Vishnoi's book, Chapter 6 Mar 31, Lecture 23

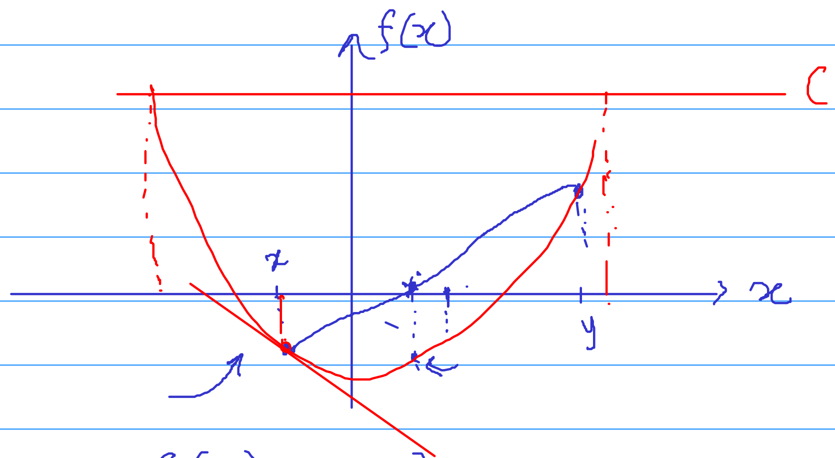
Gradient Descent

- For convex minimization

Def: A function $f : K \to \mathbb{R}$ (K is convex) is said to be convex if

for all $x, y \in K$ and $\lambda \in [0,1]$

$$f(\lambda x + (1-\lambda) y) \leq \lambda f(x) + (1-\lambda) f(y)$$



Examples: $f(x) = x^2 + 5x + 3$

$f(x_1, x_2) = \dfrac{1}{x_1 x_2}$ for $x_1 \geq 0,\ x_2 \geq 0$

$f(x) = \sin x$ for $\pi \leq x \leq 2\pi$

Equivalently,

- $\{x : f(x) \leq c\}$ is convex for all $c \in \mathbb{R}$

- Univariate $\rightarrow$ $f'(x)$ is non-decreasing.

  $\rightarrow$ $f''(x) \geq 0$

  $\rightarrow$ Above the linear approximation (tangent)

$$f(y) \geq \underbrace{f(x) + f'(x)(y-x)}$$

Multivariate:

$$f: \mathbb{R}^n \to \mathbb{R}$$

Gradient

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n} \right)$$

Fact: Directional derivative in direction $u \in \mathbb{R}^n$

$$\frac{d}{dt} f(x+tu) = u_1 \frac{\partial f}{\partial x_1} + u_2 \frac{\partial f}{\partial x_2} + \cdots + u_n \frac{\partial f}{\partial x_n}$$

$$= \langle \nabla f, u \rangle = u^T \nabla f$$

Claim A differentiable function f is convex if and only if

(above the linear approximation)

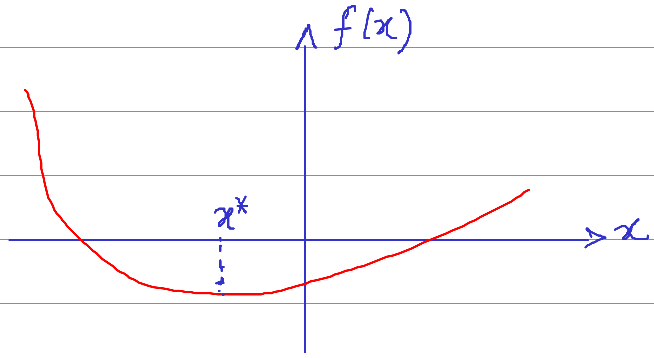
$\forall\ x, y \in K$

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

Unconstrained convex minimization

Convex $f: \mathbb{R}^n \longrightarrow \mathbb{R}$

find $x^* \in \mathbb{R}^n$ s.t. $f(x^*) \leq f(x)\ \forall\ x \in \mathbb{R}$

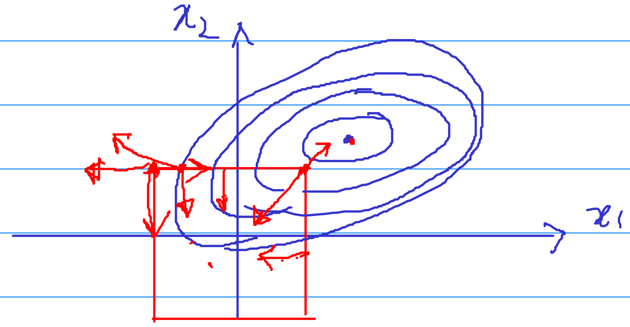


Claim x^* is a minimizing point iff

$$\nabla f(x^*) = 0 \qquad \left(\frac{\partial f}{\partial x_1} = 0,\ \frac{\partial f}{\partial x_2} = 0, \ldots, \frac{\partial f}{\partial x_n} = 0 \right)$$

Above claim not true in the constrained setting.

$\nabla f(x^*)$ is a positive combination of the tight constraints at x^*



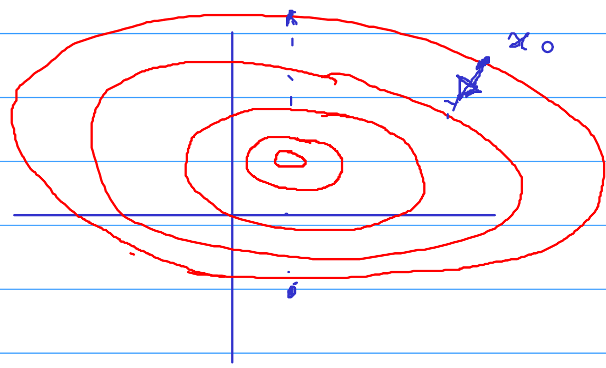
$f(x_1, x_2)$

Gradient Descent (unconstrained)

Iterative $x_0 \to x_1 \to x_2 \to \cdots \to x_T$

Move in the direction of $-\nabla f(x)$

Why?



$x_i \longrightarrow x_i + tu$

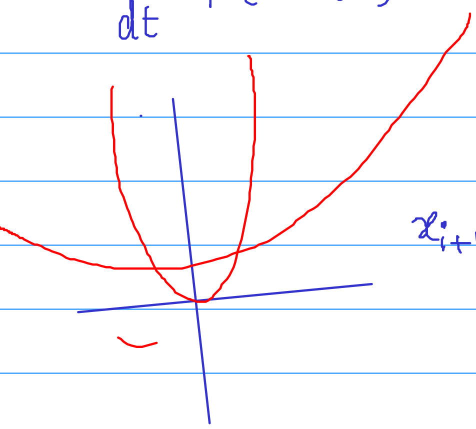
for small $t \in \mathbb{R}$

$u \in \mathbb{R}^n$ unit vector

$f(x_1, x_2)$

$$\frac{d}{dt} f(x+tu) = \langle \nabla f(x), u \rangle$$

choose $u = -\dfrac{\nabla f(x)}{\|\nabla f(x)\|}$



$$x_{i+1} \longleftarrow x_i - \gamma \nabla f(x_i)$$

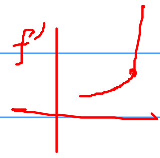
for some constant $\gamma > 0$

step size

To decide the step size we need

- Lipschitz continuity of gradient

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



for every $x, y \in \mathbb{R}^n$,

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

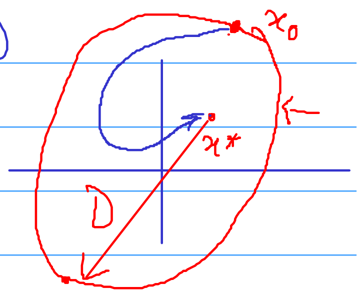
for some constant $L > 0$

Other parameters

- Good initial point (not far from x^*)

$$\max \{ \|x - x^*\| : f(x) \leq f(x_0) \} \leq D$$

$$\Rightarrow \|x_0 - x^*\| \leq D$$



- Error Parameter ε

<u>Theorem</u>: Given ε, L, D, we can choose γ so that in $T = O(LD^2/\varepsilon)$ iterations, we get x with $f(x) \leq f(x^*) + \varepsilon$

If L, D constant then no. of iterations $O(1/\varepsilon)$ (independent of n).

Analysis:

Lemma $\forall\, x, y \in \mathbb{R}^n$

$$f(y) - \left(f(x) + \langle \nabla f(x), y - x \rangle\right) \leq \frac{L}{2} \|y - x\|^2$$

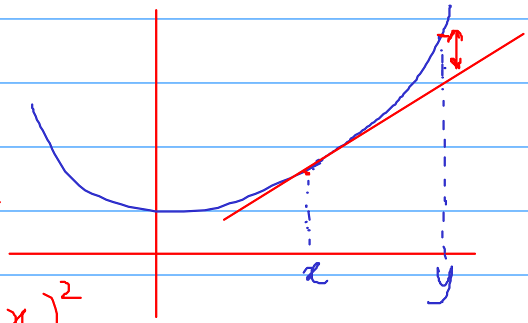
Proof sketch

$$f(y) - f(x) = \int_x^y f'(\alpha)\, d\alpha$$

$g(t)$
$= f(x + t(y-x))$
$t \in [0,1]$

$$\leq \int_x^y \left[f'(x) + L(\alpha - x)\right] d\alpha$$

$$= f'(x)(y-x) + \frac{L}{2}(y-x)^2$$



$$f(x_{i+1}) - f(x_i) \leq \langle \nabla f(x_i), \underbrace{x_{i+1} - x_i}_{-\gamma \nabla f(x_i)} \rangle + \frac{L}{2} \|\underbrace{x_{i+1} - x_i}\|^2$$

$$= -\gamma \|\nabla f(x_i)\|^2 + \frac{L}{2}\gamma^2 \|\nabla f(x_i)\|^2$$

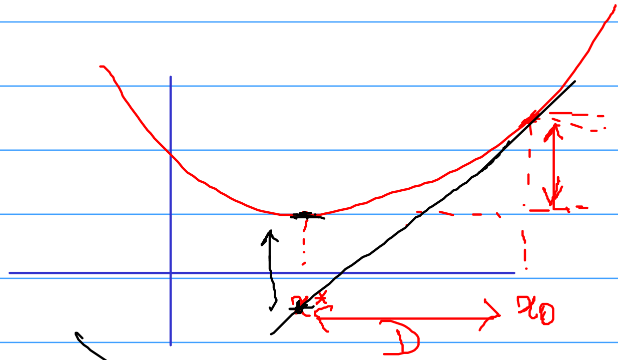
minimize $-\gamma + \dfrac{\gamma^2 L}{2}$

Choose $\gamma = 1/L$ ← step size

$$f(x_{i+1}) - f(x_i) \leq \frac{-1}{2L} \underbrace{\|\nabla f(x_i)\|^2} \quad \text{——} \quad (1)$$

Define

$$R_i = f(x_i) - f(x^*)$$



② $R_0 = f(x_0) - f(x^*) \leq LD^2$

$\|\nabla f(x_0)\| \leq LD$

$$f(x^*) \geq f(x_0) + \langle \underbrace{\nabla f(x_0)}_{\leq LD}, \underbrace{x^* - x_0}_{\leq D} \rangle$$

from ①

③ $$R_i - R_{i+1} = f(x_i) - f(x_{i+1}) \geq \frac{1}{2L} \|\nabla f(x_i)\|^2$$

Similar as above

$$R_i \leq D \cdot \|\nabla f(x_i)\|$$

$$\|\nabla f(x_i)\| \geq \frac{R_i}{D}$$

From ③ $$R_i - R_{i+1} \geq \frac{1}{2L} \frac{R_i^2}{D^2}$$

How many iterations to go from LD^2 to ε

$R_i^2 \to R_i^2/4$

Going from R_i to $R_i/2$

$$k \cdot \frac{R_i^2}{2LD^2 \cdot 4} \approx \frac{R_i}{2} \qquad k \approx \frac{4LD^2}{R_i}$$

$$R_0 \longrightarrow \frac{R_0}{2} \longrightarrow \frac{R_0}{4} \; - - - - \rightarrow \; \varepsilon$$

$$\frac{4LD^2}{R_0} + \frac{4LD^2}{R_0} \times 2 + \frac{4LD^2 \times 4}{R_0} + \cdots \frac{4LD^2}{R_0} \cdot 2^r$$

r = no. of phases

$= \log\left(\frac{R_0}{\varepsilon}\right)$

Total number of iterations $\leq \frac{4LD^2 \times 2^{r+1}}{R_0} = O\left(\frac{LD^2}{\varepsilon}\right)$

What if L not known?

Is dependence on ε optimal?

A lower bound of $\frac{1}{\sqrt{\varepsilon}}$ is known.

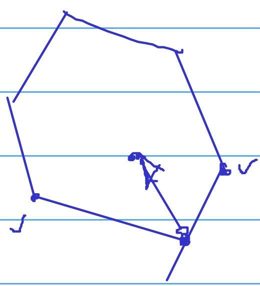
accelerated GD achieves $O(1/\sqrt{\varepsilon})$

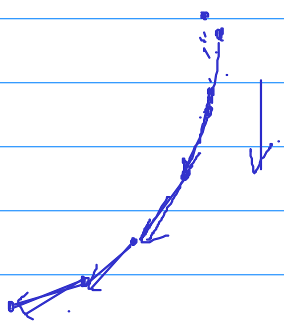
In other words for ℓ bit precision, we need 2^{ℓ} iterations

Pseudopolynomial time

$\boxed{\text{Apr 4, Lec 24}}$

Constrained convex minimization.





Gradient direction may lead us outside the feasible region.

$\text{proj}_K(x) \leftarrow$ point in K closest to x.

Similar convergence guarantees can be shown.

Can we apply this to solve LPs?

Not clear how to project to the given polytope.

Different Approach

Suppose we want to test whether there is a perfect matching in a given bipartite graph G

Equivalently whether following LP is feasible.

$$\begin{cases} x_e \geq 0 & \text{for } e \in E \\ \sum_{e \in \delta(v)} x_e = 1 & \text{for } v \in V. \end{cases}$$

Projection operator for this polytope might not be easy.

Define $Q = \{ x \in \mathbb{R}^E : x_e \geq 0 \text{ for } e \in E \}$

$H = \{ x \in \mathbb{R}^E : \sum_{e \in \delta(v)} x_e = 1 \text{ for } v \in V \}$

Que: is $Q \cap H$ empty?

Define

$$d_H(x) = \text{dist}(x, H)$$

$$\min_{x \in Q} d_H^2(x) = 0 \quad \text{iff} \quad Q \cap H \text{ is non-empty.}$$

$$d_Q(x) = \text{dist}(x, Q)$$

$$\rightarrow \quad \min_{x \in H} d_Q^2(x) = 0 \quad \text{iff} \quad Q \cap H \text{ is non-empty.}$$

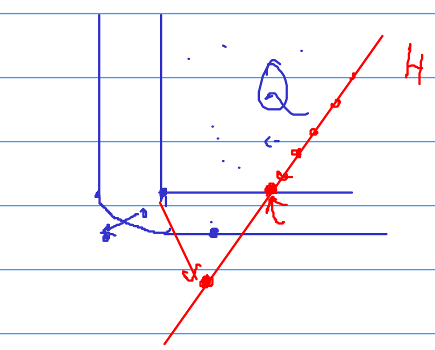
L, D, ε Projected gradient descent.

Projection on H is simple $Ax = b$

$(z - x)$

$$d_Q^2(x) = \sum_{e : x_e < 0} x_e^2$$

$$\nabla d_Q^2(x)(e) = \begin{cases} 0 & x_e \geq 0 \\ 2x_e & x_e < 0 \end{cases}$$



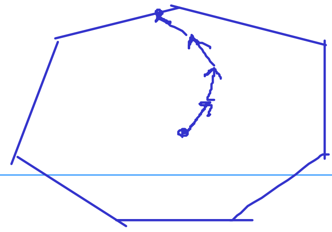
$L = 2$ ✓

Initial feasible Point

Claim $D = O(\sqrt{m})$.

Claim $\varepsilon = \frac{1}{m}$ is good enough.

If $Q \cap H$ is empty then $\text{dist}(Q, H) > \frac{1}{m}$



Interior Point Methods

→ Karmarkar 1984 showed that LPs can be solved by IPM in polynomial time.

→ Beats Simplex in some cases

→ Also applies to more general convex programs.

→ Need objective function and constraints explicitly

Ref: Vishnoi Chapter 9, 10

$\min_x f(x)$

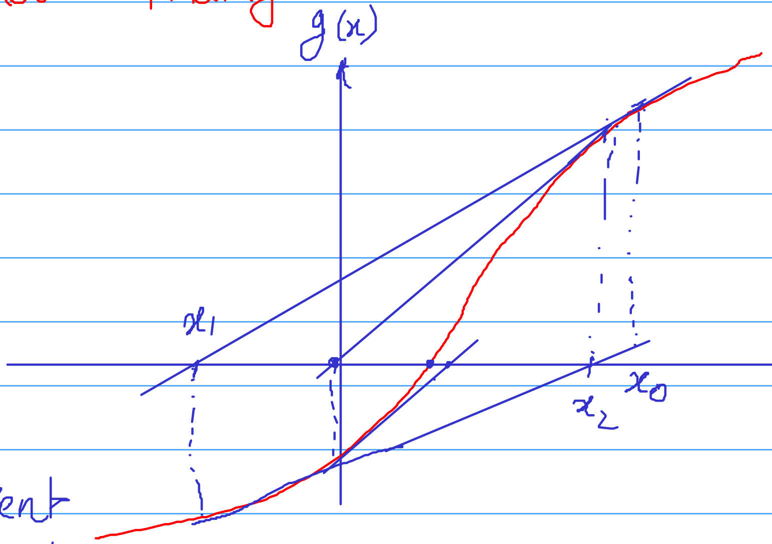
$f'(x) = 0$

Newton's method for Root finding

$g(x) = f'(x)$

find Root of $g(x)$

x_0



Much faster than gradient descent

$$\tilde{g}_0(x) = g'(x_0)(x - x_0) + g(x_0)$$

$$g'(x_0)(x_1 - x_0) + g(x_0) = 0$$

$$x_1 \leftarrow x_0 - \frac{g(x_0)}{g'(x_0)} \rightarrow g'(x_0)(x_0 - x_0) = g(x_0)$$

$$x_1 \leftarrow x_0 - \frac{f'(x_0)}{f''(x_0)} \qquad x_1 \leftarrow x_0 - \frac{1}{L} f'(x_0)$$

$\frac{1}{2}, \frac{1}{4}, \frac{1}{16}, \dots$

t iterations.

$\frac{1}{2^{2^t}}$

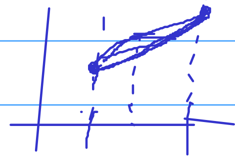
Quadratic Convergence.

- $|x_1 - r| \leq M \, |x_0 - r|^2$

ε-close

$t = \log\log\frac{1}{\varepsilon}$

Mean Value theorem



$\exists\, \alpha \in (r, x_0)$

s.t. $\quad g'(\alpha) = \dfrac{g(x_0) - g(r)}{x_0 - r}$

$$g(r) = g(x_0) + g'(\alpha)(r - x_0)$$

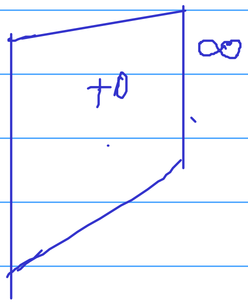
$\exists\, \alpha \in (r, x_0)$

$$g(r) = g(x_0) + g'(x_0)(r - x_0) + \underbrace{g''(\alpha) \cdot \frac{(r-x_0)^2}{2}}$$

$$0 = g'(x_0)(x_0 - x_1) + g'(x_0)(r - x_0) + g''(\alpha)\frac{(r-x_0)^2}{2}$$

$$= g'(x_0)(r - x_1) + g''(\alpha)\frac{(r-x_0)^2}{2}$$

$$|x_1 - r| = \frac{|g''(\alpha)|}{2|g'(x_0)|}(r - x_0)^2$$



$$M := \max_{\alpha \in (x_0, r)} \frac{|g''(\alpha)|}{2|g'(x_0)|} \qquad \frac{|f'''(\alpha)|}{2|f''(x_0)|}$$

only when $|r - x_0| < 1$

Newton's method in multivariate case

$f: \mathbb{R}^n \to \mathbb{R}$ (twice differentiable)

Degree 2 approximation around x_0

$$\tilde{f}(x) = f(x_0) + (x-x_0)^T \nabla f(x_0) + \frac{1}{2}(x-x_0)^T \nabla^2 f(x_0)(x-x_0)$$

Here $\nabla^2 f(x_0)$ is the Hessian matrix whose (i,j) entry is $\frac{\partial^2 f}{\partial x_i \partial x_j}(x_0)$

$\nabla^2 f(x_0) \succeq 0$ iff f is convex.

$\nabla^2 f(x_0)$ is invertible if f is strictly convex.

Newton step goes to the point x_1 minimizing $\tilde{f}(x)$

So, $\nabla \tilde{f}(x_1) = 0$

$$\Rightarrow \quad \nabla f(x_0) + \nabla^2 f(x_0)(x_1 - x_0) = 0$$

$$\boxed{x_1 = x_0 - [\nabla^2 f(x_0)]^{-1} \nabla f(x_0)}$$

Newton's method converges if the starting point is close enough to the minimizing point x^*.

When we are not sure about closeness to x^*, we can used damped Newton's method.

$$x_1 = x_0 - \alpha [\nabla^2 f(x_0)]^{-1} \nabla f(x_0)$$

$0 < \alpha < 1$ is chosen appropriately so that $f(x_1) < f(x_0)$

Interior point methods were recently applied to find an $\tilde{O}(m)$ time algorithm for max flow.

Best known combinatorial algorithm is $O(m\sqrt{n})$.

Barrier function

Idea is to forget about constraints and add an appropriate Barrier function to the objective function which keeps you away from going outside the boundary.

$$\begin{array}{l} \min f(x) \\ \text{s.t. } Ax \leq b \end{array} \longrightarrow \min f(x) + B(x)$$

Ideal choice $\quad B(x) = \begin{cases} 0 & \text{if } Ax \leq b \text{ satisfied} \\ \infty & \text{otherwise.} \end{cases}$

But we need a smooth barrier function should be

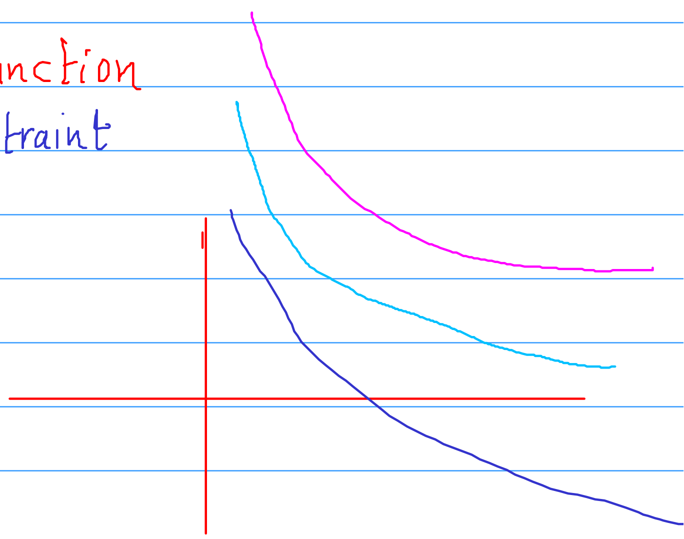
1. Strictly convex $\nabla^2 B \succ 0$
2. Go to ∞ when go close to the boundary
3. Should decrease as we move away from boundary

Candidates for Barrier function

Say $x \geq 0$ is a constraint

$-\log x$, $1/x$, $e^{1/x}$

Used most often



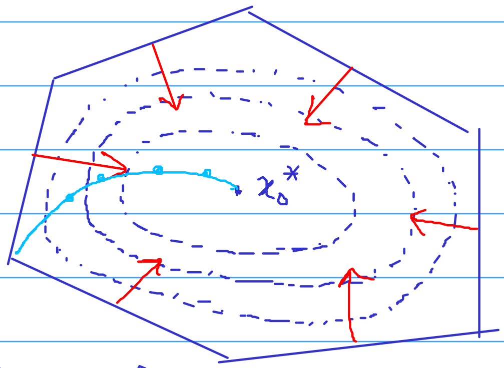
Each constraint will contribute to the Barrier function

$$\min\ w^T x \quad \text{s.t.} \quad a_i^T x \leq b_i \text{ for } 1 \leq i \leq k \longrightarrow \min\ \eta\, w^T x + \sum_{i=1}^{k} -\log(b_i - a_i^T x)$$

Define

x_η^* as point minimizing

$$\phi_\eta(x) = \eta w^T x + \sum_{i=1}^{k} -\log(b_i - a_i^T x)$$



$x_0^* \longleftarrow$ Some kind of center of the polytope

Define x^* as point minimizing $w^T x$ s.t. $Ax \leq b$.

Claim $\lim_{\eta \to \infty} x_\eta^* = x^*$

Central path $= \{ x_\eta^* : \eta \geq 0 \}$

Let's see what should be η for a desired ε-closeness to the actual optimal value.

For any η, x_η^* will satisfy

$$\nabla \phi_\eta(x_\eta^*) = 0$$

$$\Rightarrow \eta w + \sum_{i=1}^{k} \frac{a_i}{(b_i - a_i^T x_\eta^*)} = 0$$

This can be used to give a lower bound on the optimal value $w^T x^*$

$$\omega^T x^* = \frac{-1}{\eta} \sum_{i=1}^{k} \frac{a_i^T x^*}{b_i - a_i^T x_\eta^*}$$

use the fact that $a_i^T x^* \leq b_i$

$$\omega^T x^* \geq \frac{-1}{\eta} \sum_{i=1}^{k} \frac{b_i}{b_i - a_i^T x_\eta^*}$$

Let's now see objective value at x_η^*

$$\omega^T x_\eta^* = \frac{-1}{\eta} \sum_{i=1}^{k} \frac{a_i^T x_\eta^*}{b_i - a_i^T x_\eta^*}$$

From above two we get

$$\omega^T x_\eta^* - \omega^T x^* \leq \sum_{i=1}^{k} \frac{1}{\eta} = \frac{k}{\eta}$$

Thus for desired accuracy ε,

set $\eta = \frac{k}{\varepsilon}$.

By minimizing

$$\frac{k}{\varepsilon} \omega^T x + \sum_{i=1}^{k} -\log\left(b_i - a_i^T x\right)$$

we get a point whose objective value is ε-close to optimal.

Note that ε may be required to be 2^{-n}. Gradient descent or damped Newton's method may be too slow.

Apr 11, Lecture 26.

Newton's Step

If the function satisfies the required conditions about the second / third derivatives

then if x_0 is close to x^*

then $x_1 = x_0 - [\nabla^2 \phi(x_0)]^{-1} \nabla \phi(x_0)$

is very close to x^*.

close and very close are quantified in terms of the following quantity

$$\nabla \phi(x_0)^T \left[\nabla^2 \phi(x_0) \right]^{-1} \nabla \phi(x_0)$$

Algorithm

Given : η_0 and a point x_0 which is very close to $x^*_{\eta_0}$

for $t = 1, 2, \dots$

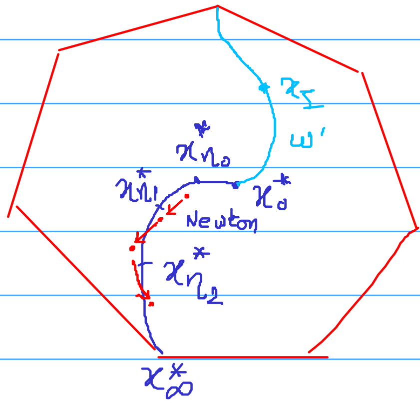
$$\eta_t = \eta_{t-1} \times \left(1 + \frac{1}{\sqrt{n}}\right)$$

If x_{t-1} is very close to $x^*_{\eta_{t-1}}$

then x_{t-1} is close to $x^*_{\eta_t}$

$$x_t \leftarrow x_{t-1} - \left[\nabla^2 \phi_{\eta_t}(x_{t-1})\right]^{-1} \nabla \phi_{\eta_t}(x_{t-1})$$

Stop when: $\eta_t \geq \frac{k}{\varepsilon}$



$$\eta_0 = \exp(-n)$$

$$\eta_0 \left(1 + \frac{1}{\sqrt{n}}\right)^T = \frac{K}{\varepsilon}$$

$$T = \text{poly}(n) \cdot \log K/\varepsilon$$

To find η_0 and $x^*_{\eta_0}$, we will start from the given initial feasible point x_I and to a reverse walk towards the center. (opposite of the above algo)

Can we find ω' s.t. x_I lies on the central path corresponding to ω'.

$$\nabla\left(\eta' \omega'^T x + B(x)\right) \text{ at } x_I = 0$$

$$\eta' \omega' + \nabla B(x_I) = 0$$

Choose $\eta' = 1$ and $\omega' = -\nabla B(x_I)$