262B-Lecture 9

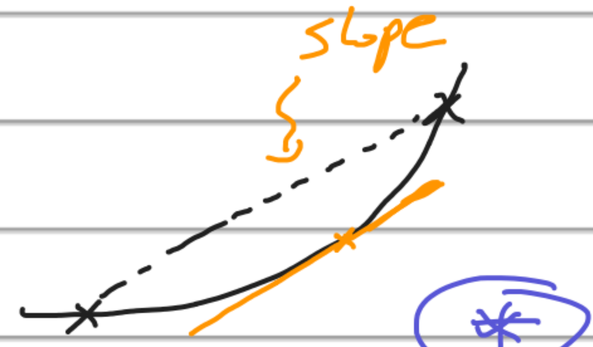
Date created: 2021.02.16
N. of Pages: 14

Newton's method:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \underbrace{\nabla^2 f(x^{(k)})^{-1}} \nabla f(x^{(k)})$$

need to have access to Hessian

$\Rightarrow$ Can we approximate it ?



estimation method:

$$\nabla^2 f(x^{(k+1)})(x^{(k+1)} - x^{(k)}) \simeq \nabla f(x^{(k+1)}) - \nabla f(x^{(k)})$$

(*)

$\downarrow$ new algorithm

$$f''(x) \sim \frac{f'(x+\varepsilon) - f'(x)}{(x+\varepsilon) - x}$$

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \underbrace{D^{(k)}} \nabla f(x^{(k)})$$

estimates inverse of Hessian

(*) $\Rightarrow$ $\underbrace{D^{(k+1)}}_{\text{matrix}} \underbrace{(\nabla f(x^{(k+1)}) - \nabla f(x^{(k)}))}_{\text{vector}} = \underbrace{x^{(k+1)} - x^{(k)}}_{\text{vector}}$

If $n = 1$ $\Rightarrow$ we can solve for $D^{(k+1)}$

If $n > 1$ $\Rightarrow$ n equations & $\frac{n(n+1)}{2}$ variables in $D^{(k+1)}$

So, $D^{(k+1)}$ can't be uniquely found.

what if I find an optimal $D^{(k+1)}$?

Fact : $D^{(k+1)}$ can't be too far away

from $D^{(k)}$ $\Rightarrow$ Smoothness in Hessian

$D^{(k+1)}$: optimal solution to

$$\min_{D} \| D - D^{(k)} \|_{Q} \Rightarrow \text{weighted 2-norm}$$

$$\text{s.t.} \quad D = D^T$$

$$D\left(\underbrace{\nabla f(x^{(k+1)}) - \nabla f(x^{(k)})}\right) = \underbrace{x^{(k+1)} - x^{(k)}}$$

$\Rightarrow$ This has a closed-form solution.

$\Rightarrow$ $D^{(k+1)}$ is a function of $D^{(k)}$.

Definition: $q^{(k)} = \nabla f(x^{(k+1)}) - \nabla f(x^{(k)})$

$$p^{(k)} = x^{(k+1)} - x^{(k)}$$

- Pick some arbitrary $D^{(0)} > 0$.

- $$D^{(k+1)} = D^{(k)} + \left(1 + \frac{(q^{(k)})^T D^{(k)} q^{(k)}}{(p^{(k)})^T q^{(k)}}\right) \frac{p^{(k)} (p^{(k)})^T}{(p^{(k)})^T p^{(k)}}$$

$$- \frac{D^{(k)} q^{(k)} (p^{(k)})^T + p^{(k)} (q^{(k)})^T D^{(k)}}{(p^{(k)})^T q^{(k)}}$$

BFGS rule : solution to previous optimization for some $\|\cdot\|$

Now : $x^{(k+1)} = x^{(k)} - \alpha^{(k)} D^{(k)} \nabla f(x^{(k)})$

where $D^{(k)}$ is obtained based on BFGS $\Longrightarrow$ quasi Newton

Since finding D as an estimate of $(\nabla^2 f)^{-1}$ wasn't unique, it is <u>not</u> the case that

$$\lim_{k \to \infty} D^{(k)} \longrightarrow \lim_{k \to \infty} \nabla^2 f(x^{(k)})^{-1}$$

Since we started with $x^{(k+1)} - x^{(k)}$ to estimate

Hessian, the method just worked through directional derivative.

$\Rightarrow \quad -\underbrace{D^{(k)}}_{\text{BFGS}} \nabla f(x^{(k)}) \longrightarrow$ Newton's direction

as $k \to \infty$

(independent of the choice of $D^{(0)}$)

Convergence analysis if $\nabla^2 f(x) \geq \underbrace{m}_{>0} I$

strongly convex case

gradient alg. $\longrightarrow$ linear convergence

- quasi-newton alg. $\longrightarrow$ superlinear $\lim_{k \to \infty} \frac{e^{(k+1)}}{e^{(k)}} = 0$

newton alg. $\longrightarrow$ quadratic convergence

Conjugate gradient method :

so far, iteration complexity

↓

$O(\log\log\frac{1}{\varepsilon})$, $O(\log\frac{1}{\varepsilon})$, $O(\frac{1}{\sqrt{\varepsilon}})$, $O(\frac{1}{\varepsilon})$,

$O(\frac{1}{\varepsilon^2})$

This is about the number of iterations.

what is the role of n ?

Computational complexity = end-to-end complexity

~ number of iterations × complexity per iteration

e.g. $O(\log\frac{1}{\varepsilon})$?

Gradient alg. : $\Delta x^{(k)} = -\nabla f(x^{(k)})$

Complexity = $O(n)$ if complexity of evaluating each partial derivative is $O(1)$.

Newton's method :

$$\Delta x^{(k)} = -\nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)}) \quad (*)$$

Complexity = $O(n^3)$ if you use the naive method of taking the inverse & doing the multiplication.

$\Rightarrow$ Can we improve this ?

$(*) \Rightarrow \underbrace{\nabla^2 f(x^{(k)})}_{A} \underbrace{\Delta x^{(k)}}_{x} = \underbrace{-\nabla f(x^{(k)})}_{b}$

Find x satisfying $Ax = b$

New method to solve equations.

Newton's method : $A > 0$

General case : $A \not> 0$

$Ax = b \Longrightarrow \underbrace{A^T A}_{\text{new } A \geq 0} x = \underbrace{A^T b}_{\text{new } b}$

Let's focus on solving $\underbrace{Q}_{>0} x_* = b$

This corresponds to solving

$$\min_x \frac{1}{2} x^T Q x - b^T x$$

Definition: A set of nonzero vectors $d^{(1)}, d^{(2)}, \ldots, d^{(k)}$ are called Q-conjugate if $(d^{(i)})^T Q (d^{(j)}) = 0 \quad \forall\, i, j \in \{1, \ldots, k\}, i \neq j$

Thm: Q-conjugacy implies $d^{(1)}, \ldots, d^{(k)}$ are linearly independent.

To prove by contradiction:

$$d^{(k)} = \underbrace{\alpha_1}_{\in \mathbb{R}} d^{(1)} + \underbrace{\alpha_2}_{\in \mathbb{R}} d^{(2)} + \cdots + \underbrace{\alpha_{k-1}}_{\in \mathbb{R}} d^{(k-1)}$$

$$\Rightarrow \underbrace{(d^{(k)})^T Q d^{(k)}}_{= 0} = \sum_{i=1}^{k-1} \alpha_i \underbrace{(d^{(k)})^T Q d^{(i)}}_{0}$$

$= 0 \Rightarrow$ since $Q > 0$: $d^{(k)} = 0$ $\times$

Devise a new algorithm:

- Consider $\underline{n}$ (Q)-conjugate vectors

$d^{(0)}, d^{(1)}, \ldots, d^{(n-1)}$ $\Rightarrow$ treat them as directions

- $x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)}$, $k = 0, 1, \ldots$

where $x^{(0)}$ is arbitrary and

$$\alpha^{(k)} \longrightarrow \min_{\alpha} f(x^{(k)} + \alpha d^{(k)})$$

positive or negative

(*)

Focus on the quadratic case:

$$f(x) = \frac{1}{2} x^T Q x - b^T x$$

(*) $$0 = \frac{\partial f(x^{(k)} + \alpha d^{(k)})}{\partial \alpha} \Big|_{\alpha = \alpha^{(k)}}$$

$$= (d^{(k)})^T \nabla f(x^{(k)} + \alpha^{(k)} d^{(k)})$$

$$\Rightarrow \quad 0 = (d^{(k)})^T (Q(x^{(k)} + \alpha^{(k)} d^{(k)}) - b)$$

$$\Rightarrow \quad \alpha^{(k)} = \frac{(d^{(k)})^T (b - Q x^{(k)})}{(d^{(k)})^T Q d^{(k)}}$$

> 0 due to Q-conjugacy

Define: $M^{(k)} = \{ x \mid x = x^{(0)} + v \quad \text{where}$

$v \in$ subspace spanned by $d^{(0)}, \ldots, d^{(k)} \}$

Make a space $\longrightarrow$ shift it by $x^{(0)}$

by $d^{(0)}, \ldots, d^{(k)}$

$\Rightarrow \dim M^{(k)} = k+1 \quad \& \quad \boxed{M^{(n-1)} = \mathbb{R}^n}$

$$\underbrace{M^{(0)}}_{1\text{-}d} \subset \underbrace{M^{(1)}}_{2\text{-}d} \subset \underbrace{M^{(2)}}_{3\text{-}d} \cdots \subset M^{(n-1)}$$

Thm: $x^{(k+1)}$ is solution to

$$\min_{x \in M^{(k)}} f(x) \qquad \forall k$$

By product: $K = n-1$

$\Rightarrow$ $x^{(n)}$ is solution to $\min_{x \in M^{(n-1)}} f(x)$

$= \min_{x \in \mathbb{R}^n} f(x)$

$\Rightarrow$ This algorithm ($x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)}$)

stops after at most $\underline{n}$ steps.

Proof: $\left. \frac{\partial f(x^{(i)} + \alpha d^{(i)})}{\partial \alpha} \right|_{\alpha = \alpha^{(i)}} = 0$

$\forall\, i$

$\Rightarrow$ $\boxed{(d^{(i)})^T \nabla f(x^{(i+1)}) = 0 \qquad \forall\, i}$ (*)

pick $i \in \{0, \dots, k-1\}$:

$(d^{(i)})^T \nabla f(x^{(k+1)}) = (d^{(i)})^T (Q x^{(k+1)} - b)$

$$x^{(k+1)} \longrightarrow \underbrace{x^{(k)} + \alpha^{(k)} d^{(k)}}_{x^{(k-1)} + \alpha^{(k-1)} d^{(k-1)}}$$

$$\vdots$$

$$\underbrace{\qquad\qquad}_{x^{(i+1)} + \alpha^{(i+1)} d^{(i+1)}}$$

$$\Rightarrow (d^{(i)})^T \nabla f(x^{(k+1)}) = (d^{(i)})^T (Q x^{(i+1)} - b) + \sum_{j=i+1}^{k} (d^{(i)})^T Q d^{(j)} \alpha^{(j)}$$

$$\nabla f(x^{(i+1)}) \qquad\qquad 0$$

$$= (d^{(i)})^T \nabla f(x^{(i+1)}) = 0$$

due to (*)

$$\Rightarrow \boxed{(d^{(i)})^T \nabla f(x^{(k+1)}) = 0 \qquad i = 0, 1, \ldots, k} \quad (**)$$

note: $x^{(k+1)} = x^{(0)} + \alpha^{(0)} d^{(0)} + \alpha^{(1)} d^{(1)} + \cdots + \alpha^{(k)} d^{(k)}$

$$\Rightarrow \frac{\partial f(x^{(0)} + \beta_0 d^{(0)} + \beta_1 d^{(1)} + \cdots + \beta_k d^{(k)})}{\partial \beta_i} \Bigg|_{\beta_i = \alpha^{(i)}} = 0$$

due to (**) $\leftarrow$ $i = 0, \ldots, k$

$x^{(k+1)}$ is a solution to

$$\min_{\beta_0,\dots,\beta_k} f(x^{(0)} + \beta_0 d^{(0)} + \cdots + \beta_k d^{(k)})$$

$$= \min_{x \in M^{(k)}} f(x)$$

Conclusion: at every iteration the search space expands by dim ≥ 1.

Generate Q-Conjugate directions:

Given linearly independent vectors $v^{(0)}, \dots, v^{(n-1)}$ we can generate Q-Conjugate vectors $d^{(0)}, \dots, d^{(n-1)}$ such that

subspace spanned by $d^{(0)}, \dots, d^{(i)}$
$=$ subspace spanned by $v^{(0)}, \dots, v^{(i)}$ for $i = 0, \dots, n-1$

$i = 0 \Longrightarrow$ pick $d^{(0)} = v^{(0)}$

Have a constructive proof by induction.

Assume we have generated $d^{(0)}, \ldots, d^{(i)}$

$\longrightarrow$ How to pick $d^{(i+1)}$?

By induction step :

span of $d^{(0)}, \ldots, d^{(i)}$ = Span of $v^{(0)}, \ldots, v^{(i)}$

aim to

span of $d^{(0)}, \ldots, d^{(i+1)}$ = span of $v^{(0)}, \ldots, v^{(i+1)}$

To achieve this, we write :

$$d^{(i+1)} = v^{(i+1)} + \beta_0 d^{(0)} + \beta_1 d^{(1)} + \cdots + \beta_i d^{(i)}$$

For $j = 0, \ldots, i$ $\quad$ $= 0$ to get Q-Conjugacy

$$\Rightarrow (d^{(j)})^T Q d^{(i+1)} = (d^{(j)})^T Q v^{(i+1)} +$$

$$\beta_0 (d^{(j)})^T Q d^{(0)} + \cdots + \beta_i (d^{(j)})^T Q d^{(i)}$$

$$= (d^{(j)})^T Q v^{(i+1)} + \beta_j (d^{(j)})^T Q d^{(j)}$$

$$\Rightarrow \beta_j = - \frac{(d^{(j)})^T Q v^{(i+1)}}{(d^{(j)})^T Q d^{(j)}}$$

Now, $d^{(i+1)} = v^{(i+1)} + \beta_0 d^{(0)} + \cdots + \beta_i d^{(i)}$

with above coefficients will work.

Conjugate gradient method:

$$-\nabla f(x^{(0)}), -\nabla f(x^{(1)}), \ldots, -\nabla f(x^{(n-1)})$$

$$\Downarrow$$

generate Q-Conjugate directions

$$\Rightarrow x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)}$$