2023.03.16

Unconstrained minimization

임승현, 김종민

9.1 Unconstrained minimization problems

Unconstrained minimization problem

$$\text{minimize} \quad f(x)$$

- f convex, twice continuously differentiable (hence $\mathbf{dom}\, f$ open)
- we assume optimal value $p^\star = \inf_x f(x)$ is attained (and finite)

■ Necessary and sufficient condition

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

■ Unconstrained minimization methods

- produce sequence of points $x^{(k)} \in \mathbf{dom}\, f$, $k = 0, 1, \ldots$ with $f(x^{(k)}) \to p^\star$

Initial point and sublevel set

- Initial point (starting point)

  - $x^{(0)} \in \mathbf{dom}\, f$
  - sublevel set $S = \{x \mid f(x) \leq f(x^{(0)})\}$ is closed

  2nd condition is hard to verify, except when *all* sublevel sets are closed:

  - equivalent to condition that $\mathbf{epi}\, f$ is closed
  - true if $\mathbf{dom}\, f = \mathbf{R}^n$
  - true if $f(x) \to \infty$ as $x \to \mathbf{bd}\, \mathbf{dom}\, f$

Examples

$$f(x) = \log(\sum_{i=1}^{m} \exp(a_i^T x + b_i))$$

■ Optimaltiy condition

$$\nabla f(x^\star) = \frac{1}{\sum_{j=1}^{m} \exp(a_j^T x^\star + b_j)} \sum_{i=1}^{m} \exp(a_i^T x^\star + b_i) a_i = 0$$

■ Domain

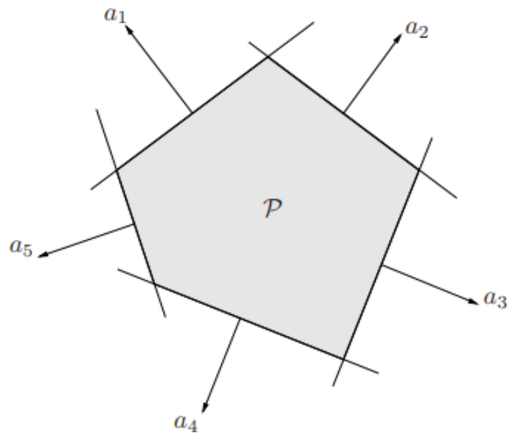
$$\mathbf{dom}\, f = \mathbf{R}^n$$

→ Any point can be chosen as the initial point!

$$f(x) = -\sum_{i=1}^{m} \log(b_i - a_i^T x)$$

■ Domain

$$\mathbf{dom}\, f = \{x \mid a_i^T x < b_i,\ i = 1, \ldots, m\}$$



→ The initial point must satisfy the strict inequalities $a_i^T x^{(0)} < b_i$.

Strong convexity and implications

- f is strongly convex on S if there exists an $m > 0$ such that

$$\nabla^2 f(x) \succeq mI \qquad \text{for all } x \in S$$

■ Implications

- For x, y on S

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

$$\begin{aligned} f(y) &\geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2}\|y - x\|_2^2 \\ &\geq f(x) + \nabla f(x)^T (\tilde{y} - x) + \frac{m}{2}\|\tilde{y} - x\|_2^2 \\ &= f(x) - \frac{1}{2m}\|\nabla f(x)\|_2^2. \end{aligned}$$

$f(x) - p^\star \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$ 을 stopping condition 으로 사용할 수 있다!

Strong convexity and implications

■ Upper bound on $\nabla^2 f(x)$

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2}\|y - x\|_2^2$$

→ S is bounded. Therefore, the $\nabla^2 f(x)$ is bounded above on S

$$\nabla^2 f(x) \preceq MI$$

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{M}{2}\|y - x\|_2^2$$

$$p^\star \leq f(x) - \frac{1}{2M}\|\nabla f(x)\|_2^2$$

■ $mI \preceq \nabla^2 f(x) \preceq MI$

- f는 strong convexity 조건을 만족한다고 가정했으므로, positive definite.
- f의 Hessian matrix 는 가장 큰 eigenvalue M과 가장 작은 eigenvalue m을 가진다.

Condition number

■ Condition number of the matrix $\nabla^2 f(x)$

$$mI \preceq \nabla^2 f(x) \preceq MI$$

- The ratio $\frac{M}{m}$ is an upper bound on the condition number of the matrix $\nabla^2 f(x)$ (or the sublevel sets of f)

  → The ratio of its largest eigenvalue to its smallest eigenvalue

→ Condition number는 Hessian matrix의 가장 크고 작은 eigenvalue비율이고 descent method의 수렴 속도와 관련이 있다!

■ Geometric interpretation

- The width of a convex set C in the direction $q, where\ \|q\| = 1$

$$W(C,q) = \sup_{z \in C} q^T z - \inf_{z \in C} q^T z$$

$$W_{\min} = \inf_{\|q\|_2 = 1} W(C,q), \qquad W_{\max} = \sup_{\|q\|_2 = 1} W(C,q)$$

$$\mathbf{cond}(C) = \frac{W_{\max}^2}{W_{\min}^2}$$

→ The condition number of C gives a measure of its anisotropy (condition number가 작을수록 원에 가까운 형태를 가진다.)

9.2 Descent methods

Descent methods

$$x^{(k+1)} = x^{(k)} + t^{(k)}\Delta x^{(k)} \quad \text{with } f(x^{(k+1)}) < f(x^{(k)})$$

- other notations: $x^+ = x + t\Delta x$, $x := x + t\Delta x$
- Δx is the *step*, or *search direction*; t is the *step size*, or *step length*
- from convexity, $f(x^+) < f(x)$ implies $\nabla f(x)^T \Delta x < 0$
  ($i.e.$, Δx is a *descent direction*)

■ General descent method

given a starting point $x \in \mathbf{dom}\, f$.
repeat
1. Determine a descent direction Δx.
2. *Line search*. Choose a step size $t > 0$.
3. *Update*. $x := x + t\Delta x$.

until stopping criterion is satisfied. $\|\nabla f(x)\|_2 \leq \eta$

Exact line Search

■ Line search

- Descent method에서 step size t를 결정하는 방법

  → step size가 optimal point로 수렴하지 않을 수도 있고, 반대로 너무 작으면 optimal point에 수렴하는 속도가 느려짐 → 적절한 step size를 찾는 방법

■ Exact line search

$$t = \text{argmin}_{t>0} f(x + t\Delta x)$$

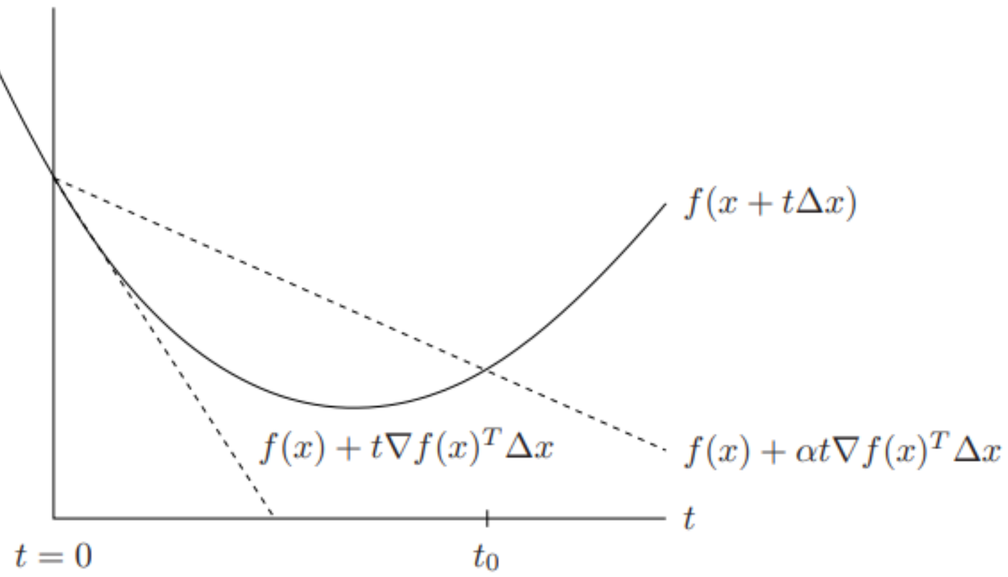
- Δx 상에서 f 를 최소화하는 t 를 찾는 문제!
- 각 스텝마다 argmin 값을 구해야 하기 때문에 효율적이지 못하다.

Backtracking line search

■ Backtracking line search

given a descent direction Δx for f at $x \in \mathbf{dom}\, f$, $\alpha \in (0, 0.5)$, $\beta \in (0, 1)$.
$t := 1$.
while $f(x + t\Delta x) > f(x) + \alpha t \nabla f(x)^T \Delta x$, $\quad t := \beta t$.

→ 하나의 step을 가보고 ($t = 1$), 해당 step에서 너무 많이 이동했으면, step size를 줄여서 ($t := \beta t$) 다시 이동하는 과정을 반복하여 t를 찾는 방법



9.3 Gradient descent methods

Gradient descent method

■ Gradient descent method

given a starting point $x \in \mathbf{dom}\, f$.
repeat
1. $\Delta x := -\nabla f(x)$.
2. *Line search.* Choose step size t via exact or backtracking line search.
3. *Update.* $x := x + t\Delta x$.

until stopping criterion is satisfied.

- stopping criterion usually of the form $\|\nabla f(x)\|_2 \leq \epsilon$

→ 현재의 point x에서 가장 빨리 증가하는 방향인 $\nabla f(x)$의
반대 방향으로 search direction으로 정하는 방법

Convergence analysis for exact line search

■ Recall) ① $mI \preceq \nabla^2 f(x) \preceq MI$ ② $f(y) \leq f(x) + \nabla f(x)^T (y-x) + \frac{M}{2}\|y-x\|_2^2$

③ $f(x) - p^\star \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$ ④ $p^\star \leq f(x) - \frac{1}{2M}\|\nabla f(x)\|_2^2$

■ Analysis for exact line search

Using the notation $x^+ = x + t\Delta x$ for $x^{(k+1)} = x^{(k)} + t^{(k)}\Delta x^{(k)}$ where $\Delta x = -\nabla f(x)$

Define the function $\tilde{f}(t) = f\big(x - t\nabla f(x)\big)$ Recall) exact line search $t = \operatorname{argmin}_{t>0} f(x + t\Delta x)$

1) ②식에서 y 대신 $x - t\nabla f(x)$ 대입

$$\tilde{f}(t) \leq f(x) - t\|\nabla f(x)\|_2^2 + \frac{Mt^2}{2}\|\nabla f(x)\|_2^2$$

Convergence analysis for exact line search

2) (1)에서의 식의 우변은 $t = \frac{1}{M}$ 에서 최소, t_{exact}는 $\tilde{f}(x)$를 최소화하는 step length

$$f(x^+) = \tilde{f}(t_{\text{exact}}) \leq f(x) - \frac{1}{2M}\|\nabla(f(x))\|_2^2$$

→ exact line search를 통해 가장 최적의 t를 대입. 즉 새로운 x^+를 가장 잘 update해도 위의 식은 성립.

3) (2)의 식의 양변에서 optimal value p^*를 빼주고, 식 ③을 이용해 정리.

$$f(x^+) - p^\star \leq f(x) - p^\star - \frac{1}{2M}\|\nabla f(x)\|_2^2$$

$$f(x^+) - p^\star \leq (1 - m/M)(f(x) - p^\star)$$

$$f(x^{(k)}) - p^\star \leq c^k(f(x^{(0)}) - p^\star)$$

where $c = 1 - m/M$ → c는 1보다 작으므로 결국 수렴, m/M 값은 condition number와 관련

Condition number

■ Condition number of the matrix $\nabla^2 f(x)$

$$mI \preceq \nabla^2 f(x) \preceq MI$$

- The ratio $\frac{M}{m}$ is an upper bound on the condition number of the matrix $\nabla^2 f(x)$ (or the sublevel sets of f)

  → The ratio of its largest eigenvalue to its smallest eigenvalue

→ Condition number는 Hessian matrix의 가장 크고 작은 eigenvalue비율이고 descent method의 수렴 속도와 관련이 있다!

■ Geometric interpretation

- The width of a convex set C in the direction $q, where\ \|q\| = 1$

$$W(C,q) = \sup_{z \in C} q^T z - \inf_{z \in C} q^T z$$

$$W_{\min} = \inf_{\|q\|_2 = 1} W(C,q), \qquad W_{\max} = \sup_{\|q\|_2 = 1} W(C,q)$$

$$\mathbf{cond}(C) = \frac{W_{\max}^2}{W_{\min}^2}$$

→ The condition number of C gives a measure of its anisotropy (condition number가 작을수록 원에 가까운 형태를 가진다.)

Examples

■ Quadratic problem in R^2

$$f(x) = \frac{1}{2}(x_1^2 + \gamma x_2^2)$$

- Optimal point는 (0, 0), optimal value는 0
- Sublevel sets f의 condition number은 $\dfrac{\max\{1,\gamma\}}{\min\{1,\gamma\}} = \max\{\gamma, 1/\gamma\}$
- Starting point $x^{(0)}$는 exact line search 방법에 의해 $(\gamma, 1)$ 로 주어짐.

유도 과정: 연습문제 9.6

$$f(x^{(k)}) = \frac{\gamma(\gamma+1)}{2}\left(\frac{\gamma-1}{\gamma+1}\right)^{2k} = \left(\frac{\gamma-1}{\gamma+1}\right)^{2k} f(x^{(0)})$$

1) $\gamma = 1$, (condition number) = 1
→ exact solution이 한 번에 구해짐.

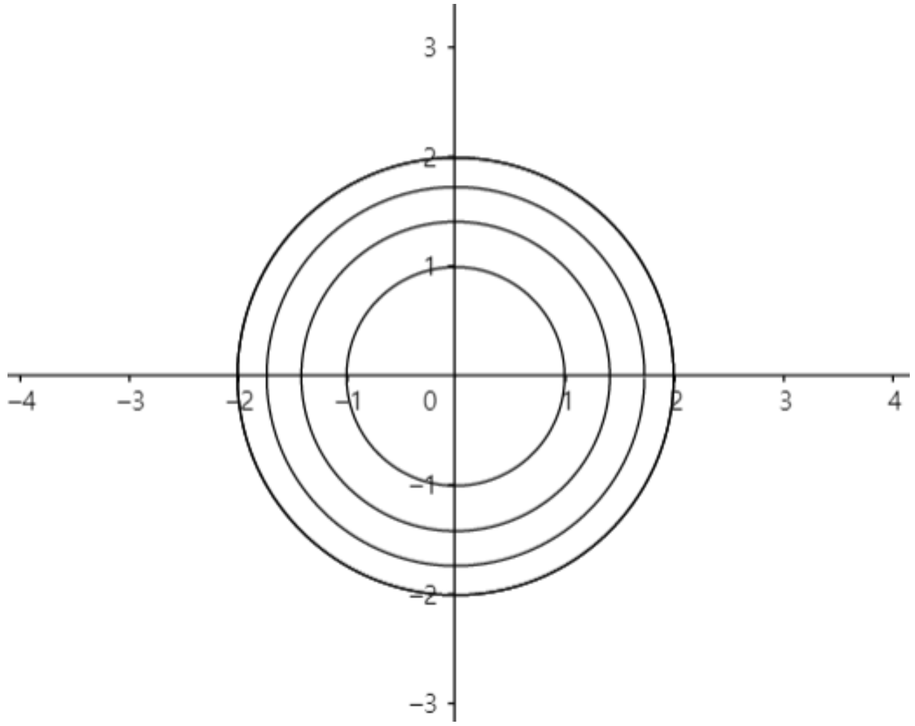
2) $\gamma \gg 1$ or $\gamma \ll 1$, condition number는 커짐.
→ convergence 속도는 매우 느려짐.

→ Condition number가 작을수록 optimal point로 빨리 수렴한다.

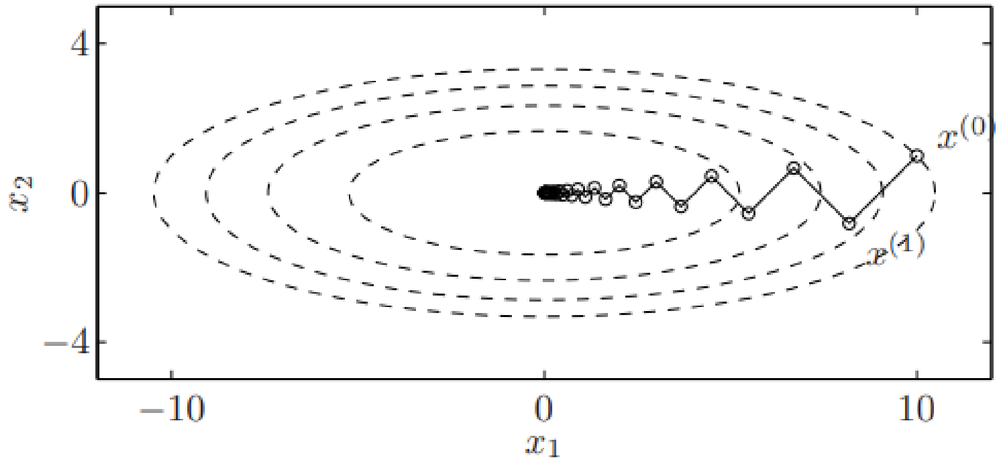
Examples

■ Quadratic problem in R^2

$$f(x) = \frac{1}{2}(x_1^2 + \gamma x_2^2)$$



1) $\gamma = 1$, (condition number) = 1
→ exact solution이 한 번에 구해짐.

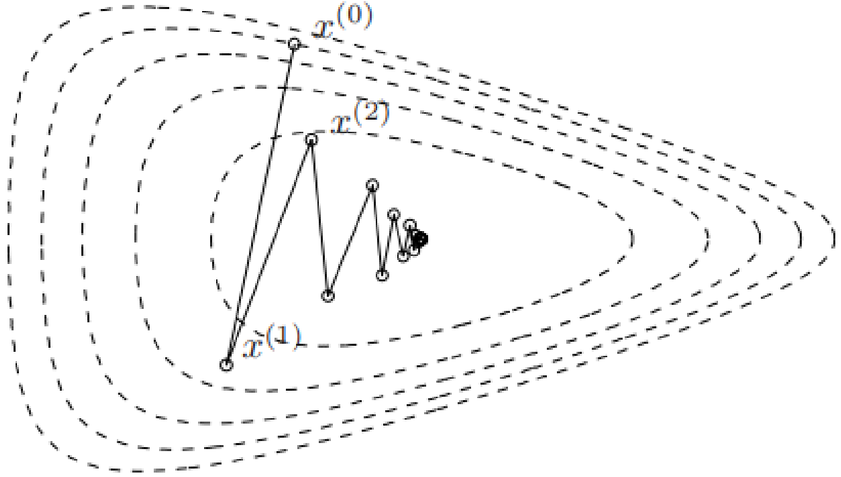


2) $\gamma = 10$, (condition number) = 10
→ convergence 속도는 매우 느려짐.

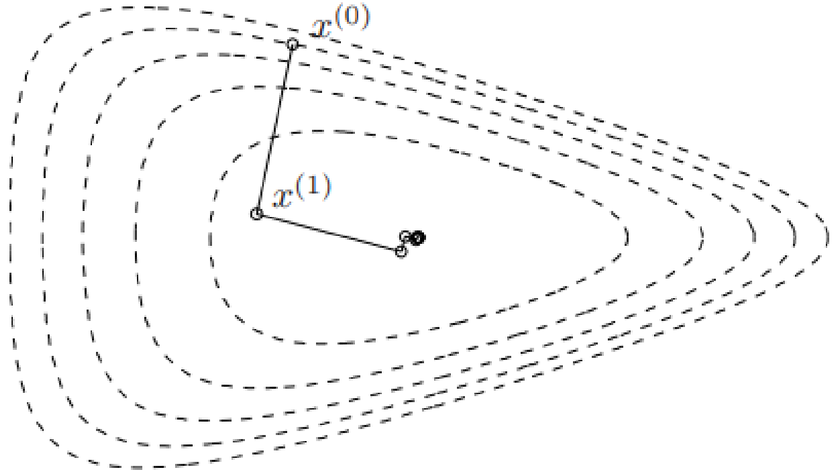
Examples

- Nonquadratic problem in R^2

$$f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}$$



Backtracking line search with $\alpha = 0.1, \beta = 0.7$



Exact line search

9.4 Steepest descent method

Steepest descent method

■ Steepest descent method

$$f(x+v) \approx \hat{f}(x+v) = f(x) + \nabla f(x)^T v$$

- 아주 작은 step v (descent direction if directional derivative is negative)에 대해서 위의 식이 성립
→ v의 크기를 적절히 제한하고 방향을 결정해야 함.

■ Normalized steepest descent direction

$$\Delta x_{\text{nsd}} = \text{argmin}\{\nabla f(x)^T v \mid \|v\| = 1\}$$

- v는 $f(x)$ 의 1차 근사식을 가장 빨리 감소시키는 방향.
즉, $\nabla f(x)^T v$ 를 minimize 하는 v를 찾아 search direction 으로 정함

- 여기서 $\| v \| = 1$ 로 제한을 주어 $argmin$ 값이 $-\infty$ 가 되지 않도록 하며,
기준이 되는 norm을 정해주어야 함. (ex) Euclidean norm, quadratic norm)

Steepest descent direction and Dual norm

■ (Unnormalized) Steepest descent direction

$$\Delta x_{\text{sd}} = \|\nabla f(x)\|_* \Delta x_{\text{nsd}}$$

Recall) Normalized steepest descent direction

$$\Delta x_{\text{nsd}} = \text{argmin}\{\nabla f(x)^T v \mid \|v\| = 1\}$$

■ Dual norm (A.1.6)

Let $\|\cdot\|$ be a norm on $\mathbf{R}^n$. The associated *dual norm*, denoted $\|\cdot\|_*$, is defined as

$$\|z\|_* = \sup\{z^T x \mid \|x\| \leq 1\}.$$

Norm	Dual norm
l_1	l_∞
l_∞	l_1
l_2	l_2

■ Steepest descent step ($\nabla f(x)^T v$)

$$\nabla f(x)^T \Delta x_{\text{sd}} = \|\nabla f(x)\|_* \nabla f(x)^T \Delta x_{\text{nsd}} = -\|\nabla f(x)\|_*^2$$

Since $\nabla f(x)^T \Delta x_{\text{nsd}} = -\|\nabla f(x)\|$ by definition of dual norm.

Steepest descent for Euclidean, quadratic norm

■ Steepest descent for Euclidean norm

$$\Delta x_{\text{sd}} = -\nabla f(x)$$

→ 앞에서 배웠던 gradient descent method와 동일

Recall) Steepest descent direction

$$\Delta x_{\text{sd}} = \|\nabla f(x)\|_* \Delta x_{\text{nsd}}$$

■ Steepest descent for quadratic norm

$$\|z\|_P = (z^T P z)^{1/2} = \|P^{1/2} z\|_2 \qquad \text{where } P \in \mathbf{S}^n_{++}$$

$$\Delta x_{nsd} = argmin\,\{\nabla f(x)^T v \mid \|v\|p = (v^T p v)^{1/2} = \left\|P^{1/2} v\right\|_2 = 1$$

$$\nabla f(x) + \lambda(2Pv) = 0$$
$$\rightarrow v \propto -P^{-1}\nabla f(x)$$

→ by KKT condition

$$\Delta x_{\text{sd}} = -P^{-1}\nabla f(x)$$

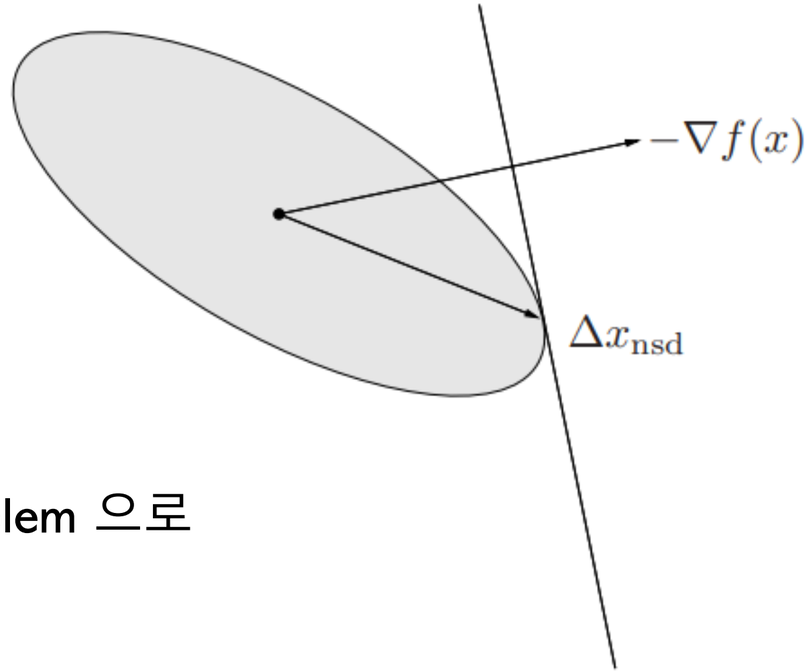
$$\Delta x_{\text{nsd}} = -\left(\nabla f(x)^T P^{-1} \nabla f(x)\right)^{-1/2} P^{-1}\nabla f(x)$$

→ by $\|z\|_* = \|P^{-1/2} z\|_2$

Steepest descent for quadratic norm

■ Normalized steepest descent direction

$$\Delta x_{\text{nsd}} = \text{argmin}\{\nabla f(x)^T v \mid \|v\| = 1\}$$



■ Interpretation via change of coordinates

Define $\bar{u} = P^{1/2}u$, then we have $\|u\|_P = \|\bar{u}\|_2$

$\bar{f}(\bar{u}) = f(P^{-1/2}\bar{u}) = f(u)$ → 좌표변환을 통해 equivalent problem 으로

Gradient method를 $\bar{f}$에 적용

$$\Delta\bar{x} = -\nabla\bar{f}(\bar{x}) = -P^{-1/2}\nabla f(P^{-1/2}\bar{x}) = -P^{-1/2}\nabla f(x)$$

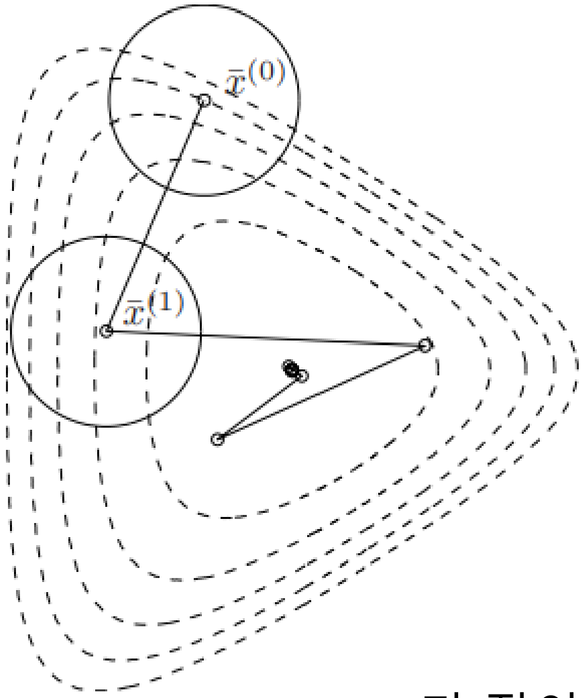
$$\Delta x = P^{-1/2}\left(-P^{-1/2}\nabla f(x)\right) = -P^{-1}\nabla f(x) \quad \rightarrow \quad \Delta x_{\text{sd}} = -P^{-1}\nabla f(x) \text{ 와 동일한 결과}$$

→ quadratic norm $\|\cdot\|_P$ 에서의 steepest descent method 의 결과는, $\bar{x} = P^{1/2}x$ 을 통해 좌표변환을 한 후 gradient method를 적용한 결과와 같다.

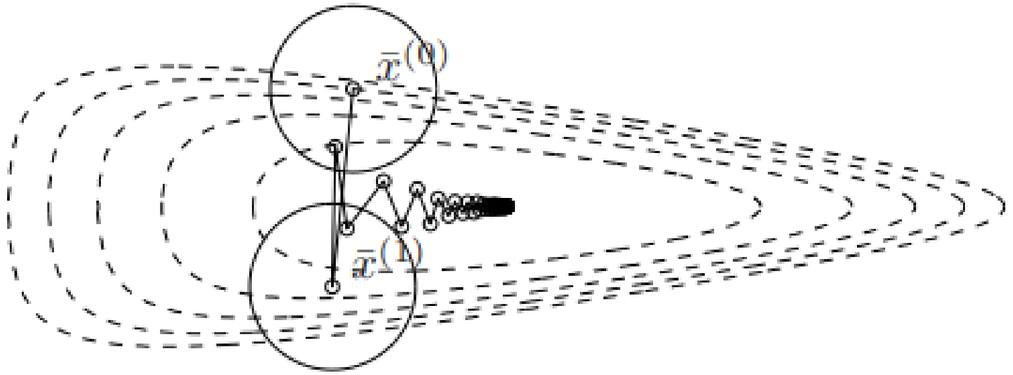
Steepest descent for quadratic norm

● Quadratic norm $\|\cdot\|_P$ 에서의 steepest descent method 의 결과는, $\bar{x} = P^{1/2}x$ 을 통해 좌표변환을 한 후 gradient method를 적용한 결과와 같다.

→ Quadratic norm을 정의하는 행렬 P 를 잘 선택하면, condition number 을 줄여 convergence 속도를 빠르게 할 수 있다.



→ condition number가 작아져 convergence 속도 빨라짐



→ condition number가 커져서 convergence 속도 느려짐

Steepest descent for l_1 norm

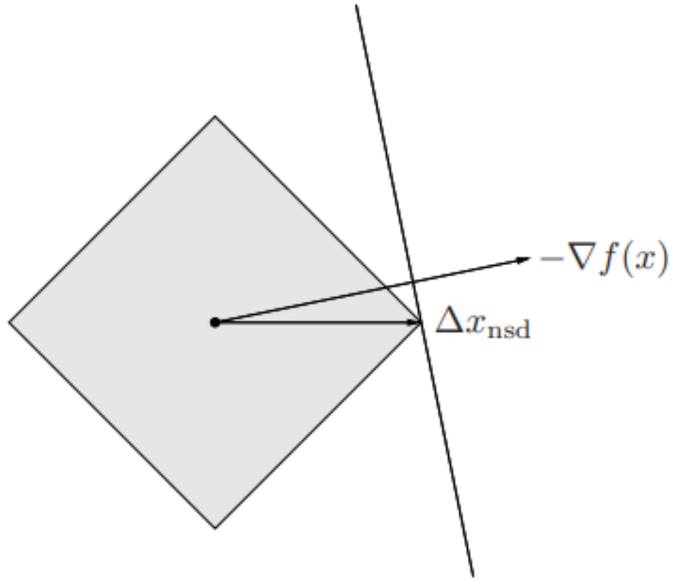
■ Normalized steepest descent direction

$$\Delta x_{\text{nsd}} = \text{argmin}\{\nabla f(x)^T v \mid \|v\|_1 \leq 1\}$$

$$\Delta x_{\text{nsd}} = -\text{sign}\left(\frac{\partial f(x)}{\partial x_i}\right) e_i$$

$$\Delta x_{\text{sd}} = \Delta x_{\text{nsd}} \|\nabla f(x)\|_\infty = -\frac{\partial f(x)}{\partial x_i} e_i$$

→ l_1의 dual norm은 l_∞ norm



Recall) l_∞ norm : $|\mathbf{x}|_\infty = \max_i |x_i|$

→ $\nabla f(x)$ 의 성분 중에서, 절댓값의 크기가 가장 큰 basis vector 방향으로 direction을 정함

9.5 Newton's method

Newton step

$$\Delta x_{\mathrm{nt}} = -\nabla^2 f(x)^{-1} \nabla f(x)$$ is called the Newton step

- Positive definiteness of hessian implies that

$$\nabla f(x)^T \Delta x_{\mathrm{nt}} = -\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x) < 0$$

unless $\nabla f(x) = 0$, so the Newton step is a descent direction.

- Update equation

  - Gradient Descent

$$x^{(k)} = x^{(k-1)} - t_k \cdot \nabla f(x^{(k-1)}), \quad k = 1, 2, 3, \ldots$$

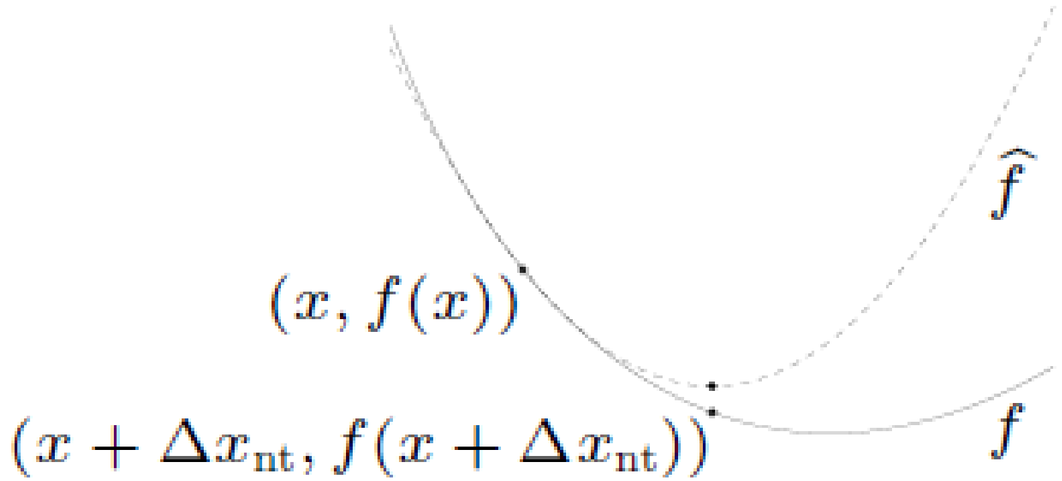
  - Newton's method

$$x^{(k)} = x^{(k-1)} - \left(\nabla^2 f(x^{(k-1)})\right)^{-1} \nabla f(x^{(k-1)}), \quad k = 1, 2, 3, \ldots$$

Newton step – Interpretations

- 1. minimizer of second-order approximation
  - $x + \Delta x_{\mathrm{nt}}$ minimizes second order approximation

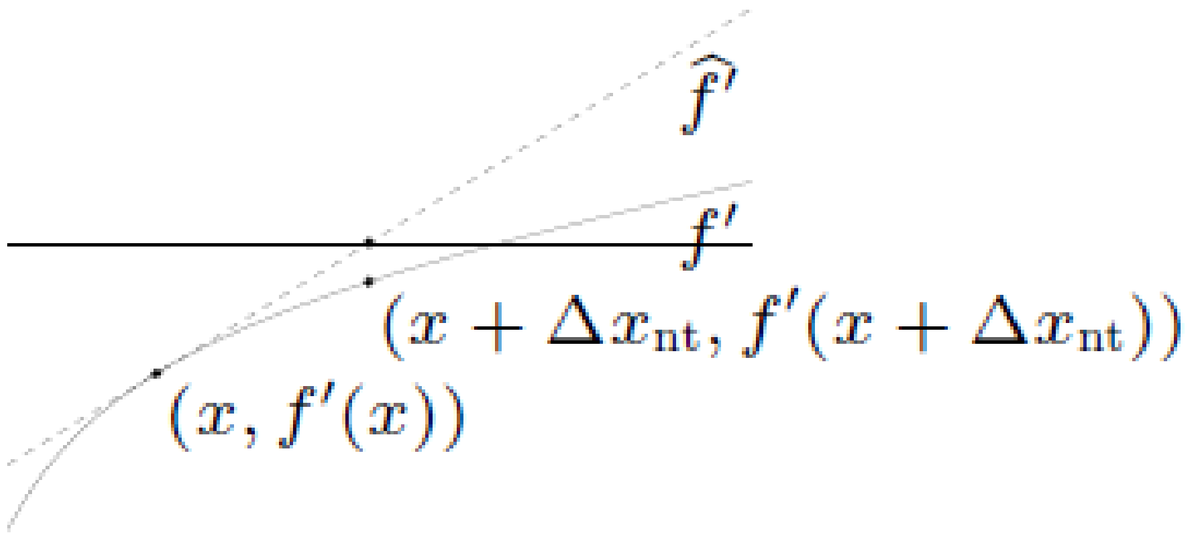
$$\widehat{f}(x+v) = f(x) + \nabla f(x)^T v + \frac{1}{2} v^T \nabla^2 f(x) v$$



Newton step – Interpretations

- 2. Solution of linearized optimality condition
  - $x + \Delta x_{\mathrm{nt}}$ solves linearized optimality condition

$$\nabla f(x+v) \approx \nabla \widehat{f}(x+v) = \nabla f(x) + \nabla^2 f(x) v = 0$$



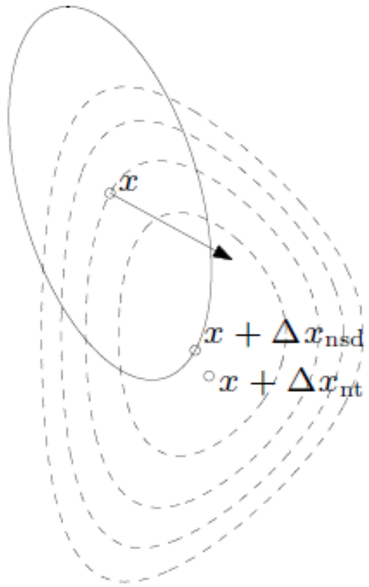
Newton step – Interpretations

■ 3. Steepest descent direction in Hessian norm

- The Newton step is also the steepest descent direction at x, for the quadratic norm defined by the Hessian

- $\triangle x_{nt}$ is steepest descent direction at x in local Hessian norm

$$\|u\|_{\nabla^2 f(x)} = \left(u^T \nabla^2 f(x) u\right)^{1/2}$$



dashed lines are contour lines of f; ellipse is $\{x + v \mid v^T \nabla^2 f(x) v = 1\}$

arrow shows $-\nabla f(x)$

Newton step – Affine invariance

- Suppose $T \in \boldsymbol{R^{m\times n}}$ *is nonsingular, and define* $\bar{f}(y) = f(Ty)$. *then*

$$\nabla \bar{f}(y) = T^T \nabla f(x), \qquad \nabla^2 \bar{f}(y) = T^T \nabla^2 f(x) T,$$

where $x = Ty$. The Newton step for $\bar{f}$ at y is therefore

$$\begin{aligned} \Delta y_{\mathrm{nt}} &= -\left(T^T \nabla^2 f(x) T\right)^{-1} \left(T^T \nabla f(x)\right) \\ &= -T^{-1} \nabla^2 f(x)^{-1} \nabla f(x) \\ &= T^{-1} \Delta x_{\mathrm{nt}}, \end{aligned}$$

where Δx_{nt} is the Newton step for f at x. Hence the Newton steps of f and $\bar{f}$ are related by the same linear transformation, and

$$x + \Delta x_{\mathrm{nt}} = T(y + \Delta y_{\mathrm{nt}}).$$

Newton step – Newton decrement

- **a measure of the proximity of x to x***
(현재 위치한 x가 optimal point인 x*와 얼마나 가까운지 나타내는 지표)

$$\lambda(x) = \left(\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x)\right)^{1/2}$$

■ Why? (properties)

- gives an estimate of $f(x) - p^\star$, using quadratic approximation $\hat{f}$:

$$f(x) - \inf_y \hat{f}(y) = \frac{1}{2}\lambda(x)^2$$

- equal to the norm of the Newton step in the quadratic Hessian norm

$$\lambda(x) = \left(\Delta x_{\mathrm{nt}}^T \nabla^2 f(x) \Delta x_{\mathrm{nt}}\right)^{1/2}$$

- directional derivative in the Newton direction: $\nabla f(x)^T \Delta x_{\mathrm{nt}} = -\lambda(x)^2$
- affine invariant (unlike $\|\nabla f(x)\|_2$)

Newton's method (Algorithm)

---

given a starting point $x \in \mathbf{dom}\, f$, tolerance $\epsilon > 0$.

repeat

1. *Compute the Newton step and decrement.*
   $\Delta x_{\mathrm{nt}} := -\nabla^2 f(x)^{-1} \nabla f(x)$; $\quad \lambda^2 := \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x)$.
2. *Stopping criterion.* **quit** if $\lambda^2/2 \leq \epsilon$.
3. *Line search.* Choose step size t by backtracking line search.
4. *Update.* $x := x + t\Delta x_{\mathrm{nt}}$.

---

affine invariant, *i.e.*, independent of linear changes of coordinates:

Newton iterates for $\tilde{f}(y) = f(Ty)$ with starting point $y^{(0)} = T^{-1}x^{(0)}$ are

$$y^{(k)} = T^{-1}x^{(k)}$$

● x, y가 서로 다른 값으로 시작해도 결국 같은 값으로 수렴함을 의미

Convergence analysis

■ Assumptions

- f strongly convex on S with constant m
- $\nabla^2 f$ is Lipschitz continuous on S, with constant $L > 0$:

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_2 \leq L\|x - y\|_2$$

(L measures how well f can be approximated by a quadratic function)

(L이 작다 → hessian의 차이가 크지 않다 → function f가 quadratic한 형태에 가깝다)

Convergence analysis

■ Outline: there exists constants $\eta \in (0, m^2/L), \gamma > 0$ such that

- if $\|\nabla f(x)\|_2 \geq \eta$, then $f(x^{(k+1)}) - f(x^{(k)}) \leq -\gamma$
- if $\|\nabla f(x)\|_2 < \eta$, then

$$\frac{L}{2m^2}\|\nabla f(x^{(k+1)})\|_2 \leq \left(\frac{L}{2m^2}\|\nabla f(x^{(k)})\|_2\right)^2$$

첫번째 식의 의미
※ gradient 값이 eta보다 같거나 크면 → 다음 iteration에서의 값이 최소 -gamma만큼 감소

두번째 식의 의미
※ backtracking linesearch에서의 t=1 (한번만에 멈춤) and 첫번째 식은 언젠가 두번째 식을 만족하게 되는데,
한번 두번째 식을 만족하면 recursive하게 계속 두번째 식을 만족

Convergence analysis

- Outline: there exists constants $\eta \in (0, m^2/L), \gamma > 0$ such that
  - if $\|\nabla f(x)\|_2 \geq \eta$, then $f(x^{(k+1)}) - f(x^{(k)}) \leq -\gamma$
  - if $\|\nabla f(x)\|_2 < \eta$, then

$$\frac{L}{2m^2}\|\nabla f(x^{(k+1)})\|_2 \leq \left(\frac{L}{2m^2}\|\nabla f(x^{(k)})\|_2\right)^2$$

Stopping criterion

$$f(x) - p^\star \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$$

In Gradient descent

$$f(x^{(k)}) - p^\star \leq c^k(f(x^{(0)}) - p^\star)$$

Convergence analysis

- damped Newton phase ($\|\nabla f(x)\|_2 \geq \eta$)
  - most iterations require backtracking steps
  - function value decreases by at least γ
  - if p* > $-\infty$, this phase ends after at most $(f(x^{(0)}) - p^*)/\gamma$ iterations
- quadratically convergent phase ($\|\nabla f(x)\|_2 < \eta$)
  - all iterations use step size t=1
  - $\|\nabla f(x)\|_2$ converges to zero quadratically: if $\|\nabla f(x^{(k)})\|_2 < \eta$, then

$$\frac{L}{2m^2}\|\nabla f(x^l)\|_2 \leq \left(\frac{L}{2m^2}\|\nabla f(x^k)\|_2\right)^{2^{l-k}} \leq \left(\frac{1}{2}\right)^{2^{l-k}}, \qquad l \geq k$$

Conclusion

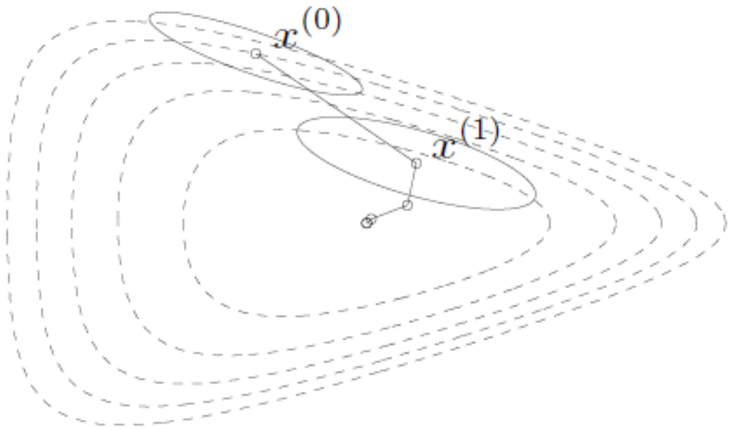
number of iterations until $f(x) - p^* \leq \epsilon$ is bounded above by

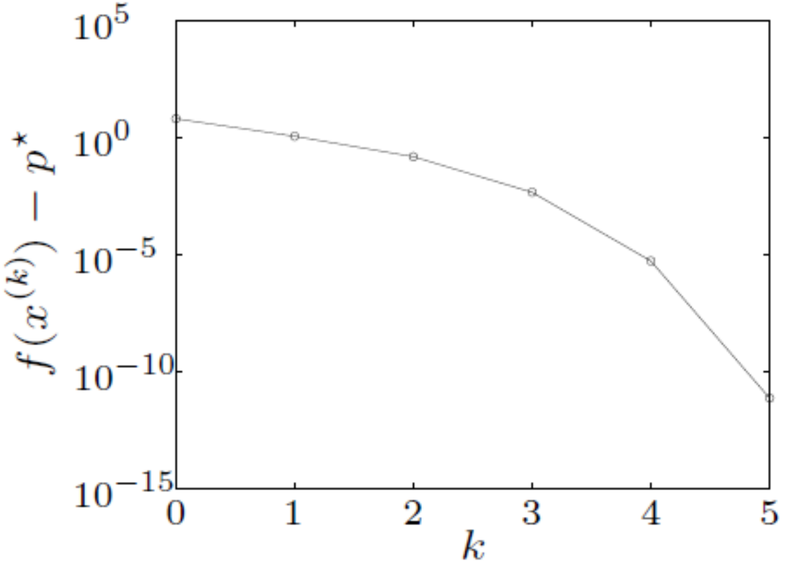
$$\frac{f(x^{(0)}) - p^\star}{\gamma} + \log_2 \log_2(\epsilon_0/\epsilon)$$

- γ, ϵ_0 are constants that depend on m, L, $x^{(0)}$
- second term is small (of the order of 6) and almost constant for practical purposes
- in practice, constants m, L (hence γ, ϵ_0) are usually unknown
- provides qualitative insight in convergence properties ($i.e.$, explains two algorithm phases)

Examples

■ in $\boldsymbol{R^2}$ $\quad f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}.$

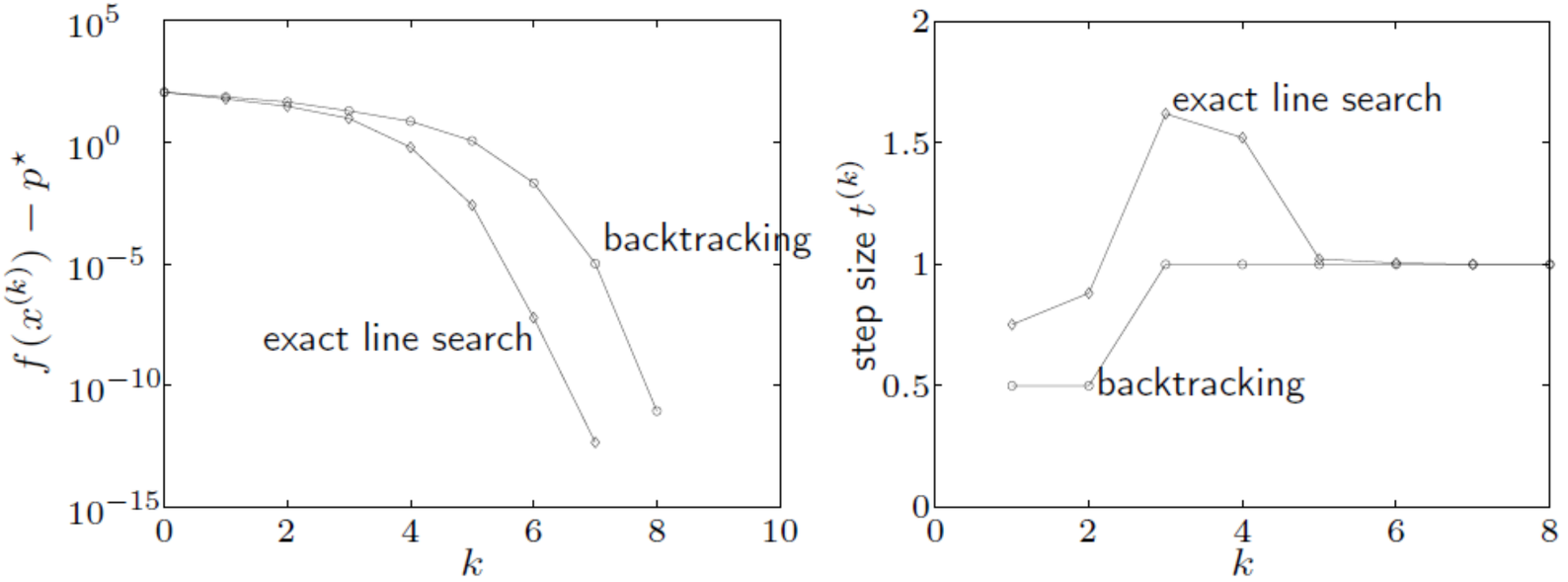




- backtracking parameters $\alpha = 0.1$, $\beta = 0.7$
- converges in only 5 steps
- quadratic local convergence

Examples

- in $\boldsymbol{R^{100}}$

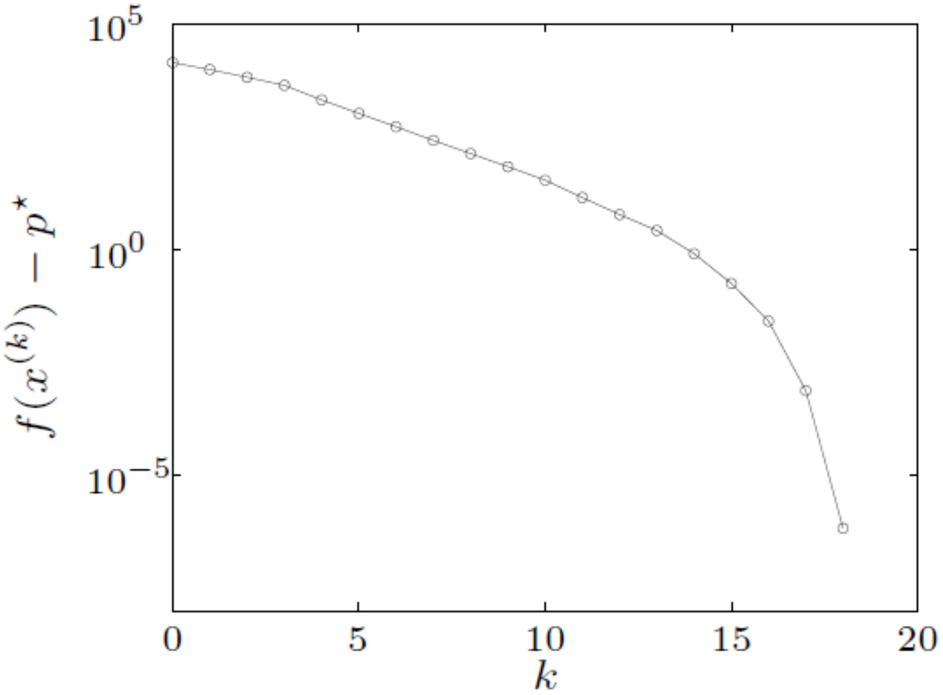


- backtracking parameters $\alpha = 0.01$, $\beta = 0.5$
- backtracking line search almost as fast as exact l.s. (and much simpler)
- clearly shows two phases in algorithm

Examples

- in $\boldsymbol{R^{10000}}$

$$f(x) = -\sum_{i=1}^{10000} \log(1 - x_i^2) - \sum_{i=1}^{100000} \log(b_i - a_i^T x)$$



- backtracking parameters $\alpha = 0.01$, $\beta = 0.5$.
- performance similar as for small examples

Summary

■ Advantages

● ***Convergence of Newton's method is rapid in general***, and quadratic near x⋆.
Once the quadratic convergence phase is reached, at most six or so iterations are required to produce a solution of very high accuracy.

● ***Newton's method is affine invariant.*** It is insensitive to the choice of coordinates, or the condition number of the sublevel sets of the objective.

● ***Newton's method scales well with problem size.*** Its performance on problems in R^10000 is similar to its performance on problems in R^10, with only a modest increase in the number of steps required.

● The good performance of Newton's method is ***not dependent on the choice of algorithm parameters***.
In contrast, the choice of norm for steepest descent plays a critical role in its performance.

9.7 Implementation

Implementation

main effort in each iteration: evaluate derivatives and solve Newton system
(computational cost를 줄이기 위한 방법들 소개)

$$\Delta x_{\mathrm{nt}} = -\nabla^2 f(x)^{-1} \nabla f(x)$$

$$H\Delta x = -g \quad \text{(where } H = \nabla^2 f(x),\ g = \nabla f(x)\text{)}$$

■ via Cholesky factorization

$$H = LL^T, \qquad \Delta x_{\mathrm{nt}} = -L^{-T} L^{-1} g, \qquad \lambda(x) = \|L^{-1} g\|_2$$

- cost $(1/3)n^3$ flops for unstructured system
- cost $\ll (1/3)n^3$ if H sparse, banded

Implementation

- example of dense Newton system with structure

$$f(x) = \sum_{i=1}^{n} \psi_i(x_i) + \psi_0(Ax + b), \qquad H = D + A^T H_0 A$$

(f is a separable function, plus a function that depends on a low dimensional affine function of x)

- assume $A \in \mathbf{R}^{p \times n}$, dense, with $p \ll n$
- D diagonal with diagonal elements $\psi_i''(x_i)$; $H_0 = \nabla^2 \psi_0(Ax + b)$

- method1: from H, solve via dense Cholesky factorization: cost (1/3)n^3
- method2: factor $H_0 = L_0 L_0^T$ (textbook pg.512)

write Newton system as $D\Delta x + A^T L_0 w = -g, \qquad L_0^T A \Delta x - w = 0$

eliminate Δx from first equation; compute w and Δx from

$$(I + L_0^T A D^{-1} A^T L_0) w = -L_0^T A D^{-1} g, \qquad D\Delta x = -g - A^T L_0 w$$

cost: $2p^2 n$ (dominated by computation of $L_0^T A D^{-1} A^T L_0$)

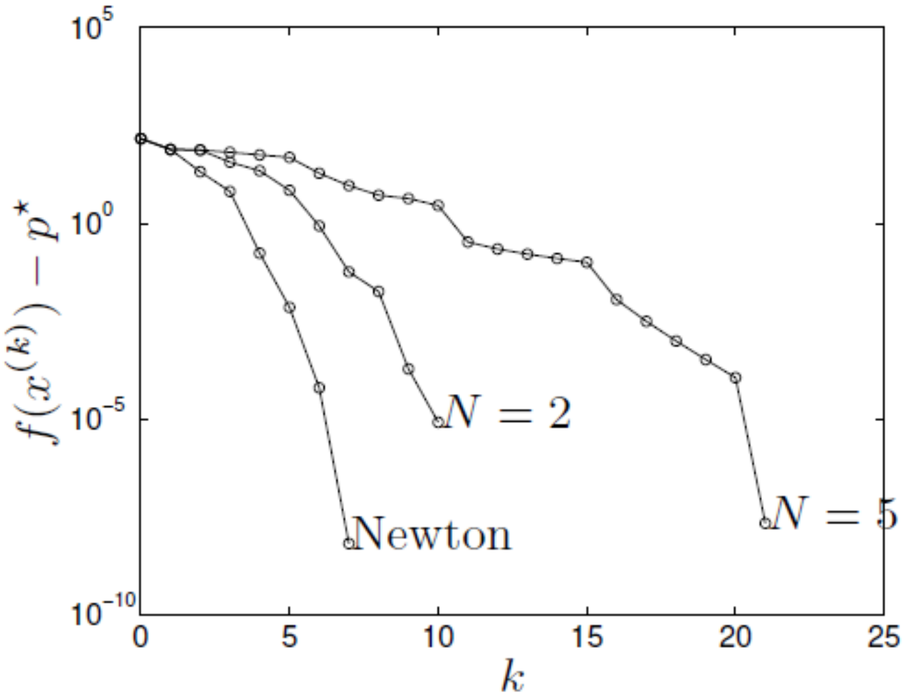
9. Examples

Example 9.31

■ 9.31 (Some approximate Newton methods)

- Newton 방법에서 Hessian과 cost를 매번 구하는 것은 어려움
→ replace the Hessian by a positive definite approximation that makes it easier

- **(a) *Re-using the Hessian* : evaluate and factor the Hessian only every N iterations**



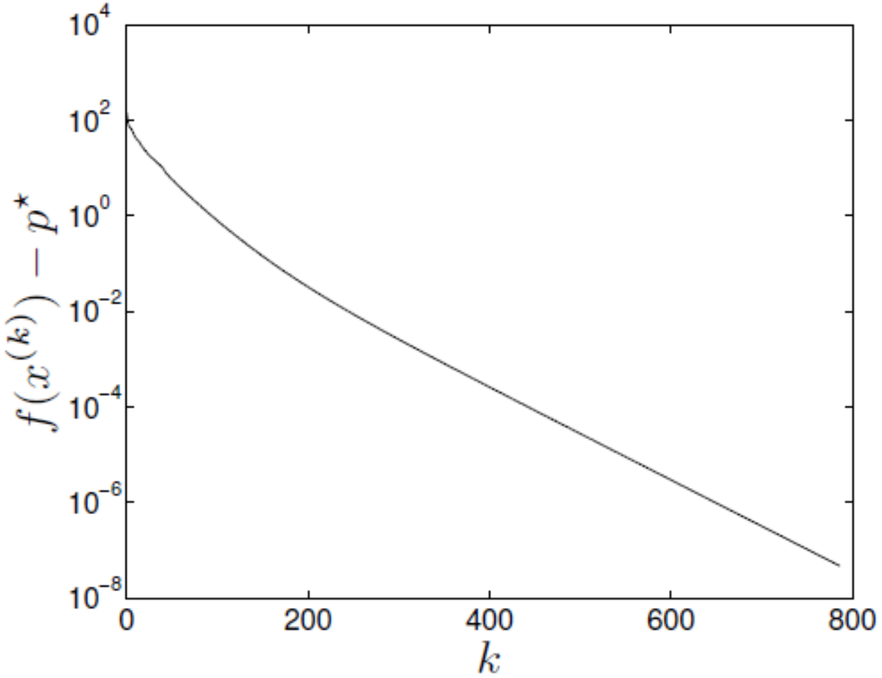
- **Result : the speed of convergence deteriorates rapidly as N increases**

Example 9.31

■ 9.31 (Some approximate Newton methods)

- Newton 방법에서 Hessian과 cost를 매번 구하는 것은 어려움
→ replace the Hessian by a positive definite approximation that makes it easier

● **(b) *Diagonal approximation* : replace the Hessian by its diagonal ($\partial^2 f(x)/\partial x_i^2$ 만 계산)**



● **Result : the algorithm converges very much like the gradient method**

Example 9.32

■ 9.32 (Gauss-Newton method for convex nonlinear least-squares problems)

- **In optimization problem, Newton method needs second derivative, but Gauss-Newton method only use first derivative.**

- **We minimize a function of the form**

$$f(x) = \frac{1}{2}\sum_{i=1}^{m} f_i(x)^2$$ **where f_i are twice differentiable**

- **The gradient and Hessian of f at x are given by**

$$\nabla f(x) = \sum_{i=1}^{m} f_i(x)\nabla f_i(x), \qquad \nabla^2 f(x) = \sum_{i=1}^{m} \left(\nabla f_i(x)\nabla f_i(x)^T + f_i(x)\nabla^2 f_i(x)\right)$$

- **The Gauss-Newton method uses the search direction**

$$\Delta x_{\mathrm{gn}} = -\left(\sum_{i=1}^{m} \nabla f_i(x)\nabla f_i(x)^T\right)^{-1}\left(\sum_{i=1}^{m} f_i(x)\nabla f_i(x)\right)$$

Example 9.32

■ 9.32 (Gauss-Newton method for convex nonlinear least-squares problems)

- **Test Gauss-Newton method on some problem instances of the form**

$$f_i(x) = (1/2)x^T A_i x + b_i^T x + 1$$

■ Solution

- **Non-linear model can be written as (using the first-order approximation)**

$$f_i(x+v) \approx f_i(x) + \nabla f_i(x)^T v$$

- **We obtain the approximation**

$$f(x+v) \approx \frac{1}{2}\sum_{i=1}^{m}(f_i(x) + \nabla f_i(x)^T v)^2$$

- **알고리즘 설명 : https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=jerrypoiu&logNo=221544119561**

Example 9.32

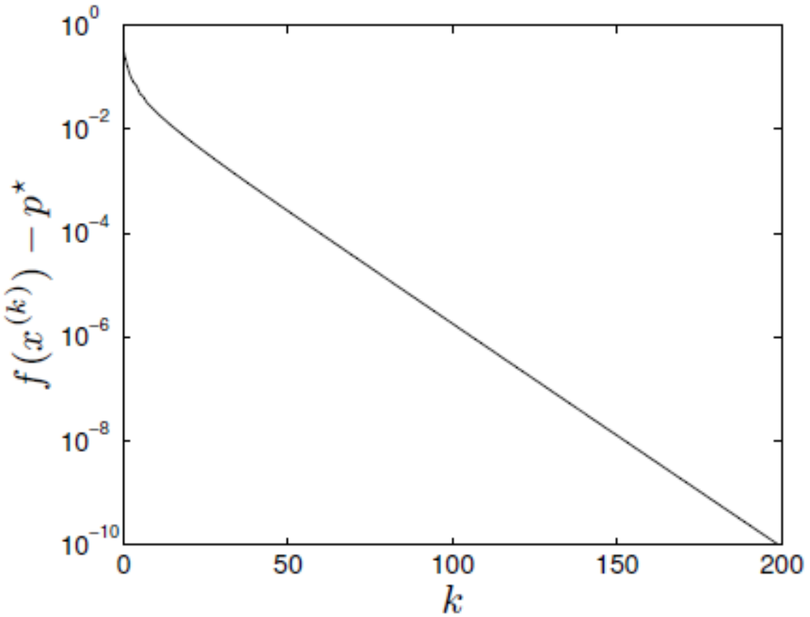
■ 9.32 (Gauss-Newton method for convex nonlinear least-squares problems)

- **Test Gauss-Newton method on some problem instances of the form**

$$f_i(x) = (1/2)x^T A_i x + b_i^T x + 1$$

■ Result

- **converges linearly, and much more slowly than Newton's method**
- **It works well if either $\nabla^2 f_i$ is small, or f_i is small**



Reference

- S.Boyd and L.Vandenberghe (2004), “Convex optimization”, Chapters 9
- https://lecture.cdsl.kr/cvxopt
- https://vimeo.com/32272323
- https://web.stanford.edu/class/ee364a/lectures.html
- https://www.stat.cmu.edu/~ryantibs/convexopt-S15/lectures/14-newton.pdf
- https://darkpgmr.tistory.com/142
- 2011년 2학기 서울대학교 전기공학부 대학원 강의, thanks to 심형보 교수님

감사합니다