Optimization in General - (i.e, not just Deep Learning)

A/Prof Richard Yi Da Xu
Yida.Xu@uts.edu.au
Wechat: aubedata
https://github.com/roboticcam/machine-learning-notes

University of Technology Sydney (UTS)

February 18, 2018

- **Your aim** to find:

$$\arg\min_{\mathbf{x}} (f(\mathbf{x}))$$

- How? Solve $\nabla f(\mathbf{x}_n) = 0$! But in many scenarios, this isn't easy!
- The rate of change of $f(x, y)$ in the direction of the unit vector $u = (a, b)$ is called the directional derivative $\mathrm{d}_u f(x, y)$. The definition of the directional derivative is:

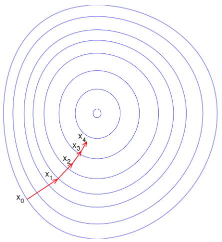
$$\mathrm{d}_u f(x, y) = \lim_{h \to 0} \frac{f(x + ah, y + bh) - f(x, y)}{h}$$

- **Theorem** the minimum directional derivative of a differentiable function f at (x_0, y_0) is $-|\triangledown f(x_0, y_0)|$ and occurs for u with the opposite direction as $\triangledown f(x_0, y_0)$

Gradient Descend

Here is where **Gradient Descend** algorithm may help. The iterative algorithm looks something like:

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \alpha_n \nabla f(\mathbf{x}_n), \qquad n \geq 0$$



Moral of the story, you must know how to compute the objective function's **derivative**.

Newton methods

- taylor expansion of $f(\mathbf{x})$ around $\mathbf{x}_n$ in 1-D:

$$f(x_n + \Delta x) \approx f(x_n) + f'(x_n)\Delta x + \frac{1}{2} f''(x_n)\Delta x^2$$

- we need to find what is the "right" value of Δx that minimises $f(.)$:

$$\begin{aligned}
\frac{\mathrm{d}f(x_n + \Delta x)}{\mathrm{d}\Delta x} &= \frac{\mathrm{d}}{\mathrm{d}\Delta x}\left(f(x_n) + f'(x_n)\Delta x + \frac{1}{2} f''(x_n)\Delta x^2 \right) = f'(x_n) + f''(x_n)\Delta x \\
f'(x_n) + f''(x_n)\Delta x &= 0 \implies \Delta x = \frac{-f'(x_n)}{f''(x_n)} \\
x_{n+1} &= x_n + \Delta x \\
&= x_n - (f''(x_n))^{-1} f'(x_n)
\end{aligned}$$

- taylor expansion of $f(\mathbf{x})$ around $\mathbf{x}_n$ in higher dimension:

$$\implies \mathbf{x}_{n+1} = \mathbf{x}_n - \underbrace{\left(f''(\mathbf{x}_n)\right)^{-1}}_{\alpha_n} \nabla f(\mathbf{x}_n)$$

- $f''(\mathbf{x}_n)$ is called Hessian matrix.

Some basics

- **mean value theorem**:

  if $f(x)$ is defined and **continuous** on interval $[a, b]$ and differentiable on (a,b), then there is at least one number $c \in (a, b)$ s.t:

$$f'(c) = \frac{f(b) - f(a)}{b - a}$$

- **matrix norm**:

$$\begin{aligned} \|A\| &= \sup\{\|Ax\| : x \in K^n \text{ with } \|x\| = 1\} \\ &= \sup\left\{\frac{\|Ax\|}{\|x\|} : x \in K^n \text{ with } x \neq 0\right\} \end{aligned}$$

- when B is symmetric matrix, $\|B\| = \max\{\lambda_i(B)\}$

To prove A is symmetric matrix $\implies \|A\| = \max\{\lambda_i(A)\}$

- matrix $B = A^\top A$ is **symmetric** matrix
- **fact**: any symmetric matrix, there is an orthonormal basis of eigenvectors $\{b_i\}_{i=1}^n$, with real eigenvalues $\{\lambda_i\}_{i=1}^n$

$$Bb_i = \lambda_i b_i$$

- $B = A^\top A \implies \lambda_i$ must be non-negative real numbers, since we can write:

$$\begin{aligned} b_i^\top B b_i &= b_i^\top \lambda_i b_i = \lambda_i \\ &= b_i^T A^T A b_i = (Ab_i)^T A b_i = ||Ab_i||_2^2 \geq 0 \end{aligned}$$

- **unit vectors** x, i.e., $||x||_2 = 1$ can also be written as:

$$\left\{ x : x = \sum_{i=1}^n y_i b_i, \text{ with} \sum_{i=1}^n y_i^2 = 1 \right\}$$

this is because:

$$\begin{aligned} &\left(y_1 b_1^\top + y_2 b_2^\top + \cdots + y_n b_n^\top\right)(y_1 b_1 + y_2 b_2 + \cdots + y_n b_n) \\ =& y_1 b_1^\top (y_1 b_1 + y_2 b_2 + \cdots + y_n b_n) + \cdots + y_n b_n^\top (y_1 b_1 + y_2 b_2 + \cdots + y_n b_n) \\ =& y_1^2 + \cdots + y_n^2 \\ =& 1 \end{aligned}$$

To prove A is symmetric matrix $\implies \|A\| = \max\{\lambda_i(A)\}$

- in a same way, we can write:

$$
\begin{aligned}
x^\top(A^\top A)x = x^\top Bx &= \sum_{i=1}^{n} \lambda_i y_i^2 \\
&= \left(\sum_{i=1}^{n} y_i b_i,\right)^\top B \left(\sum_{i=1}^{n} y_i b_i,\right) \\
&= \sum_{i=1}^{n} y_i b_i^\top B y_i b_i = \sum_{i=1}^{n} y_i^2 b_i^\top B b_i \\
&= \sum_{i=1}^{n} \lambda_i y_i^2
\end{aligned}
$$

We can now rewrite the 2-norm squared of A as

$$
\begin{aligned}
\|A\|_2^2 &= \max_{\{x:\|x\|=1\}} \left\{\|Ax\|_2^2\right\} = \max_{\{x:\|x\|=1\}} \left\{x^\top(A^\top A)x\right\} \\
&= \max_{\left\{\{y_1,\dots,y_n\}\text{s.t.} \sum y_i^2=1\right\}} \sum_{i=1}^{n} \lambda_i y_i^2 \\
&= \max\{\lambda_i\}
\end{aligned}
$$

the above must occur when the y_i correspond to $\max\{\lambda_i\}$ is one, and the rest $\{y_i\}$ are zeros.

Convexity conditions

- **zero-order condition**: line above curve

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y) \qquad \forall x, y \in \mathbb{R}^n \qquad \forall 0 \leq \theta \leq 1$$

- **first-order condition**: curve globally above tangent

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

- **second-order condition**: curve flat or curved upwards in every direction

$$0 \preceq \nabla^2 f(x) \qquad \forall x \in \mathbb{R}^n$$

exercise which convex function generates a flat (constant) $\nabla^2 f(x)$

Monotonicity of gradient

- f is said to be **monotone** (non-decreasing) if $\forall (x, y), (x', y') \in \mathbb{R}^2$:

$$(x \leq x' \text{ AND } y \leq y') \implies f(x, y) \leq f(x', y')$$

  think about the case of fixing one variable $y = y'$
- **exercise** does it imply monotone in both x and y?
- then $f : \mathbb{R}^n \to \mathbb{R}^n$ is monotone mapping:

$$(f(x) - f(y))^\top (x - y) \geq 0$$

  visualised by drawing two separate vectors $x - y$ and $f(x) - f(y)$: both needs to be in the same quadrant
- likewise, $\nabla f : \mathbb{R}^n \to \mathbb{R}^n$ is monotone mapping::

$$(\nabla f(x) - \nabla f(y))^\top (x - y) \geq 0$$

  think it in terms of $\nabla^2 f$ being positive

proof of Monotonicity of gradient

- if f is differentiable and **convex**, then using **first-order condition**:

$$f(y) \geq f(x) + \nabla f(x)^{\top}(y-x) \qquad f(x) \geq f(y) + \nabla f(y)^{\top}(x-y)$$

- then the proof is:

$$-f(x) \geq -f(y) + \nabla f(x)^{\top}(y-x) \qquad f(x) \geq f(y) + \nabla f(y)^{\top}(x-y)$$

add them up:

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

f is Lipschitz-continuous

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

this means that function f **can not** change too quickly:

- consider l_2-regularized logistic regression, change usual notation $\theta \to x$, and $x_i \to d_i$

$$f(x) = \sum_{i=1}^{n} \log\left(1 + \exp(-y_i(x^\top d_i))\right) + \frac{\lambda}{2}\|x\|^2$$

- f(x) is convex
- first term **is** Lipschitz continuous, second term **is not**.
- $\mu I \preceq \nabla^2 f(x) \preceq LI$ where $L = \frac{1}{4}\|A\|_2^2 + \lambda$ and $\mu = \lambda$
- gradient is Lipschitz-continuous
- function is strongly-convex

Lipschitz-Continuous **Gradient**

- taylor expansion: for some z:

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

- this does not look like the **usual** taylor expansion.
- remember the mean-value theorem:

$$f'(c) = \frac{f(b) - f(a)}{b - a} \implies f(b) = f(a) + f'(c)(b - a)$$

- **mean value theorem** only gives the existence of such a point c, and not a method for how to find c

$$f(b) = [f(a) + f'(a)(b - a)] + \frac{f''(c)}{2}(b - a)^2$$

Lipschitz-Continuous **Gradient**

- taylor expansion: for some z:

$$
\begin{aligned}
f(y) &= f(x) + \nabla f(x)^\top (y-x) + \frac{1}{2}(y-x)^\top \nabla^2 f(z)(y-x) \\
&\le f(x) + \nabla f(x)^\top (y-x) + \frac{L}{2}\|y-x\|^2
\end{aligned}
$$

- because

$$\nabla^2 f(z) \preceq LI$$

- $\nabla^2 f(z)$ is a symmetric positive defintite matrix, means that

$$
\begin{aligned}
&\nabla^2 f(z) \preceq LI \implies \|\nabla^2 f(z)\| \le \|LI\| \implies \|\nabla^2 f(z)\| \le L \\
&\implies \underbrace{\|\nabla^2 f(z)\| = \frac{\|\nabla f(x) - \nabla f(y)\|}{\|x-y\|}}_{\text{mean value theorem}} \le L \\
&\implies \underbrace{\|\nabla f(x) - \nabla f(y)\| \le L\|x-y\|}_{\text{gradient of Lipschitz-continous function will change at least } L}
\end{aligned}
$$

optimisation using Lipschitz-Continuous **Gradient**

- optimising the upperbound:

$$
\begin{aligned}
f(y) &\le f(x) + \nabla f(x)^\top (y - x) + \frac{L}{2}\|y - x\|^2 \\
\frac{\mathrm{d}f(y)}{\mathrm{d}y} &= \nabla f(x) + L(y - x) = 0 \\
\implies Ly &= Lx - \nabla f(x) \\
\implies y &= x - \frac{1}{L}\nabla f(x)
\end{aligned}
$$

- how much do we reduce? substitute $y = x - \frac{1}{L}\nabla f(x)$ into $f(y)$:
- $\nabla^2 f(z)$ is a symmetric positive defintite matrix, means that

$$
\begin{aligned}
f(y) &\le f(x) + \nabla f(x)^\top (y - x) + \frac{L}{2}\|y - x\|^2 \\
&= f(x) + \nabla f(x)^\top \left(x - \frac{1}{L}\nabla f(x) - x \right) + \frac{L}{2}\left\| \left(x - \frac{1}{L}\nabla f(x) \right) - x \right\|^2 \\
&= f(x) + \nabla f(x)^\top \left(-\frac{1}{L}\nabla f(x) \right) + \frac{L}{2}\left\| \frac{1}{L}\nabla f(x) \right\|^2 \\
&= f(x) - \frac{L}{2}\|\nabla f(x)\|^2 \implies f(x) - f(y) \ge \frac{L}{2}\|\nabla f(x)\|^2
\end{aligned}
$$

an update decreases at least $\frac{L}{2}\|\nabla f(x)\|^2$

Properties of Strong-Convexity

- taylor expansion: for some z:

$$
\begin{aligned}
f(y) &= f(x) + \nabla f(x)^\top (y-x) + \frac{1}{2}(y-x)^\top \nabla^2 f(\color{red}{z})(y-x) \\
&\leq f(x) + \nabla f(x)^\top (y-x) + \frac{L}{2}\|y-x\|^2
\end{aligned}
$$

- because

$$
\nabla^2 f(z) \preceq \color{blue}{LI}
$$

- Traditional gradient descent approach: $\theta_{n+1} = \theta_n - \alpha_n \left(\sum_{i=1}^N x_i^T \theta - y_i \right)$
- However, think about what if N is $1,000,000$, which happens often in the BIG DATA era.
- Stochastic Gradient Descent HELPS!

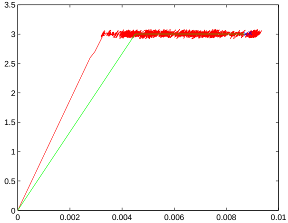
A simple example:

$$F(\theta) = \|\mathbf{x}^T\theta - \mathbf{y}\|^2 = \sum_{i=1}^{N} \left(x_i^T\theta - y_i\right)^2$$

$$\begin{aligned}\nabla F(\theta) &= 2\mathbf{x}^T(\mathbf{x}\theta - \mathbf{y}) \\ &\propto \mathbf{x}\theta - \mathbf{y} \\ &= \sum_{i=1}^{N} x_i^T\theta - y_i\end{aligned}$$

- Traditional gradient descent approach: $\theta_{n+1} = \theta_n - \alpha_n \left(\sum_{i=1}^{N} x_i^T\theta - y_i\right)$
- However, think about what if N is $1,000,000$, which happens often in the BIG DATA era.
- Stochastic Gradient Descent HELPS!

A quick demo to illustrate Stochastic Gradient Descent (2)



Idea, instead of

$$\theta_{n+1} = \theta_n - \alpha_n \left(\sum_{i=1}^{N} x_i^T \theta - y_i \right)$$

Each iteration, we select randomly a data point pair (x_j, y_j), and do:

$$\theta_{n+1} = \theta_n - \alpha_n \left(x_j^T \theta - y_j \right) \qquad j \sim U(1, \dots N)$$

It surprisingly works quite well in many settings. See demo

- The objective function:

$$E_D(\mathbf{w}) + \alpha E_W(\mathbf{w})$$

- Example:

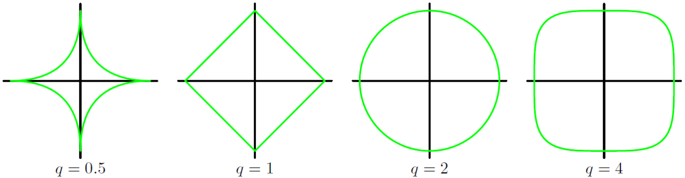
$$\frac{1}{2}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^T\phi(\mathbf{x}_n)\right)^2 + \frac{\alpha}{2}\mathbf{w}^T\mathbf{w} \implies \mathbf{w}_{\text{ML}} = \left(\alpha\mathbf{I} + \mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T\mathbf{t}$$

- A generalised example:

$$\frac{1}{2}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^T\phi(\mathbf{x}_n)\right)^2 + \frac{\alpha}{2}\sum_{j=1}^{M}|w_j|^q \implies \mathbf{w}_{\text{ML}} \text{ not so easy to obtain}$$

Diagrams of ϕ_j and struggle between $E_D(\mathbf{w})$ and $\alpha E_W(\mathbf{w})$

Plot of various norm functions: q-norm $\|\mathbf{w}\|_q := \left(\sum_{i=1}^{n} |w_i|^q \right)^{1/q} = 1$:



minimise $E_D(\mathbf{w}) + \alpha E_W(\mathbf{w})$ becomes the "tradeoff" between the two: