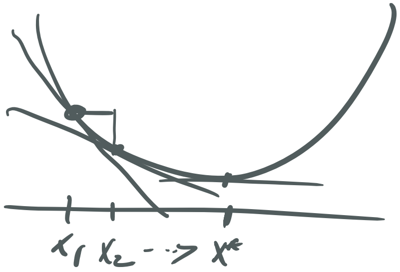
SO2-1

Gradient descent

$$\boxed{x_{t+1} = x_t - \eta \nabla f(x_t)} \qquad (GD)$$

Gradient descent algorithm

- for finding a minimum $f(x)$
- iteratively
- x_1 initial random guess $\to$ $f(x_1)$?
- update $x_1 \to x_2$ by taking a step into the neg. direction of the slope w/ stepsize η
- repeat until find soln.



Gradient descent

$f(\underset{=}{w}) = \|\underset{=}{X}w - \underset{=}{y}\|^2$

Lin. alg. { – Gram; – QR }

Numerical

$f(x) = \|\underset{=}{A}x - \underset{=}{b}\|^2$ (o)

Cost of solving (regularized) least squares with GD

Analytical (x)

$x_{t+1} = x_t - \eta \nabla f(x_t)$

$= x_t - \eta\left(\cancel{X^T(Xw - y) + \lambda w}\right)$

$\left(A^T(Ax_t - b) + \lambda x_t\right)$

$\boxed{O(nd^2 + \boxed{d^3})}$

cost (1 step update) : $O(nd)$

$\boxed{t < \max\{d, \frac{d^2}{n}\}}$

cost for t steps : $\boxed{O(ndt)}$ $\cancel{X^T(Xw - y) + \lambda w}$

L GD is fast enough

Q: How fast is GD?

Smoothness

Lipschitz continuous objective gradients

A differentiable function f is called L-smooth if there exists an $L > 0$ such that the following satisfies:

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\| \quad \forall\{x, y\} \ .$$

Smoothness

Lipschitz continuous objective gradients

A differentiable function f is called L-smooth if there exists an $L > 0$ such that the following satisfies:

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\| \quad \forall\{x, y\} \ .$$

- Gradient does not change arbitrarily quickly.
- Intuitively, without it the gradient would not be useful to decrease f.
- It is essential for convergence analyses of most gradient based methods.
- This is a fairly weak assumption and holds true for most ML models (including neural networks with smooth activations).

Smoothness

An important consequence of Lipschitz continuous objective gradient:

$$f(y) \leq f(x) + \nabla f(x)^\top (y - x) + \frac{L}{2}\|y - x\|^2 \ .$$

Proof (recall ftc):

Smoothness

An important consequence of Lipschitz continuous objective gradient:

$$f(y) \leq f(x) + \nabla f(x)^\top (y - x) + \frac{L}{2}\|y - x\|^2 \ .$$

Illustration:

Gradient descent progress bound

Under the quadratic upper bound, we are interested in how much progress gradient descent can make at each step.

Gradient descent progress bound

Under the quadratic upper bound, we are interested in how much progress gradient descent can make at each step.

Consider gradient descent with $\eta = 1/L$.

$$x_{t+1} = x_t - \frac{1}{L}\nabla f(x_t) \ .$$

Gradient descent progress bound

Under the quadratic upper bound, we are interested in how much progress gradient descent can make at each step.

Consider gradient descent with $\eta = 1/L$.

$$x_{t+1} = x_t - \frac{1}{L}\nabla f(x_t) \ .$$

Plugging this into the bound gives

Convergence of gradient descent

From the C^1 definition of convex function, we can get

$$f(x_t) \leq f(x^*) + \nabla f(x_t)^\top (x_t - x^*)$$

Convergence of gradient descent

From the C^1 definition of convex function, we can get

$$f(x_t) \leq f(x^*) + \nabla f(x_t)^\top (x_t - x^*)$$

Plugging this into the progress bound we derived previously gives

Convergence of gradient descent

Thank you

Any questions?

Credits

A lot of material in this course is borrowed or derived from the following:

- Numerical Optimization, Jorge Nocedal and Stephen J. Wright.
- Convex Optimization, Stephen Boyd and Lieven Vandenberghe.
- Convex Optimization, Ryan Tibshirani.
- Optimization for Machine Learning, Martin Jaggi and Nicolas Flammarion.
- Optimization Algorithms, Constantine Caramanis.
- Advanced Machine Learning, Mark Schmidt.