MATH173A Optimization for Data Science & Machine Learning

Rabbittac
https://www.rabbittac.xyz/post/math173a-optimization-notes

December 24, 2020

Contents

MATH173A Optimization **Fall 2020**

Lecture 2: Convex Function & Convex Set

Lecturer: Rayan Saab *Scribes: Rabbittac*

Goal: To solve the form of problem

$$\min_{x\in\Omega} f(x) \tag{2.1}$$

where Ω is a **constraint** and $f(x)$ is **objective function**.

When Ω is the whole space, or $\Omega = \mathbb{R}^n$, we have no constraint and we call this an unconstrained optimization problem.

Examples:
- $\min_{x_1,x_2} 2x_1^2 + 3x_2^2 - 4x_1x_2 + 7$
- $\min_{x\in\mathbb{R}^n} \sum_{i=1}^m |a_i^T x - b_i|^2$

When Ω is a strict subset of $\mathbb{R}^n$, we say the optimization problem is constrained.

Examples:
- $\min_{2\le x_1\le 5, 3\le x_2\le 7} 2x_1^2 + 3x_2^2 - 4x_1x_2 + 7$
- $\min_{x\in B_2^n} \sum_{i=1}^m |a_i^T x - b_i|^2$ where $B_2^n = \{x \in \mathbb{R}^n : \sum_{i=1}^n x_i^2 \le 1\}$

We say that x^* is a solution of 2.1 if:
- $x^* \in \Omega$ (x^* is feasible)
- $\forall x \in \Omega, f(x^*) \le f(x)$

We say that x^* is a **local solution** of 2.1 if:
- $x^* \in \Omega$
- There is a neighborhood N around x^* such that $\forall x \in N \wedge \Omega : f(x^*) \le f(x)$

Remark: Strict local minimum — replace $\le$ by $<$

Question: How to check if a certain point $\hat{x}$ is optimal or even locally optimal?

Definition 2.1 (Convex Function) *We say $f : \Omega \to \mathbb{R}$ is **convex** if $\forall x, y \in \Omega$, and $\forall \alpha \in [0, 1]$ we have*

$$f(\alpha x + (1-\alpha)y) \le \alpha f(x) + (1-\alpha) f(y)$$

We say f is strictly convex if "$\le$" is replaced by "$<$".

e.g. $f(x) = x^2$ is convex

Proof:

$$\begin{aligned} f(\alpha x + (1-\alpha)y) &= (\alpha x + (1-\alpha)y)^2 \\ &= \alpha^2 x^2 + (1-\alpha)^2 y^2 + 2\alpha(1-\alpha)xy \\ \alpha f(x) + (1-\alpha) f(y) &= \alpha x^2 + (1-\alpha) y^2 \end{aligned}$$

Then $(\alpha f(x) + (1-\alpha)f(y)) - f(\alpha x + (1-\alpha)y)$

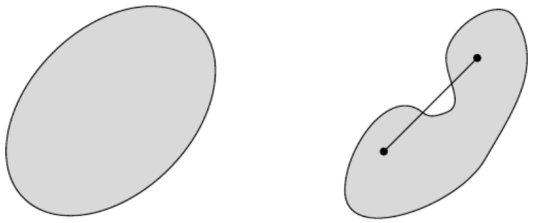
$$= \alpha(1-\alpha)x^2 + [(1-\alpha) - (1-\alpha)^2]y^2 - 2\alpha(1-\alpha)xy$$
$$= \alpha(1-\alpha)(x-y)^2 \geq 0$$

■

e.g. $f(x) = x^3$ is not convex

Definition 2.2 (Convex Set) *A set C is convex if $\forall x_1, x_2 \in C$ and $\forall \alpha \in [0,1]$, we have*

$$\alpha x_1 + (1-\alpha)x_2 \in C$$



(a) Convex set (b) Non-convex set

Fun Facts:
- If C_1 and $C2$ are convex sets, then $C_1 \cap C_2$ is convex
- The intersection of any collection of convex sets is convex
- $C1 \cup C2$ is not necessarily convex

Theorem 2.3 *Consider the optimization problem*

$$\min_{x \in \Omega} f(x)$$

where $f : \Omega \to \mathbb{R}$ is a convex function and Ω is a convex set. Then if x^* is a local minimum, it is also a global minimum.

MATH173A Optimization **Fall 2020**

Lecture 3: Properties of Convex Functions

Lecturer: Rayan Saab *Scribes: Rabbittac*

Proof to last lecture's theorem

Proof: Towards a contradiction, suppose that x^* is a local minimum that is not a global minimum. Then $\exists w \in \Omega : f(w) < f(x^*)$

Since f is convex, $\forall \alpha \in [0,1]$

$$\begin{aligned} f(\alpha w + (1-\alpha)x^*) &\leq \alpha f(w) + (1-\alpha) f(x^*) \\ &< \alpha f(x^*) + (1-\alpha) f(x^*) \\ &= f(x^*) \end{aligned}$$

In short: $f(\alpha w + (1-\alpha)x^*) < f(x^*)$. This means every point z on the line segment connecting w to x^* has $f(z) < f(x^*)$. So x^* cannot be a local minimum. This contradicts our original assumption that x^* is not a globally minimum.

So we conclude that x^* is a global minimum. ■

But we still need an algorithm to find minimum. Before that, some concepts are to be reviewed.

Definition 3.1 (Gradient) *For $f : \mathbb{R}^n \to \mathbb{R}$, the **gradient** is defined* [1]

$$\begin{aligned} \nabla f(x) &= \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n} \right) \\ &= \left(\frac{\partial f}{\partial x_i} \right)_{i=1}^{n} \end{aligned}$$

Hessian is a generalization of 2nd derivative.

Definition 3.2 (Hessian) *The **Hessian** is an $n \times n$ matrix with*

$$\nabla^2 f(x) = \left(\frac{\partial^2 f}{\partial x_i \partial x_j} \right)_{i,j=1,\ldots,n}$$

e.g. For $f : \mathbb{R}^2 \to \mathbb{R}$ given by $f(x_1, x_2) = 2x_1^2 + x_2^4 - 2x_2^2 + 1$.

$\nabla f(x_1, x_2) = (4x_1, 4x_2^3 - 4x^2)$

$\nabla^2 f(x_1, x_2) = \begin{pmatrix} 4 & 0 \\ 0 & 12x_2^2 - 4 \end{pmatrix}$

Definition 3.3 (Directional Derivative) *The **directional derivative** is rate of change in the direction of the unit vector $\vec{v}$*

[1] Also written as $(\nabla) f(x)$ or $\nabla f|_x$

Recall that a partial derivative of $f : \mathbb{R}^n \to \mathbb{R}$ is given by $\frac{\partial f}{\partial x_1} = \lim_{h\to 0} \frac{f(x_1+h, x_2, x_3) - f(x_1, x_2, x_3)}{h} = \lim_{h\to 0} \frac{f(\vec{x} + h\vec{i}) - f(\vec{x})}{h}$

Theorem 3.4 *The directional derivative of a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ in the direction of the unit vector v is $\langle \nabla f, v = \sum_{i=1}^n \frac{\partial f}{\partial x_i} \cdot v_i \rangle$*

e.g. $f(x_1, x_2, x_3) = x_1^2 e^{-x_2 x_3}$. Find the rate of change of f in the direction of $\vec{v} = (\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})$ at $(1, 0, 0)$.

$\nabla f = (2x_1 e^{-x_2 x_3}, -x_1^2 x_3 e^{-x_2 x_3}, -x_1^2 x_2 e^{-x_2 x_3}) \implies \langle \nabla f, v \rangle|_{1,0,0} = \langle (2, 0, 0), (\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}) \rangle = \frac{2}{\sqrt{3}}$

Remark: Make sure to check that $\|v\| = 1$. Otherwise you require to replace v by $\frac{v}{\|v\|}$.

Lemma 3.5 *The gradient is in the direction of greatest increase of the function.*

Proof: Suppose $\|\vec{n}\|$ is such that $\|n\| = 1$ then the rate of change of f in the direction of n is $\langle \nabla f, n \rangle = \|\nabla f\| \|n\| \cos(\theta)$ where θ is angle between ∇f and n.

So if $\nabla f \neq 0$, $\langle \nabla f, n \rangle$ is greatest when $\theta = 0$, i.e. when $\vec{n}$ and ∇f are in the same direction ■

e.g. $f(x_1, x_2) = x_1^2 + x_2^3$. The direction of greatest increase of $f(x_1, x_2)$ at $p = (2, 3)$ is $\nabla f|_p = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2} \right)\Big|_p = (4, 27)$

For function of single variable

$$f(y) = f(x) + f'(\xi)(y - x)$$

where ξ is somewhere between x and y, i.e. $f'(\xi) = \frac{f(y)-f(x)}{y-x}$

Theorem 3.6 (Taylor's Theorem) *Generally, for function of single variable*

$$f(y) = f(x) + f'(\xi)(y - x) + \frac{f''(x)(y-x)^2}{2!} + \cdots + f^{(n)}(\xi)\frac{(y-x)^n}{n!}$$

If $f : \mathbb{R}^n \to \mathbb{R}$ is twice continually differentiable

$$f(y) = f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(\xi)(y - x)^2$$

[2] Also know as Taylor series approximation of f around point x

MATH173A Optimization **Fall 2020**

Lecture 4: Equivalent Conditions for Convexity

Lecturer: Rayan Saab *Scribes: Rabbittac*

Checking convexity by definitions is tedious, so we want to explore other methods to check.

Theorem 4.1 *If $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable then f is convex if and only if $\forall x, y \in \mathbb{R}^n$, we have*

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

Note: The right hand side is part of Taylor expansion of f.

Definition 4.2 (Positive Semi-definite) *$A \in \mathbb{R}^{n \times n}$ is **positive semi-definite** (PSD) if and only if $\forall x \in \mathbb{R}^n : x^T A x \geq 0$. And eigenvalues of A are greater than 0.*

Theorem 4.3 *If f is twice continuously differentiable then f is convex and only if $\forall x$:*

$$\nabla^2 f(x) \text{ is positive and semi-definite}$$

e.g. $A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$ is positive semi-definite.

Checking by definition: consider $x \in \mathbb{R}^3, x = (x_1, x_2, x_3)$, then $x^T A x = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} =$

$2x_1^2 + 2x^2 - 2x_1 x_2 - 2x_2 x_3 + x^3 = x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 \geq 0$.

e.g. Check $f : \mathbb{R}^2 \to \mathbb{R}^3$ given by $f(a, b, c) = a^4 + b^2 + c^2$ is convex.

Note f is twice continuously differentiable. Then $\nabla f(a, b, c) = (4a^3, 2b, 2c), \nabla^2 f(a, b, c) = \begin{pmatrix} 12a^2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}$.

The Hessian is PSD. Then f is convex.

Definition 4.4 (Monotone) *A mapping $g : \mathbb{R}^n \to \mathbb{R}^n$ is called **monotone** if $\forall x, y \in domain(g)$*

$$\langle g(x) - g(y), x - y \rangle \geq 0$$

e.g. $g : \mathbb{R}^n \to \mathbb{R}^n, g(x) = 2x$, then $\forall x, y \in \mathbb{R}^n$, $\langle g(x) - g(y), x - y \rangle = \langle 2x - 2y, x - y \rangle = 2\|x - y\|^2 \geq 0$. This shows g is monotone.

Theorem 4.5 *A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only if ∇f is monotone.*

Proof: (I.) Convex ⇒ gradient monotone: by previous theorem(4.1),

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

Adding both gives:

$$\begin{aligned} f(y) + f(x) &\geq f(x) + f(y) + \nabla f(x)^T (y - x) + \nabla f(y)^T (x - y) \\ &\Rightarrow \nabla f(x)^T (y - x) + \nabla f(y)^T (x - y) \geq 0 \\ &\Rightarrow (\nabla f(y) - \nabla(x))^T (y - x) \geq 0 \end{aligned}$$

So ∇f is monotone.

(II.) gradient monotone ⇒ convex:

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

To do this we will parameterize the line segment between x and y, and define $g : \mathbb{R} \to \mathbb{R}$ that

$$\begin{aligned} g(t) &= f(x + t(y - z)) \\ \Rightarrow g'(t) &= \nabla f(x + t(y - z))^T (y - x) \end{aligned}$$

Now use the fact that ∇f is monotone to write

$$\langle \nabla f(x + t(y - x)) - \nabla f(x), t(y - x) \rangle \geq 0$$

Then $\forall t \in [0, 1] : g'(t) \geq g'(0)$

$$\begin{aligned} \int_0^1 g'(0) dt &\geq \int_0^1 g'(0) dt \\ &\Rightarrow g(1) \geq g(0) + g'(0) \\ &\Rightarrow f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle \end{aligned}$$

■

MATH173A Optimization **Fall 2020**

Lecture 5: Gradient Descent

Lecturer: Rayan Saab *Scribes: Rabbittac*

Definition 5.1 (Descent Direction) *$\vec{v}$ is a descent direction for $f : \mathbb{R}^n \to \mathbb{R}$ at x if $\langle \vec{v}, \nabla f \mid_x \rangle < 0$.*

Note: Why this definition makes sense? Taylor's Theorem tells that $f(\vec{x} + \mu\vec{v}) = f(\vec{x}) + \mu\vec{v}^T \nabla f(\vec{\xi})$ so $\vec{\xi} = \vec{x} + a\vec{v}$ where $a \in [0, \mu]$. But if ∇f is continuous then $\exists \mu$ that is small enough such that $\forall a \in [0, \mu]$: $v^T \nabla f(\vec{x} + a\vec{v}) < 0$. This in turn implies that $f(x + \mu v) < f(x)$. So moving in the direction of v by a small amount reduces the value of f.

e.g. $f(\vec{x}) = x_1^2 + 2x_2^2$. $\nabla f(x) = (2x_1, 4x_2)$. At $\vec{x} = (1, 1), \nabla f|_{(1,1)} = (2, 4)$. Let $\vec{v} = (0, -1)$, then $\langle \vec{v}, \nabla f|_{(1,1)} \rangle = -4 < 0$. So $\vec{v}$ is a descent direction for f at $(1, 1)$.

Let's check $f(x + \mu v)$ compared to $f(x)$. $f(\vec{x} + \mu\vec{v}) = x_1^2 + 2(x_2 - \mu)^2 = (x_1 + \mu v_1, x_2 + \mu v_2)$. So $f(\vec{x} + \mu\vec{v})$ at $(1, 1)$ is $f(\vec{x} + \mu\vec{v}) = 1 + 2(1 - \mu)^2 = 3 + 2\mu^2 - 2\mu$ while $f(\vec{x}) = 3$. So $f(\vec{x} + \mu\vec{v}) < f(\vec{x})$ whenever $0 < \mu < 1$.

Note: If we choose $\vec{v} = -\nabla f(x)$, we always get a descent direction provided $\nabla f(x) \neq 0$.

Gradient descent is a popular and important algorithm and widely for its simplicity and computationally cheap.

Goal: To find a (local) minimizer for function $f : \mathbb{R}^n \to \mathbb{R}$. i.e. find x^* s.t. $\forall x \in N, f(x^*) < f(x)$ where N is a neighborhood around.

Idea: Suppose you make a guess x and now you want to improve it. You can pick a descent direction, e.g. $\vec{v} = -\nabla f(x)$ and move by a small amount in that direction: $f(\vec{x} + \mu\vec{v}) < f(\vec{x})$.

Algorithm 5.1 Gradient Descent

1: choose $x^{(0)} \in \mathbb{R}^n$
2: **for** $t = 1, 2, \ldots$ until a stopping criterion **do**
3: Set $x^{(t)} = x^{(t-1)} - \mu \nabla f(x^{(t-1)})$
4: **end for**

e.g. $f : \mathbb{R}^2 \to \mathbb{R}$ and $f(x) = x_1^2 + 2x_2^2$. Suppose we start at $x^{(0)} = (2, 3)$ and suppose we choose $\mu = 0.1$. Then GD gives $x^{(1)} = x^{(0)} - \mu \nabla f(x^{(0)}) = (2, 3) - 0.1 \times (4, 12) = (1.6, 1.8)$, $x^{(2)} = (1.6, 1.8) - 0.1 \times (3.2, 7.2) = (1.28, 1.02)$, $\ldots$, $x^{10} \approx (0.2147, 0.0181)$ ...

Issues when applying GD:
- How do we choose μ?
- How do we know when to stop?

Theorem 5.2 ***(Necessary Condition for Optimality)***

1. *If f is continuously differentiable and x^* is a local minimum then*

$$\nabla f(x^*) = 0$$

2. If $\nabla^2 f$ is continuous and x^* is a local minimum

$$\nabla^2 f(x^*) \succeq {}^{3} 0$$

This theorem means that if f is continuously differentiable and has a continuous Hessian, then we must have

$$\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succeq 0$$

Proof seen next lecture.

[3] *Notation:* $A \succeq 0$ if and only if A is PSD.

MATH173A Optimization **Fall 2020**

Lecture 6: Condition for Optimality

Lecturer: Rayan Saab *Scribes: Rabbittac*

5.2 means that if f is continuously differentiable and has continuous Hessian then we must have

$$\nabla f(x^*) = 0$$

$$\nabla^2 f(x^*) \succeq 0$$

Proof:

(1) Suppose by way of contradiction that $\nabla f(x^*) \neq 0$, then $\vec{v} = -\nabla f(x^*)$ is a descent direction. So x^* is not a local minimum and we have a contradiction.

(2) Since x^* is a local minimum then $f(x^* + t\vec{v}) \geq f(x^*)$ $\forall \vec{v}$ and t that are small enough. Now apply Taylor's theorem

$$f(x^* + tv) = f(x^*) + tv^T \nabla f(x^*) + \frac{1}{2} t^2 v^T \nabla^2 f(x^* + \tilde{t}v) v$$

for $\tilde{t} \in (0, t)$. Then

$$f\left(x^* + tv\right) - f\left(x^x\right) = \frac{1}{2} t^2 v^T \nabla^2 f(x^* + \tilde{t}v) v \geq 0$$

Taking limits as $t \to 0$ we also have $\tilde{t} \to 0$. This gives $v^T \nabla^2 f(x^*) v \geq 0$, which is equivalent to saying $\nabla^2 f(x^*) \succeq 0$. ■

Theorem 6.1 (Sufficient Condition for Optimality) *If f is twice continuously differentiable and x^* satisfies*

$$\nabla f(x^*) = 0 \quad \textit{and} \quad \nabla^2 f(x^*) \succ 0$$

then x^* is a local minimum.

Proof: $\forall \vec{h}$ with $\|h\|$ small

$$f(x^* + h) = f(x^*) + h^T \nabla f(x^*) + \frac{1}{2} h^T \nabla^2 f(x^* + \alpha h) h > 0$$

by positive definiteness of $\nabla^2 f(x^*)$ and continuity of $\nabla^2 f$. Then $f(x^* + h) \geq f(x^*)$. Then x^* is a local minimum. ■

e.g. Let $f(x) = x_1^3 + 2x_2^2$. Then $\nabla f(x) = (3x_1^2, 4x_2)$. Setting $\nabla f(x^*) = 0$ gives $x^* = (0, 0)$ and $\nabla^2 f(x) = \begin{pmatrix} 6x_1 & 0 \\ 0 & 4 \end{pmatrix}$. $\nabla^2 f(x^*) = \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix} \succeq 0$. But $f(x^*) = f(0, 0) = 0$ and $\forall \epsilon > 0 : f(-\epsilon, 0) = -\epsilon^3 < 0$. So x^* is not a local minimum.

e.g. $f(x) = \frac{1}{2} x_1^2 + x_1 x_2 + 2x_2^2 - 4x_1 - 4x_2 - x_2^3$. Then $\nabla f(x) = \begin{pmatrix} x_1 + x_2 - 4 \\ x_1 + 4x_2 - 4 - 3x_2^2 \end{pmatrix}$ and $\nabla^2 f(x) = \begin{pmatrix} 1 & 1 \\ 1 & 4 - 6x_2 \end{pmatrix}$. Setting $\nabla f(x) = 0$ gives $x^* = (4, 0)$ and $x^{**} = (3, 1)$. To check, $\nabla^2 f(x^{**}) = \begin{pmatrix} 1 & 1 \\ 1 & -2 \end{pmatrix}$ has eigenvalues $\lambda_1 = \frac{1}{2}(-1 - \sqrt{13}) < 0$. So $\nabla^2 f(x^{**})$ is not positive semidefinite. So x^{**} is not a local minimum.

Check x^* we find $\nabla^2 f(x^*) = \begin{pmatrix} 1 & 1 \\ 1 & 4 \end{pmatrix}$ which is positive definite. So x^* satisfies the conditions of the sufficient condition theorem. Then x^* is a local minimum.

Now we assume f is convex, then

Theorem 6.2 *Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex and continuously differentiable, then x^* is a global minimum if and only if $\nabla f(x^*) = 0$.*

Proof: (1) Suppose x^* is a global minimum, then there is no descent direction from x^*. Then $\nabla f(x^*) = 0$.

(2) Suppose $\nabla f(x^*) = 0$, then convexity gives $\forall x : f(x) \geq f(x^*) + \nabla f(x^*)^T (x - x^*)$. Then $f(x) \geq f(x^*)$. Then x^* is a global minimum. ■

MATH173A Optimization **Fall 2020**

Lecture 7: L-Lipschitz

Lecturer: Rayan Saab *Scribes: Rabbittac*

Definition 7.1 (L-Lipschitz) *A function $f : \Omega \to \mathbb{R}$ is **L-Lipschitz** if $\forall x, y \in \Omega$*

$$|f(x) - f(y)| \leq L\|x - y\|$$

How much f changes between x and y is determined by how far x and y are distant from each other.

e.g. $f : \mathbb{R} \to \mathbb{R}$, $f(x) = |x|$ is Lipschitz because $|f(x) - f(y)| = ||x| - |y|| \leq |x - y|$. The Lipschitz constant is 1.

Lemma 7.2 *If f is L-Lipschitz, convex, and differentiable, then*

$$\|\nabla f(x)\| \leq L$$

$\forall x \in \Omega$.

Proof: For any $x, y \in \Omega$

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

Pick $x = y + \nabla f(y)$. So $L\|\nabla f(y)\| \geq \|\nabla f(y)\|^2$, thus $\|\nabla f(y)\| \leq L$. ■

Theorem 7.3 *$\forall x \in \Omega : \|\nabla f(x)\| \leq L$, then*

$$|f(x) - f(y)| \leq L\|x - y\|$$

Proof: By Taylor's theorem, for some $t \in (0, 1)$

$$f(x) - f(y) = \nabla f(tx + (1 - t)y)^T (x - y)$$

So $|f(x) - f(y)| = |\nabla f(tx + (1 - t)y)^T (x - y)| \leq \|f(tx + (1 - t)y)\|\|(x - y)\| \leq L\|x - y\|$. ■

The next theorem presents one result on the choice of μ and the convergence of GD.

Theorem 7.4 *Let f be convex, differentiable, L-Lipschitz and let*

$$\|x^{(0)} - x^*\| \leq R$$

$$\|\nabla f(x)\| \leq L$$

Choose $\mu = \frac{R}{L\sqrt{t}}$. Then

$$f(\frac{1}{t}\sum_{s=0}^{t} x^{(s)}) - f(x^*) \leq \frac{RL}{\sqrt{t}}$$

e.g. Consider $f(x_1, x_2) = \sin(x_1) + x_2$ and notice that $\nabla f(x_1, x_2) = \begin{pmatrix} \cos x_1 \\ 1 \end{pmatrix}$. Then $\|\nabla f\| \leq \sqrt{\cos^2(x_1 + 1)} \leq \sqrt{2}$. f is Lipschitz with $L = \sqrt{2}$. So if we run GD for t-steps with $\mu^{(t)} = \frac{R}{\sqrt{2}\sqrt{t}}$ and we will get within $\frac{R\sqrt{2}}{\sqrt{t}}$ of a local minimum.

Proof: First, note that $f(x^*) \geq f(x^{(s)}) + \nabla f(x^{(s)})^T (x^* - x^{(s)})$. Then

$$f(x^*) - f(x^{(s)}) \geq \nabla f(x^{(s)})^T (x^* - x^{(s)})$$

Next, note that $x^{(s+1)} = x^{(s)} - \mu \nabla f(x^{(s)})$ so

$$\nabla f(x^{(s)}) = \frac{x^{(s)} - x^{(s+1)}}{\mu}$$

$f(x^{(s)}) - f(x^*) \leq \frac{1}{\mu} \langle x^{(s)} - x^{(s+1)}, x^{(s)} - x^* \rangle$. Applying

$$a^T b = \frac{\|a\|^2 + \|b\|^2 - \|a - b\|^2}{2}$$

then

$$f(x^{(s)}) - f(x^*) \leq \frac{1}{2\mu} (\|x^{(s)} - x^*\|^2 + \|x^{(s)} - x^{(s+1)}\|^2 - \|x^{(s+1)} - x^*\|^2)$$

But $x^{(s)} - x^{(s+1)} = \mu \nabla f(x^{(s)})$. So

$$f(x^{(s)}) - f(x^*) \leq \frac{1}{2\mu} (\|x^{(s)} - x^*\|^2 - \|x^{(s+1)} - x^*\|^2) + \frac{\mu}{2} \|\nabla f(x^{(s)})\|^2$$

Summing from $s = 0$ to $t - 1$, we obtain

$$\begin{aligned} \sum_{s=0}^{t-1} \Big(f(x^{(s)}) - f(x^*) \Big) &\leq \frac{1}{2\mu} (\|x^{(s)} - x^*\|^2 - \|x^{(s+1)} - x^*\|^2) + \frac{\mu}{2} \sum_{s=0}^{t-1} \|\nabla f(x^{(0)})\|^2 \\ &\leq \frac{1}{2\mu} (R^2 + \frac{\mu}{2} L^2 t) \end{aligned}$$

Dividing by t and by convexity of f,

$$\frac{1}{t} \sum_{s=0}^{t-1} f(x^{(s)}) - f(x^*) \leq \frac{1}{2\mu t} R^2 + \frac{\mu}{2} L^2 \leq \frac{RL}{\sqrt{t}}$$

■

MATH173A Optimization **Fall 2020**

Lecture 8: Interpretation of GD

Lecturer: Rayan Saab *Scribes: Rabbittac*

Note: In practice, we may not know R exactly, but we may have some good guess of it. So we might just pick $\mu = \frac{C}{L\sqrt{t}}$ for some choice of C. This gives

$$f\left(\frac{1}{t}\sum_{s=0}^{t-1} x^{(s)}\right) - f\left(x^*\right) \leq \frac{1}{2\mu t}R^2 + \frac{\mu}{2}L^2 \leq \frac{1}{\sqrt{t}}$$

NOT DOING YET!

Definition 8.1 (Lp Norm) *Let $x \in \mathbb{R}^N$ then we define*

$$\|x\|_1 := \sum_{i=1}^{N} |x_i|$$

$$\|x\|_2 := \sqrt{\sum_{i=1}^{N} |x_i|^2}$$

$$\|x\|_p := (\sum_{i=1}^{N} |x_i|^p)^{\frac{1}{p}}$$

$$\|x\|_\infty := \max |x_i|$$

MATH173A Optimization **Fall 2020**

Lecture 9: Gradient Descent under Constraints

Lecturer: Rayan Saab *Scribes: Rabbittac*

We have solved $\min_{x\in\mathbb{R}^n} f(x)$ by running iteration $x^{(t+1)} = x^{(t)} - \mu\nabla f(x^{(t)})$. Now we focus on problem

$$\min f(x) \quad \text{s.t.} \quad x \in \Omega \subset \mathbb{R}^n$$

The challenge is that even $x^{(0)} \in \Omega$ or other $x^{(t)} \in \Omega$, there is no guarantee that $x^{(t+1)}$ given by GD is also in Ω.

Definition 9.1 (Projection) *The **projection** of a point x onto a set Ω is defined as the closest point in Ω to x. i.e.*

$$\prod_{\Omega} x = \arg\min_{y\in\Omega} \|x - y\|$$

e.g. If B_2^n[4] $= \{x : \|x\| \le 1\}$, then $\prod_\Omega(x) = \begin{cases} x & \|x\| \le 1 \\ \frac{x}{\|x\|} & \|x\| > 1 \end{cases}$. When the point is in the circle, the projection is itself; when the point is out of the circle, the projection is intersection with the circle.

e.g. If $\Omega = \{x : x_1 \ge 0\}$, then $\prod_\Omega(x) = \begin{cases} x & x \ge 0 \\ (0, x_2, x_3, \ldots, x_n) & x < 0 \end{cases}$

Theorem 9.2 (Projected SD) *In order to solve,*

$$\min f(x) \quad \textit{s.t.} \quad x \in \Omega$$

where $\Omega \subset \mathbb{R}^n$ is convex, we run the iterations

$$x^{(t+1)} = \prod_{\Omega} \left(x^{(t)} - \mu^{(t)} \nabla f(x^{(t)}) \right)$$

In explanation, it combines GD step $y^{(t+1)} = x^{(t)} - \mu^{(t)}\nabla f(x^{(t)})$ and constrained optimization problem $x^{(t+1)} = \arg\min_{x\in\Omega} \|y^{(t+1)} - x\|$

Lemma 9.3 (Properties of Projection) *If $\Omega \subset \mathbb{R}^n$ is convex and closed (not empty) and if $x \in \Omega, y \in \mathbb{R}^n$, then*

1. $\prod_\Omega(y)$
2. $\prod_\Omega(\prod_\Omega(y)) = \prod_\Omega(y)$
3. $\langle \prod_\Omega(y) - x, \prod_\Omega(y) - y \rangle \le 0$
4. $\| \prod_\Omega(y) - x\|^2 + \|y - \prod_\Omega(y)\|^2 \le \|y - x\|^2$
5. $\| \prod_\Omega(y) - x\| \le \|y - x\|$

[4] B_i^n is L_i-ball in n-dim metric space. In this case, B_2 is a circle.

MATH173A Optimization **Fall 2020**

Lecture 10: L-smooth

Lecturer: Rayan Saab *Scribes: Rabbittac*

Recall previous convergence result of GD: in order to get an error $e \leq \epsilon$, we need $T \geq \frac{1}{\epsilon^2}$ iterations. We want better solution given "nicer" f.

Definition 10.1 (L-smooth) *A function $f : \mathbb{R}^n \to \mathbb{R}$ is **L-smooth** if its gradient is L-Lipschitz. i.e.*

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$$

Theorem 10.2 *If $f : \mathbb{R}^n \to \mathbb{R}$ is L-smooth, convex, and $0 \leq \mu \leq \frac{1}{L}$, then the GD iterations*

$$x^{(t+1)} = x^{(t)} - \mu \nabla f(x^{(t)})$$

satisfying [5]

$$f(x^{(t)}) - f(x^*) \leq \frac{1}{2t\mu}\|x^{(0)} - x^*\|$$

Note: Here f has iterate itself instead of the average; the error decays linearly with number of steps.

The projected GD can be modified in the same way for L-smooth functions to obtain the same convergence rate when solving $\min_{x\in\Omega} f(x)$ when $\Omega \subset \mathbb{R}^n$ is convex.

Issue with picking μ in practice: the convergence theorem requires known L in order to set μ. But we do not know L in many cases. To interpret, we want pick $\mu^{(t)}$ to minimize

$$f(x^{(t+1)}) = f(x^{(t)}) - \mu^{(t)} \nabla f(x^{(t)})$$

But solving exact $\mu^{(t)}$ is often hard, so we settle for an approximation: at every solution t, use "Back-tracing Line Search". Note that for any descent direction $\vec{p}$

$$f(x) - \nabla f(x)^T p \leq f(x - p) \leq f(x) - \gamma \nabla f(x)^T p$$

for some small $\gamma > 0$. Pick $p = \mu \nabla f(x)$ and plug in above equation

$$f(x - \mu \nabla f(x)) \leq f(x) - \gamma\mu\|\nabla f(x)\|^2$$

So we expect that for μ that is small enough, above should hold. Then we — fix γ, start with some μ, decrease μ iteratively until [6]

$$f(x^{(t+1)}) = f(x^{(t)} - \mu^{(t)} \nabla f(x^{(t)})) \leq f(x^{(t)}) - \mu^{(t)}\gamma\|\nabla f(x^{(t)})\|^2$$

A summary of this algorithm is 11.1.

[5] We do not prove this result. But the proof uses the fact for L-smooth functions: $f(y) \leq f(x) + \nabla f(x)^T(y - x) + \frac{L}{2}\|y - x\|_2^2$.

[6] The following inequality is called **Armijo condition**

MATH173A Optimization **Fall 2020**

Lecture 11: Back-tracing Line Search

Lecturer: Rayan Saab *Scribes: Rabbittac*

Algorithm 11.1 Finding μ by Back-tracing Line Search

Pick $\beta < 1, \gamma < 1$
for each GD step t **do**
  Set $v = -\nabla f(x^{(t)})$
  Set $\mu^{(t)} = 1$
  while not $f(x^{(t)} - \mu^{(t)}\nabla f(x^{(t)})) \leq f(x^{(t)}) - \mu^{(t)}\gamma\|\nabla f(x^{(t)})\|^2$ **do**
    Set $\mu^{(t)} = \beta\mu^{(t)}$
  end while
end for

e.g. Consider $f(x_1, x_2) = (x_1 - 1)^2 + (x_1 + x_2 - 1)^2$.

$\nabla f(x_1, x_2) = \big(4(x_1 - 1)^3 + 2(x_1 + x_2 - 1), 2(x_1 + x_2 - 1)\big)$. Set $x^{(0)} = (0, 0)$, then $x^{(1)} = x^{(0)} - \mu\nabla f(x^{(0)}) = (0, 0) - \mu(-6, -2)$. We want to pick μ to minimize $f((0, 0) - \mu(-6, -2)) = f(6\mu, 2\mu) = (6\mu - 1)^4 + (8\mu - 1)^2$. Use back-tracing line search, start with $\mu = 1$: $f(x^{(1)}) = 674 > f(x^{(0)}) - \mu\gamma\|\nabla f(x^{(0)})\|^2 = -18$. So we try $\mu = 1 \times 0.8 = 0.8$, then $f(x^{(1)}) = 277.67 > f(x^{(0)}) - 0.8 \times 0.5\|\nabla f(x^{(0)})\|^2 = -14$. Then we continue trials until $\mu = 0.8^{11}$: $f(x^{(1)}) = 0.153 \leq f(x_{(0)}) - 0.5 \times 0.8^1 1\|\nabla f(x^{(0)})\|^2$. So we choose this μ and set $x^{(1)} = x^{(0)} - \mu\nabla f(x^{(0)})$. Repeat this process for $t = 2, 3 \ldots$

Theorem 11.1 *For an L-smooth convex function with $\mu^{(t)}$ set by back-tracing line search, GD gives*

$$f(x^{(t)}) - f(x^*) \leq \frac{1}{2t \min_{s=1,\ldots,t} \mu^{(s)}}$$

and

$$\min_{s=1,\ldots,t} \mu^{(s)} \geq \min(1, \frac{\beta}{L})$$

i.e. this guarantees $f(x^{(t)}) - f(x^*) \leq \frac{L}{2t\beta}$

MATH173A Optimization **Fall 2020**

Lecture 12: Newton's Method

Lecturer: Rayan Saab *Scribes: Rabbittac*

GD is derived from a 1st order Taylor

$$f(x) \approx f(x^{(t)}) + \nabla f(x^{(t)})^T (x - x^{(t)})$$

If we use a 2nd order Taylor, we get Newton's Method

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

We expect $\nabla f = 0$ at a minimum, so we take derivatives at both sides

$$\nabla f(x) \approx 0 + \nabla f(x^{(t)}) + \nabla^2 f(x^{(t)})(x - x^{(t)})$$

and setting $\nabla f = 0$ gives

$$x - x^{(t)} \approx -[\nabla^2 f(x^{(t)})]^{-1} \nabla f(x^{(t)})$$

at a minimizer

Theorem 12.1 (Newton's Method) *In order to solve* $\min f(x)$*, we run the iterations*

$$x^{(t+1)} = x^{(t)} - [\nabla^2 f(x^{(t)})]^{-1} \nabla f(x^{(t)})$$

e.g. $f(x) = x - \ln(x)$. Then $\nabla f(x) = 1 - \frac{1}{x}$ and $\nabla^2 f(x) = \frac{1}{x^2}$. Newton's method with initial point at $x^{(0)} = 0.5$ gives $x^{(t+1)} = 2x^{(t)} - (x^{(t)})^2$.

Definition 12.2 (Matrix Norm) *For a positive semi-definite matrix* M

$$\|M\| = \max_{x \neq 0} \frac{\|Mx\|}{\|x\|}$$

This theorem gives $\|Mz\| \leq \|M\|\|z\| \leq \lambda \|z\|$ where λ is the max eigenvalue of M.

Theorem 12.3 (Convergence of Newton's Method) *Let* f *be twice continuously differentiable and suppose that* x^* *has* $\nabla f(x^*) = 0$*. Suppose that for some* $h > 0$ *and all* x

$$\|\nabla^2 f\,(x^*)^{-1}\,\| \leq \frac{1}{h}$$

$$\|\nabla^2 f(x) - \nabla^2 f(x^*)\| \leq L\|x - x^*\|$$

Then if

$$\|x^{(0)} - x^*\| \leq \frac{2h}{3L} \text{ and } x^{(t+1)} = x^{(t)} - [\nabla^2 f(x^{(t)})]^{-1} \nabla f(x^{(t)})$$

We have $\forall t$

$$\|x^{(t)} - x^*\| \leq \frac{2h}{3L}$$

$$\|x^{(t+1)} - x^*\| \leq \frac{3L}{2h}\|x^{(t)} - x^*\|^2$$

Interpretation: If we start close to a local minimizer and f is nice, we converge quickly to the minimum.

e.g. $f(x) = x_1^4 + 2x_1^2x_2^2 + x_2^4$. We have $\nabla f(x) = \begin{pmatrix} 4x_1^3 + 4x_1x_2^2 \\ 4x_1^2x_2 + 4x_2^3 \end{pmatrix}$ and $\nabla^2 f(x) = \begin{pmatrix} 12x_1^2 + 4x_2^2 & 8x_1x_2 \\ 8x_1x_2 & 4x_1^2 + 12x_2^2 \end{pmatrix}$.

Start with $x^{(0)} = (1, 1)$

MATH173A Optimization **Fall 2020**

Lecture 13: Accelerating GD

Lecturer: Rayan Saab *Scribes: Rabbittac*

e.g. $f(x) = \frac{x^4}{4} - x^2 + 2x + 1$. Start newton's method at $x^{(0)} = 0$. $\nabla f(x) = x^3 - 2x + 2$, $\nabla^2 f(x) = 3x^2 - 2$. $x^{(1)} = x^{(0)} - \left[f''(x^{(0)})\right]^{-1} f'(x^{(0)}) = 1$. $x^{(2)} = x^{(1)} - \left[f''(x^{(1)})\right]^{-1} f'(x^{(1)}) = 0 = x^{(0)}$.

From this example, we see that Newton's method does not always converge.

Remarks:

- The theorem tells that Newton's method can converge very fast (in terms of number of iterations)
- Finding the inverse of the Hessian is computationally complex if n is large.

  Instead, we have following observations:

$$\begin{aligned} x^{(t+1)} &= x^{(t)} - \left[\nabla^2 f(x^{(t)})\right]^{-1} \nabla f(x^{(t)}) \\ \nabla f(x^{(t)}) &= \nabla^2 f(x^{(t)})(x^{(t+1)} - x^{(t)}) \\ \nabla^2 f(x^{(t)}) x^{(t+1)} &= \nabla^2 f(x^{(t)}) x^{(t)} - \nabla f(x^{(t)}) \end{aligned}$$

- We can modify Newton's method by adding a step size:

$$x^{(t+1)} = x^{(t)} - \mu^{(t)} \left[\nabla^2 f(x^{(t)})\right]^{-1} \nabla f(x^{(t)})$$

Recall that GD has an interpretation whereby

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

Minimizing RHS gives

$$x^{(t+1)} = x^{(t)} - \mu^{(t)} \nabla f(x^{(t)})$$

Meanwhile Newton's method approximates

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

As before, minimizing RHS gives us

$$x^{(t+1)} = x^{(t)} - \left[\nabla^2 f(x^{(t)})\right]^{-1} \nabla f(x^{(t)})$$

Theorem 13.1 (Quasi-Newton Methods) ***Quasi-Newton Methods*** *approximates the Hessian with some matrix $B^{(t)}$ which change in each iteration so that*

$$x^{(t+1)} = x^{(t)} - \mu^{(t)} \left[B^{(t)}\right]^{-1} \nabla f(x^{(t)})$$

e.g. There are several choices of $B^{(t)}$ such as BFGS method, Broyden method ...

Accelerating GD:
- GD with momentum
- GD with Nesterov's acceleration

Consider the following idea:

Definition 13.2 (GD with Momentum) *Adding the momentum term $\beta(x^{(t)} - x^{(t-1)})$ to the GD gives* [7]

$$x^{(t+1)} = x^{(t)} - \mu\nabla f(x^{(t)}) + \beta(x^{(t)} - x^{(t-1)})$$

Intuition: If $-\nabla f(x^{(t)})$ happens to be in the same direction as $x^{(t)} - x^{(t-1)}$ (the previous step) move further in that direction. Otherwise, if they are in opposite direction, move less from that direction.

e.g. Consider $f : \mathbb{R} \to \mathbb{R}$ that $f(x) = \frac{\lambda}{2}x^2$. Here momentum gives $x^{(t+1)} = x^{(t)} - \mu\lambda x^{(t)} + \beta(x^{(t)} - x^{(t-1)}) = (1 + \beta - \lambda\mu)x^{(t)} - \beta x^{(t-1)}$. This can be written $\begin{bmatrix} x^{(t+1)} \\ x^{(t)} \end{bmatrix} = M \begin{bmatrix} x^{(t)} \\ x^{(t-1)} \end{bmatrix}$ where $M = \begin{bmatrix} 1 + \beta - \lambda\mu & -\beta \\ 1 & 0 \end{bmatrix}$. M has eigenvalues $\frac{(1+\beta-\mu\lambda) \pm \sqrt{(1+\beta-\mu\lambda)^2 - 4\beta}}{2}$. This implies [8] $(x^{(t+1)})^2 \leq J\beta^t$ where J is a junk term. This tells that momentum converges at a rate of β^t to the solution in this example.

[7] This is also known as the "Heavy Ball Method" or "Polyak Momentum".
[8] This is beyond the scope of this course.

MATH173A Optimization **Fall 2020**

Lecture 14: GD with Momentum for Quadratics

Lecturer: Rayan Saab *Scribes: Rabbittac*

Theorem 14.1 *The optimal convergence rate of gradient descent for* $f(x) = \frac{1}{2}x^T Ax$ *is* $\dfrac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}$.

Proof: Start with GD and $f(x) = \frac{1}{2}x^T Ax$ where A is a symmetric positive semi-definite matrix. The GD performs the iterations

$$x^{(t+1)} = x^{(t)} - \mu A x^{(t)} = (I - \mu A)x^{(t)}$$

How does GD converge in this case?

$$\begin{aligned} x^{(t+1)} &= (I - \mu A)x^{(t)} \\ &= (I - \mu A)^2 x^{(t-1)} \\ &= (I - \mu A)^{t+1} x^{(0)} \end{aligned}$$

And consider $\|x^{(t+1)} - x^*\|$

$$\begin{aligned} \|x^{(t+1)} - x^*\| &= \|x^{(t+1)} - 0\| \\ &= \|(I - \mu A)^{t+1} x^{(0)}\| \\ &\leq \|(I - \mu A)^{t+1}\| \|x^{(0)}\| \end{aligned}$$

Let λ_i be eigenvalues of A, then

$$\begin{aligned} \|x^{(t+1)} - x^*\| &\leq \|(I - \mu A)^{t+1}\| \|x^{(0)}\| \\ &= \max_i |1 - \mu\lambda_i|^{t+1} \|x^{(0)}\| \\ &= \max\{1 - \mu\lambda_{\min}, \mu\lambda_{\max} - 1\}^{t+1} \|x^{(0)}\| \end{aligned}$$

Hence, we expect $\max\{1 - \mu\lambda_{\min}, \mu\lambda_{\max} - 1\}$ to be small. It turns out the optimal choice of μ is

$$\mu^* = \frac{2}{\lambda_{\max} + \lambda_{\min}}$$

and the corresponding rate of convergence is $\dfrac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}$. ■

Definition 14.2 (Condition Number) *The condition number of a matrix is defined*

$$\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}$$

Then the optimal convergence rate of GD can be written as $\dfrac{\kappa - 1}{\kappa + 1}$ which means that when κ is large, the convergence is slow.

Theorem 14.3 *The optimal convergence rate of gradient descent with momentum for* $f(x) = \frac{1}{2}x^T Ax$ *is* $\sqrt{\beta}$ *with* $\sqrt{\beta} = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$.

Therefore, we show that the convergence os accelerated compared to gradient descent.

Definition 14.4 (Nesterov's Acceleration) *Nesterov's Acceleration updates by*

$$y^{(t)} = x^{(t)} + \beta(x^{(t)} - x^{(t-1)})$$
$$x^{(t+1)} = y^{(t)} - \mu\nabla f(y^{(t)})$$

Interpretation:
- Take a "momentum step" so you get $y^{(t)}$
- Take a GD step from $y^{(t)}$

We can combine two equations to get

$$x^{(t+1)} = x^{(t)} + \beta(x^{(t)} - x^{(t-1)}) - \mu\nabla f(x^{(t)} + \beta(x^{(t)} - x^{(t-1)}))$$

Theorem 14.5 *The optimal convergence rate of gradient descent with Nesterov's acceleration for* $f(x) = \frac{1}{2}x^TAx$ *is* $\sqrt{\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}}}$ *with the optimal choice of* $\mu = \frac{1}{\lambda_{\max}}$ *and* $\beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$.

MATH173A Optimization **Fall 2020**

Lecture 15: Conjugate Gradient

Lecturer: Rayan Saab *Scribes: Rabbittac*

History:
- Originally developed in the 50's by Hestenes and Stiefel for solving large linear system $Ax = b$.
- First nonlinear conjugate gradient (CG) methods were developed in the 60's by Fletcher and Reeves for solving nonlinear optimization problems.

Suppose we want to minimize

$$\Phi(x) = \frac{1}{2}x^T Ax - b^T x$$

where A is symmetric positive semi-definite $n \times n$ matrix.

Since Φ is convex and $\nabla\Phi(x) = Ax - b$, then the optimizer satisfies $Ax^* = b$, which links solving linear system.

Definition 15.1 (Conjugate) *A set of vectors $\{P_1, \dots, P_\ell\}$ is conjugate with respect to a symmetric positive definite matrix A if*

$$\forall i \neq j, \quad P_i^T A P_j = 0$$

e.g. The standard basis vector $e_1, \dots, e_\ell \in \mathbb{R}^n$ where $\ell \neq n$ are conjugate with respect to I.

Theorem 15.2 (Conjugate Direction Method) *We can minimize $\Phi(x)$ by minimizing it along the individual direction in a conjugate set*

$$x^{(t+1)} = x^{(t)} + \alpha_t P_t$$

where $\alpha_t = \arg\min_{a\in\mathbb{R}} \Phi(x^{(t)} + aP_t)$.

Proof: Since Φ is quadratic, we can solve for α_t exactly by

$$\Phi(x + ap) = \frac{1}{2}(x + ap)^T A(x + ap) - (x + ap)^T b$$

To find optimal a, solving $\frac{\mathrm{d}\Phi}{\mathrm{d}a} = 0$ this gives $\Phi'(x+ap)\cdot(x+ap)' = 0$, then $(A(x+ap)-b)^T p = 0$. We obtain

$$a^* = \frac{\left(b - Ax^{(t)}\right)^T P_t}{P_t^T A P_t} = \frac{-\nabla\Phi(x^{(t)})P_t}{P_t^T A P_t}$$

■

i.e. This is picking the best a by performing the exact line search.

Theorem 15.3 *For any $x^{(0)}$ in the sequence $\{x^{(t)}\}$ generated by conjugate direction method converges to x^* using at most n steps* [9].

[9] The proof seen Nocedal & Wright 2006 in chapter 5

Conjugate Gradient Method:
- Compute P_t in the algorithm unlike the conjugate direction method
- Compute P_t using only P_{t-1}, no need to store $P_0, \ldots, P_{t-2}$
- P_t is conjugate to $P_0, P_1, \ldots, P_{t-1}$

Theorem 15.4 (Conjugate Gradient Method) *Conjugate gradient updates P_t by*

$$P_t = -\nabla\Phi(x^{(t)}) + \beta_t P_{t-1}$$

where β_t is

$$\beta_t = \frac{P_{t-1}^T A \nabla\Phi\left(x^{(t)}\right)}{P_{t-1}^T A P_{t-1}}$$

Proof: We want β_t chosen to enforce conjugacy, then

$$P_{t-1}^T A P_t = -P_{t-1}^T A \nabla\Phi(x^{(t)}) + \beta_t P_{t-1}^T A P_{t-1}$$
$$\beta_t = \frac{P_{t-1}^T A \nabla\Phi\left(x^{(t)}\right)}{P_{t-1}^T A P_{t-1}}$$

■

Algorithm 15.1 Conjugate Gradient Method Version 0

Initialize $x^{(0)}$ and $P_0 = \nabla\Phi(x^{(0)})$
for $t = 1, 2, \ldots$ **do**
$\beta_t = \dfrac{P_{t-1}^T A \nabla\Phi\left(x^{(t)}\right)}{P_{t-1}^T A P_{t-1}}$
$P_t = -\nabla\Phi(x^{(t)}) + \beta_t P_{t-1}$
$\alpha_t = -\dfrac{\nabla\Phi(x^{(t)})^T P_t}{P_t^T A P_t}$
$x^{(t+1)} = x^{(t)} + \alpha_t P_t$
end for

A more efficient implementation requires properties of conjugate gradient, which are not covered here:

Algorithm 15.2 Conjugate Gradient Method Version 1

Initialize $x^{(0)}, r_0 = Ax^{(0)} - b = \nabla\Phi(x^{(0)})$ and $P_0 = r_0$
for $t = 1, 2, \ldots$ **do**
$\alpha_t = \dfrac{r_t^T r_t}{P_t^T A P_t}$
$x^{(t+1)} = x^{(t)} + \alpha_t P_t$
$r_{t+1} = r_t + \alpha_t A P_t$
$\beta_{t+1} = \dfrac{r_{t+1}^T r_{t+1}}{r_t^T r_t}$
$P_{t+1} = -r_{t+1} + \beta_{t+1} P_t$
end for

The cost per iteration is AP_t, $P_t^T(AP_t)$, and $r^T r$, which is not much more expensive than GD.

MATH173A Optimization **Fall 2020**

Lecture 16: Preconditioned CG

Lecturer: Rayan Saab *Scribes: Rabbittac*

We want to improve CG in the quadratic setting.

Idea: – Preconditioning

Background:
- We seek $x^* : Ax^* = b$ where A is symmetric positive definite
- Performance of CG depends on the eigenvalues of A, define $\|z\|_A = \sqrt{z^T Az}$

Theorem 16.1 *If a matrix A has eigenvalues $0 < \lambda_1 \leq \cdots \leq \lambda_n$, then*

$$\|x^{(t+1)} - x^*\|_A^2 \leq (\frac{\lambda_{n-t} - \lambda_1}{\lambda_{n-t} + \lambda_1})^2 \|x^{(0)} - x^*\|_A^2$$

Note: Pick $t + 1 = n$, then $\|x^{(n)} - x^*\| \leq 0$ which means converge in n steps. Additionally, if we pick $\lambda_{n-1} = \lambda_n$, then converge of x^* takes only $n - 1$ steps. It is also true that

$$\|x^{(t+1)} - x^*\|_A \leq 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^t \|x^{(0)} - x^*\|_A$$

The idea of **preconditioning** to convert the problem into an equivalent one that has better eigenvalues

e.g. Given $\Phi(x) = \frac{1}{2} x^T Ax - b^T x$. Define[10] $\hat{x} = Cx$ and $\hat{\Phi}(\hat{x}) = \frac{1}{2} \hat{x}^T (\hat{C}^{-T} A C^{-1}) \hat{x} - b^T C^{-1} \hat{x} = \frac{1}{2} \hat{x}^T \hat{A} \hat{x} - \hat{b}^T \hat{x}$

Theorem 16.2 (Preconditioned CG) *Toward a modification of conjugate gradient method, we introduce C such that*

$$\kappa(C^{-T} A C^{-1}) < \kappa(A)$$

A detailed algorithm is describes in 16.1.

Note: If $M = I$ then this turns into CG. This precondition increases computational costs to solve $My_t = r_t$ at each step.

In order to reduce costs of solving $My = r$, we design M such that $My = r$ can be solved quickly while having favorable properties. For example, we can pick $M = \tilde{L}\tilde{L}^T$ where L is a sparse approximation to L which is Cholesky factor of A. Therefore, $C^{-T} A C^{-1} \approx -\tilde{L}^{-T} A \tilde{L}^{-1} \approx I$ which has low condition number.

Now we turn to nonlinear CG where we replace quadratic function $\Phi(x)$ with general function $F(x)$. The Fletcher-Reeves version of CG is describes in 16.2.

Note: This only needs gradient evaluations and inner products, which have similar costs to GD.

[10] *Notation:* $C^{-T} = (C^{-1})^T$

Algorithm 16.1 Preconditioned CG

Given $M = C^T C$
Initialize $x^{(0)}, r_0 = Ax^{(0)} - b = \nabla\Phi(x_0)$
Solve $My_0 = r_0$ for y_0
Set $P_0 = -y_0$
while $r_t \neq 0$ or too big **do**
  $\alpha_t = \frac{r_t^T y_t}{P_t^T A P_t}$
  $x^{(t+1)} = x^{(t)} + \alpha_t P_t$
  $r_{t+1} = r_t + \alpha_t A P_t$
  Solve $My_{t+1} = r_{t+1}$ for y_{t+1}
  $\beta_{t+1} = \frac{r_{t+1}^T y_{t+1}}{r_t^T y_t}$
  $P_{t+1} = -y_{t+1} + \beta_{t+1} P_t$
end while

Algorithm 16.2 Preconditioned Fletcher-Reeves CG

Initialize $x^{(0)}$. Set $F_0 = F(x^{(0)})$, $\nabla F_0 = \nabla F\left(x^0\right)$, $P_0 = -\nabla F_0$.
while $\nabla F_t \neq 0$, or DF_t too big **do**
  Find α_t using line-search
  Set $x^{(t+1)} = x^{(t)} + \alpha_t P_t$
  Evaluate $\nabla F_{t+1} = \nabla F\left(x^{(t+1)}\right)$
  $\beta_{t+1}^{FR} = \frac{\nabla F_{t+1}^T \nabla F_{t+1}}{\nabla F_t^T \nabla F_t}$
  $P_{t+1} = -\nabla F_{t+1} + \beta_{t+1}^{FR} P_t$
end while

But now the question is to pick α_t. The issue is that for P_t to be a descent direction, $\nabla F_t^T p_t = -\|\nabla F_t\|^2 + \beta_{t+1}^{FR} \nabla F_t^T P_{t-1}$ must be negative. If α_t minimizes $F(x^{(t)} + \alpha P_{t-1})$ then by chain rule $\nabla F_t^T P_{t-1} = 0$. Hence $\nabla F_t^T P_t < 0$. But $F(x^{(t)} + \alpha P_{t-1})$ may be difficult to minimize so α_t, coming from a line search may not guarantee that $\nabla F_t^T P_t < 0$. To solve that, ensure Wolfe's conditions are satisfied when solving for α_t

$$F(x^{(t)} + \alpha_t P_t) \leq F(x^{(t)}) + C_1 \alpha_t \nabla F_t^T P_t$$

$$\left|\nabla F(x^{(t)} + \alpha_t P_t)^T P_t\right| \leq -C_2 \nabla F_t^T P_t$$

where $0 < C_1 < C_2 < \frac{1}{2}$. Then we can show that $\nabla F_t^T P_t < 0$.

Other versions of FR-CG choose β differently.

e.g. Polak–Ribiére: $\beta_{t+1}^{TR} = \frac{\nabla F_{t+1}^T \left(\nabla F_{t+1}^T - \nabla F_t^T\right)}{\|\nabla F_t\|^2}$. This does not guarantee P_t is a descent direction. However, $\beta_{t+1}^{+} = \max\{\beta_{t+1}^{PR}, 0 =\}$ fixes this.

MATH173A Optimization **Fall 2020**

Lecture 17: Strong Convex

Lecturer: Rayan Saab *Scribes: Rabbittac*

Definition 17.1 (Strong Convex) *A function F is strong convex if $\exists c > 0$ such that $\forall x, y \in \mathbb{R}^d$*

$$F(y) \geqslant F(x) + \nabla F(x)^T (y - x) + \frac{1}{2} c \|y - x\|^2$$

and it guarantees[11]

$$(F(x) - F(x^*)) \leq \frac{\|\nabla F(x)\|^2}{2c}$$

Strong convexity means that there exists a quadratic lower bound on the growth of the function.

A simple analysis of GD: assume we start with $\|\nabla^2 F\| \leq L$

$$\begin{aligned} F(x^{(t+1)}) &= F(x^{(t)} - \alpha_t \nabla F(x^{(t)})) \\ &= F(x^{(t)}) - h^T \nabla F_{(x^{(t)})} + \frac{1}{2} h^T \nabla^2 F(\xi) h \end{aligned}$$

$h^T \nabla^2 F(\xi) h = \langle h, \nabla^2 F(\xi) h \rangle \leq \|h\| \|\nabla^2 F(\xi) h\| \leq \|h\|^2 \|\nabla^2 F(\xi)\| \leq L \|h\|^2$ So

$$\begin{aligned} F(x^{(t+1)}) &\leq F(x^{(t)}) - \alpha_t \|\nabla F(x^{(t)})\|^2 + \frac{L \alpha_t^2}{2} \|\nabla F(x^{(t)})\|^2 \\ &= F(x^{(t)}) - \alpha_t (1 - \frac{L \alpha_t}{2}) \|\nabla F(x^{(t)})\|^2 \\ &= F(x^{(t)}) - \frac{\alpha}{2} \|\nabla F(x^{(t)})\|^2 \end{aligned}$$

Then we have two possibilities

1. Assume NO strong convexity: then taking telescope sum gives

$$\begin{aligned} \frac{\alpha}{2} \sum_{t=0}^{T-1} \|\nabla F(x^{(t)})\|^2 &\leqslant \sum_{t=0}^{T-1} [F(x^{(t)}) - F(x^{(t+1)})] \\ &= F(x^{(0)}) - F(x^{(T)}) \\ &\leq F(x^{(0)}) - F(x^*) \end{aligned}$$

   Thus as $T \to \infty : \|\nabla F(x^{(T)})\|^2 \to 0$

2. Assume strong convexity with constant C so $F(x+h) \geq F(x) + h^T \nabla F(x) + \frac{C}{2} \|h\|^2$ and strong convexity also gives us[12]

$$\|\nabla F(z)\|^2 \geq 2C[F(z) - F(x^*)]$$

[11] Proof is in Hw7
[12] Proof is in Hw7

Use alternative definition

$$\begin{aligned} F(x^{(t+1)}) &\le F(x^{(t)}) - \frac{\alpha}{2}\|\nabla F(x^{(t)})\|^2 \\ F(x^{(t+1)}) - F(x^*) &\le F(x^{(t)}) - F(x^*) - \frac{\alpha}{2}\|\nabla F(x^{(t)})\|^2 \\ &\le F(x^{(t)}) - F(x^*) - \frac{\alpha}{2}\cdot 2C \cdot [F(x^{(t)}) - F(x^*)] \\ &\le (1 - \frac{C}{L})[F(x^{(t)}) - F(x^*)] \\ &\le (1 - \frac{C}{L})^{t+1}[F(x^{(0)}) - F(x^*)] \end{aligned}$$

Theorem 17.2 *GD : with strong convexity*
with $\left\|\nabla^2 F\right\| \le L$
with $\alpha = \frac{1}{L}$

guarantees

$$F(x^{(t)}) - F(x^*) \le (-\frac{C}{L})^t[F(x^{(0)}) - F(x^*)]$$

To solve the condition number, $\|\nabla^2 F\| \le L$ gives

$$F(x+h) \le F(x) + h^T\nabla F(x) + \frac{2}{2}\|h\|^2$$

which is quadratic in h, written as $Q_L(h)$.

$$F(x+h) \ge F(x) + h^T\nabla F(x) + \frac{c}{2}\|h\|^2$$

which is quadratic in h, written as $Q_C(h)$.

Note: Bad condition number causes slower convergence.

Corollary 17.3 (Convergence of GD with Momentum) *In the strongly convex cases, convergence of GD with momentum*

$$x^{(t+1)} = x^{(t)} - \alpha\nabla F(x^{(t)}) + \beta(x^{(t)} - x^{(t-1)})$$

is in the form

$$\beta^t\|x^{(0)} - x^*\|$$

where $\beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$.

This interprets that GD with momentum has better dependence on the condition number than GD so that GD with momentum can be faster than GD.

Corollary 17.4 (Convergence of GD with Nesterov Acceleration) *In the strongly convex cases, convergence of GD with Nesterov Acceleration*

$$x^{(t+1)} = x^{(t)} + \beta(x^{(t)} - x^{(t-1)}) - \mu\nabla f(x^{(t)} + \beta(x^{(t)} - x^{(t-1)}))$$

is in the form

$$(\sqrt{\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}}})^t$$

MATH173A Optimization **Fall 2020**

Lecture 18: Applications in DS&ML

Lecturer: Rayan Saab *Scribes: Rabbittac*

In data science or machine learning, we often need to optimize functions in this form

$$\min_{w} \frac{1}{N} \sum_{i=1}^{N} f\left(w_j; (x_i, y_i)\right) + \lambda R(\omega)$$

where w is model parameters, f is loss function, (x_i, y_i) is training data, R is regularization term.