Convex Optimization 10-725, Lecture 13: Hessian Matrix and Preconditioned Gradient Descent

Yuanzhi Li

Assistant Professor, Carnegie Mellon University

Today

Last lecture

- We learned proximal algorithm. This was the last lecture where we focus on the optimization foundations.

This lecture

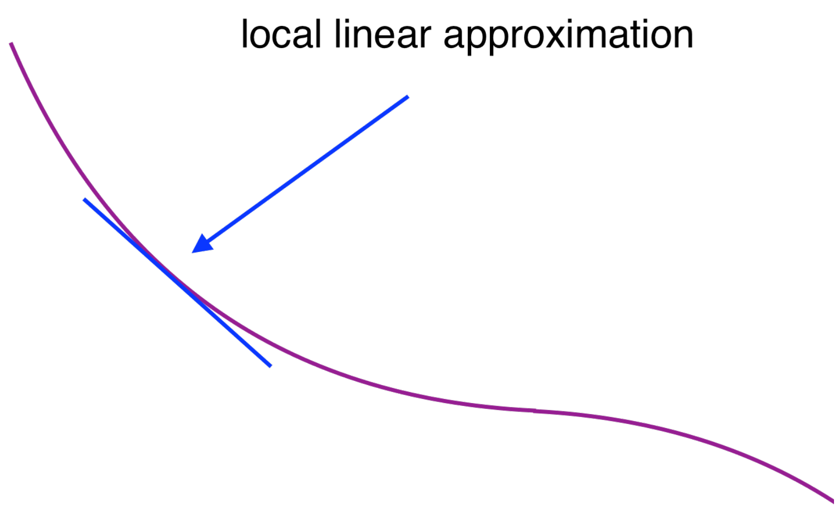
- We are going to move to the "geometry" side of convex optimization, which focus on the shape of the convex functions.
- These are the "second order methods" using the Hessian information of the convex function.

Motivation

- Gradient Descent: Looking at the first order taylor expansion of the objective function f:

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + O(\|y - x\|_2^2)$$

- *Local linear approximation* of the original function f.

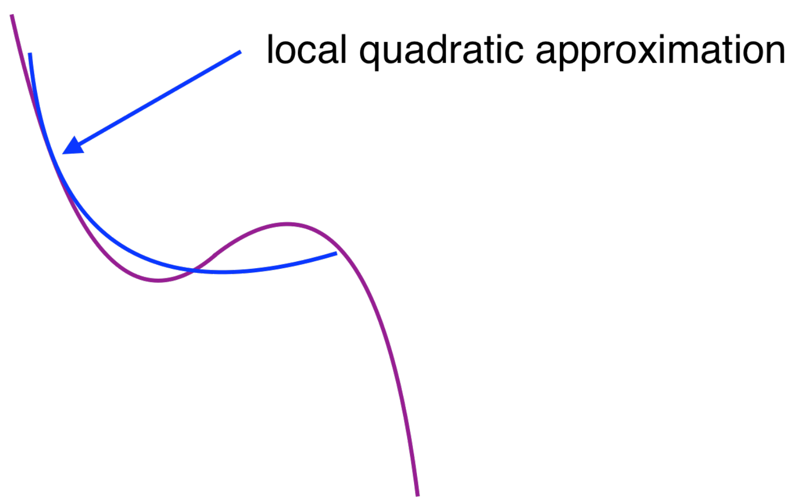


-
- This suggests us to move along the negative gradient direction $-\eta \nabla f(x)$ to decrease the objective.

Motivation

- What if we actually look at the second order taylor expansion of the objective function f?

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}(y-x)^\top \nabla^2 f(x)(y-x) + O(\|y-x\|_2^3)$$



-

Motivation

- What if we actually look at the second order taylor expansion of the objective function f?

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}(y-x)^\top \nabla^2 f(x)(y-x) + O(\|y-x\|_2^3)$$

- *Optimal* descent direction δ:

$$\min_{\delta} \{ \langle \nabla f(x), \delta \rangle + \frac{1}{2} \delta^\top \nabla^2 f(x) \delta \}$$

- When f is strictly convex at x, i.e. $\nabla^2 f(x) \succ 0$ (positive definite), there is an optimal solution for δ:

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

Motivation

- What if we actually look at the second order taylor expansion of the objective function f?

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}(y-x)^\top \nabla^2 f(x)(y-x) + O(\|y-x\|_2^3)$$

- $\nabla^2 f(x)$ gives more information about the shape of the function f around x.
- Intuitively, we have the optimal descent direction:

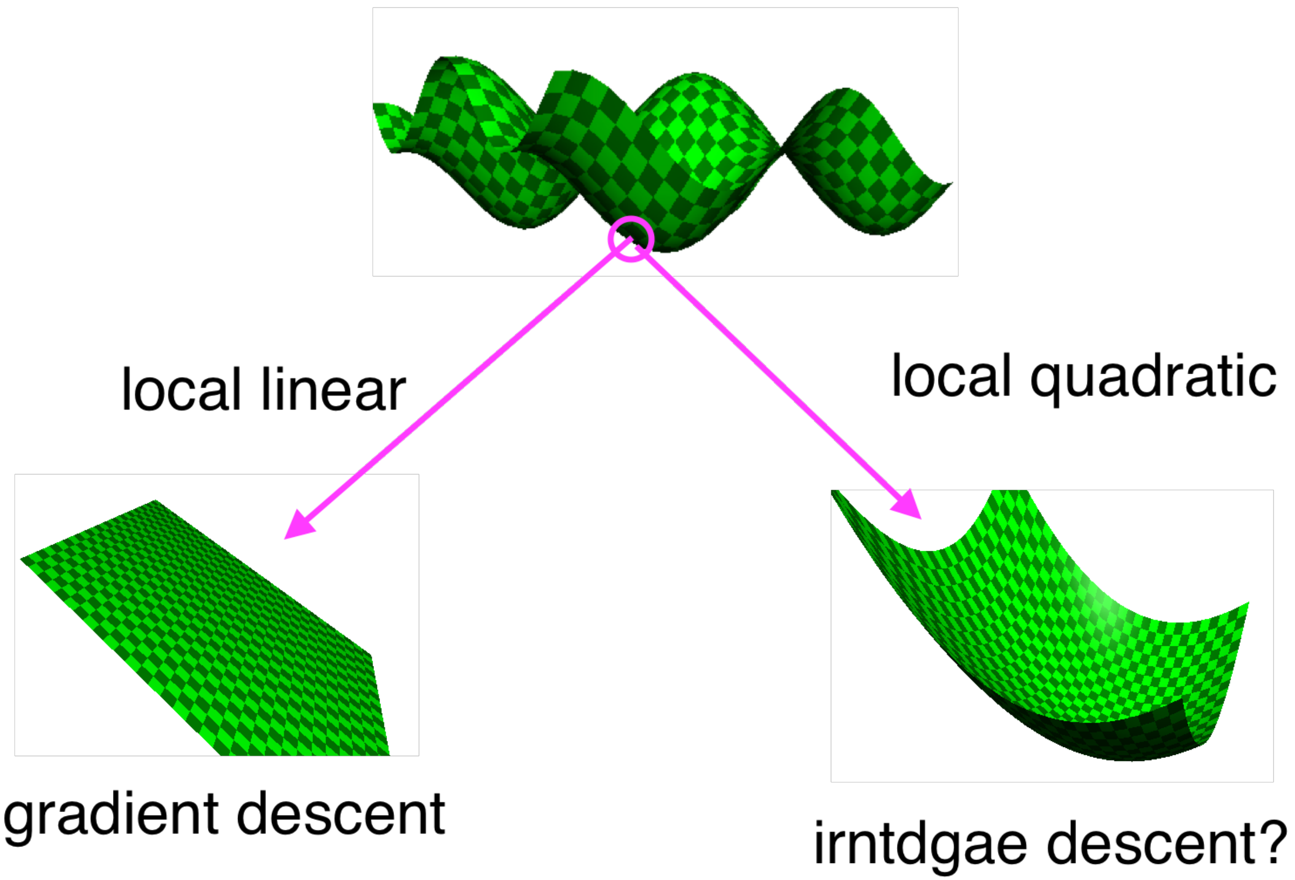
$$y = x - \eta[\nabla^2 f(x)]^{-1} \nabla f(x)$$

- How would an algorithm like this work?

Motivation

- $\nabla^2 f(x)$ gives more information about the shape of the function f around x.
- Intuitively, we have the optimal descent direction:

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



The newton's method

- Given a (second order differentiable) function f, the Newton's method is defined as:
- At every iteration t, update (typically choose $\eta = 1$):

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

- Warning: Computing $[\nabla^2 f(x_t)]^{-1}$ is typically very inefficient. Newton's method only runs fast in certain special cases.
- But first of all, does it even converge at all?

The newton's method

- 

Gradient Descent v.s. Newton

- Gradient Descent:

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + O(\|y - x\|_2^2)$$

- So when $y = x - \eta \nabla f(x)$,

$$f(y) \leq f(x) - \eta \|\nabla f(x)\|_2^2 + \eta^2 O(\|\nabla f(x)\|_2^2)$$

- So when *$\eta \leq 1/smoothness$*, Gradient Descent decreases the objective value.

Gradient Descent v.s. Newton

- Newton's Method:

$$f(y) \le f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}(y-x)^\top \nabla^2 f(x)(y-x) + O(\|y-x\|_2^3)$$

- So when $y = x - \eta[\nabla^2 f(x)]^{-1} \nabla f(x)$,

$$f(y) \le f(x) - (\eta - \eta^2/2)\nabla f(x)^\top [\nabla^2 f(x)]^{-1} \nabla f(x)$$

$$+\eta^3 O(\|[\nabla^2 f(x)]^{-1} \nabla f(x)\|_2^3)$$

- $\nabla f(x)^\top [\nabla^2 f(x)]^{-1} \nabla f(x)$ and $\|[\nabla^2 f(x)]^{-1} \nabla f(x)\|_2^3$ are not directly comparable! Unclear how to choose η globally.

Local Convergence for Newton's Method

- Newton's Method only has a local convergence guarantee.
- Theorem: Assuming x^* is a strict local minima of $f(x)$, i.e. $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*) \succeq \sigma I$ for some $\sigma > 0$.
- Assuming (Lipschitz Hessian) of f: for every x, y:

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_{spectral \ \ norm} \leq L\|x - y\|_2$$

- Then as long as $\|x_0 - x^*\|_2 \leq \frac{\sigma}{2L}$, with $\eta = 1$:

$$\|x_{t+1} - x^*\|_2 \leq \frac{2L}{\sigma}\|x_t - x^*\|_2{}^2$$

- Which is equivalent to

$$\frac{\|x_{t+1} - x^*\|_2}{\sigma/(2L)} \leq \left(\frac{\|x_t - x^*\|_2}{\sigma/(2L)}\right)^2$$

- This is known as the quadratic convergence rate of Newton's method.

Local Convergence for Newton's Method

- Newton's Method only has a local convergence guarantee.
- As long as $\|x_0 - x^*\|_2 \le \frac{\sigma}{2L}$,

$$\frac{\|x_{t+1} - x^*\|_2}{\sigma/(2L)} \le \left(\frac{\|x_t - x^*\|_2}{\sigma/(2L)}\right)^2$$

- This is known as the quadratic convergence rate of Newton's method.
- $0.9 \to 0.81 \to 0.6561 \to 0.43046721 \to 0.1853020189 \to 0.0343368382 \to 0.001179018458 \to 0.000001390084523771450\mathbf{}8 \to 0.0000000000019323349832289015$ in only 9 iterations.
- Local convergence rate of Netwon's Method is much faster than Gradient Descent, since Netwon's method uses more fine-grind *local geometry* information of the function: The Hessian.

Local Convergence for Newton's Method

- Proof: Observe that by $\nabla f(x^*) = 0$,

$$\nabla f(x_t) = \int_0^1 \nabla^2 f(x^* + s(x_t - x^*))(x_t - x^*)ds$$

- With $\eta = 1$, we can calculate that

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

$$= x_t - x^* - [\nabla^2 f(x_t)]^{-1} \int_0^1 \nabla^2 f(x^* + s(x_t - x^*))(x_t - x^*)ds$$

$$= [\nabla^2 f(x_t)]^{-1} \int_0^1 (\nabla^2 f(x_t) - \nabla^2 f(x^* + s(x_t - x^*)))(x_t - x^*)ds$$

Local Convergence for Newton's Method

- Now we have: $x_{t+1} - x^* = [\nabla^2 f(x_t)]^{-1} \int_0^1 (\nabla^2 f(x_t) - \nabla^2 f(x^* + s(x_t - x^*)))(x_t - x^*) ds$
- By Lipschitzness of the Hessian, for every $s \in [0, 1]$

$$\|\nabla^2 f(x_t) - \nabla^2 f(x^* + s(x_t - x^*))\|_{spectral\ \ norm} \le L\|x_t - x^*\|_2 \tag{1}$$

- When $\nabla^2 f(x^*) \succeq \sigma I$, and $\|x_t - x^*\|_2 \le \frac{\sigma}{2L}$, we have that $\nabla^2 f(x_t) \succeq \frac{\sigma}{2} I$.
- Therefore, we conclude that

$$\|x_{t+1} - x^*\|_2 \le (\sigma/2)^{-1} L\|x_t - x^*\|_2 \|x_t - x^*\|_2$$

- Which is

$$\|x_{t+1} - x^*\|_2 \le \frac{2L}{\sigma} \|x_t - x^*\|_2^2$$

Global Convergence for Newton's Method?

- Now we learnt the local convergence of Newton's method, what about the global convergence?
- Unclear for general functions.
- But for a special type of function, it has very fast global convergence rate as well.

Global Convergence for Newton's Method

- The *sandwich* functions: a function f such that :
- There exists a positive definite matrix A and a value $a \geq 1$ such that for every x:
$$A \preceq \nabla^2 f(x) \preceq aA$$
- Example: Quadratic function $f(x) = x^T Ax + \langle x, w \rangle + c$, then $a = 1$.
- The *sandwich* functions are functions like quadratic functions.

Global Convergence for Newton's Method

- For this function f, suppose we update at every iteration:

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

- We can repeat the same calculation

$$x_{t+1} - x^* = x_t - x^* - \eta[\nabla^2 f(x_t)]^{-1}\nabla f(x_t)$$
$$= x_t - x^* - \eta[\nabla^2 f(x_t)]^{-1}\int_0^1 \nabla^2 f(x^* + s(x_t - x^*))(x_t - x^*)ds$$

- Denote the matrix $B_t = [\nabla^2 f(x_t)]^{-1}\int_0^1 \nabla^2 f(x^* + s(x_t - x^*))ds$, then we have:

$$x_{t+1} - x^* = (I - \eta B_t)(x_t - x^*)$$

Global Convergence for Newton's Method

- Denote the matrix $B_t = [\nabla^2 f(x_t)]^{-1} \int_0^1 \nabla^2 f(x^* + s(x_t - x^*))ds$, then we have:
$$x_{t+1} - x^* = (I - \eta B_t)(x_t - x^*)$$
- Note that $A \preceq \nabla^2 f(x) \preceq aA$ for every x, so
$$\frac{I}{a^2} \preceq B_t B_t^\top \preceq a^2 I$$
- If we pick $\eta \leq \frac{1}{a}$, we have that $\|I - \eta B_t\|_{spectral\ \ norm} \leq 1 - \eta/a$, this implies:
$$\|x_{t+1} - x^*\|_2 \leq (1 - \eta/a)\|x_t - x^*\|_2$$

Global Convergence for Newton's Method

- The *sandwich* functions: a function f such that :
- There exists a positive definite matrix A and a value $a \geq 1$ such that for every x:

$$A \preceq \nabla^2 f(x) \preceq aA$$

- We have that for every $\eta \leq \frac{1}{a}$, the update

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

- Converges at a linear rate

$$\|x_{t+1} - x^*\|_2 \leq (1 - \eta/a)\|x_t - x^*\|_2$$

- This algorithm is also known as the pre-conditioned gradient descent.

Preconditioned Gradient Descent

- We have that for every $\eta \leq \frac{1}{a}$, update

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

- Actually, it doesn't matter if we use $[\nabla^2 f(x_t)]^{-1}$ or any matrix M such that

$$\frac{M}{a} \preceq \nabla^2 f(x) \preceq aM$$

- The update $x_{t+1} = x_t - \eta M^{-1}\nabla f(x_t)$ has the same convergence rate

$$\|x_{t+1} - x^*\|_2 \leq (1 - \eta/a)\|x_t - x^*\|_2$$

- M is called the pre-condition matrix, and $x_{t+1} = x_t - \eta M^{-1}\nabla f(x_t)$ is called the pre-conditioned gradient descent algorithm

Preconditioned Gradient Descent

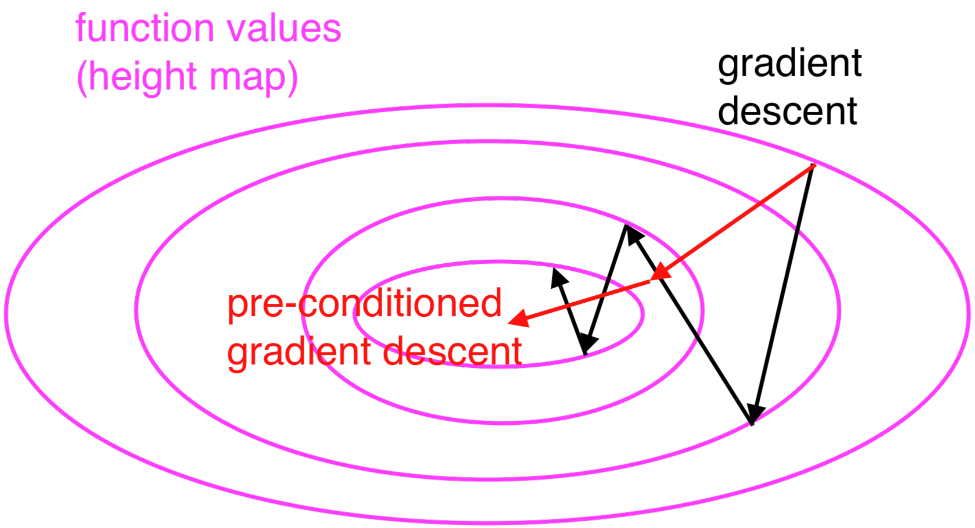
- Example: $x \in \mathbb{R}^d$, $f(x) = x^T \begin{pmatrix} 100 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \dots & & & & \\ 0 & 0 & \dots. & 1 & 0 \\ 0 & 0 & \dots. & 0 & 1 \end{pmatrix} x = x^T A x$.
- Gradient Descent $x_{t+1} = x_t - \eta(200[x_t]_1, 2[x_t]_2, 2[x_t]_3, \cdots, 2[x_t]_d)$: Can not use learning rate $\eta \geq \frac{1}{100}$ – Slow convergence.
- Pre-conditioned Gradient Descent using $M = A$:

$$x_{t+1} = x_t - 2\eta([x_t]_1, [x_t]_2, [x_t]_3, \cdots, [x_t]_d)$$

- Using any learning rate $\eta \leq 1/2$, we have a linear convergence rate $x_{t+1} = x_t(1 - 2\eta)$.

Preconditioned Gradient Descent

- The update $x_{t+1} = x_t - \eta M^{-1} \nabla f(x_t)$ changes the *geometry* to achieve a better convergence rate:



-
- The spirit of the update is extremely important, as we will see in adaptive algorithms how to design algorithms that finds this pre-condition M automatically to significantly improve the convergence rate.
- Adaptivity: Adapt to the *geometry* of $\nabla^2 f(x)$.