Week 7-8

Nicolo Michelusi

I. Optimization algorithms

- Sometimes we can explicitly solve for the optimal solution in closed form, by solving KKT conditions directly, or solving the Lagrangian and dual problems (see previous examples)
- However, often a closed-form solution is not possible, and we need to resort to numerical algorithms
  - A numerical algorithm starts from some initial estimate x_0, and iteratively generate new estimates by
    $$x_{k+1} = T(x_k)$$
    Hopefully, as $k \to \infty$, $x_k \to x^*$, the optimal solution
  - When does such a sequence converges to the optimal solution?
  - If so, how long does it take to converge to a certain accuracy? (sample complexity)
- Example: compute $\sqrt{2}$ using only $+, -, \times, /$.
  $$x = \sqrt{2} \Leftrightarrow (x-1)(x+1) = 1 \Leftrightarrow x = \frac{1}{x+1} + 1$$
  This suggests the update
  $$T(x_k) = \frac{1}{x_k + 1} + 1$$
  which is such that $T(\sqrt{2}) = \sqrt{2}$ (i.e., $\sqrt{2}$ is a fixed point of $x = T(x)$)
  To prove convergence, let $x, y \geq 1$, and consider $|T(x) - T(y)|$:
  $$|T(x) - T(y)| = \frac{|y-x|}{(x+1)(y+1)} \leq \frac{1}{4}|y-x|$$
  Therefore, choosing $y = \sqrt{2}$ we get
  $$|x_{k+1} - \sqrt{2}| = |T(x_k) - \sqrt{2}| \leq \frac{1}{4}|x_k - \sqrt{2}| \leq \cdots \leq \frac{1}{4^{k+1}}|x_0 - \sqrt{2}|$$
  and therefore x_k converges linearly to $\sqrt{2}$, by initializing it with $x_0 \geq 1$.
  However, not all algorithms converge:
  $$x = \sqrt{2} \Leftrightarrow (x-1)(x+1) = 1 \Leftrightarrow x = \frac{1}{x-1} - 1$$
  but the algorithm $x_{k+1} = \frac{1}{x_k - 1} - 1$ does not converge

Week 7-8

Nicolo Michelusi

I. Optimization algorithms

- Sometimes we can explicitly solve for the optimal solution in closed form, by solving KKT conditions directly, or solving the Lagrangian and dual problems (see previous examples)
- However, often a closed-form solution is not possible, and we need to resort to numerical algorithms
  - A numerical algorithm starts from some initial estimate x_0, and iteratively generate new estimates by
  $$x_{k+1} = T(x_k)$$
  Hopefully, as $k \to \infty$, $x_k \to x^*$, the optimal solution
  - When does such a sequence converges to the optimal solution?
  - If so, how long does it take to converge to a certain accuracy? (sample complexity)
- Example: compute $\sqrt{2}$ using only $+, -, \times, /$.
  $$x = \sqrt{2} \Leftrightarrow (x-1)(x+1) = 1 \Leftrightarrow x = \frac{1}{x+1} + 1$$
  This suggests the update
  $$T(x_k) = \frac{1}{x_k + 1} + 1$$
  which is such that $T(\sqrt{2}) = \sqrt{2}$ (i.e., $\sqrt{2}$ is a fixed point of $x = T(x)$)
  To prove convergence, let $x, y \geq 1$, and consider $|T(x) - T(y)|$:
  $$|T(x) - T(y)| = \frac{|y-x|}{(x+1)(y+1)} \leq \frac{1}{4}|y-x|$$
  Therefore, choosing $y = \sqrt{2}$ we get
  $$|x_{k+1} - \sqrt{2}| = |T(x_k) - \sqrt{2}| \leq \frac{1}{4}|x_k - \sqrt{2}| \leq \cdots \leq \frac{1}{4^{k+1}}|x_0 - \sqrt{2}|$$
  and therefore x_k converges linearly to $\sqrt{2}$, by initializing it with $x_0 \geq 1$.
  However, not all algorithms converge:
  $$x = \sqrt{2} \Leftrightarrow (x-1)(x+1) = 1 \Leftrightarrow x = \frac{1}{x-1} - 1$$
  but the algorithm $x_{k+1} = \frac{1}{x_k - 1} - 1$ does not converge

II. Algorithms for Unconstrained optimization

- Solve $\min f(x)$, f convex

  Optimality condition is

$$f'(x^*; x - x^*) \geq 0, \ \forall x$$

  When f is differentiable, the optimality condition becomes

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

- Assume f differentiable; consider the iteration of the type

$$x_{k+1} = T(x_k) = x_k - \alpha \nabla f(x_k)$$

  Note that x^* is a fixed point of the mapping $T(x)$: if $x_k = x^*$, then $T(x_k) = x^*$.

- Example: $f(x) = \frac{1}{2}x^2$

  Optimal solution is: $x^* = 0$

  $\nabla f(x_k) = x_k \Rightarrow x_{k+1} = (1-\alpha)\, x_k = (1-\alpha)^{k+1} x_0$

  If $\alpha \in (0,2) \Rightarrow x_k$ converges linearly to 0

  If $\alpha = 2 \Rightarrow x_{k+1} = (-1)^{k+1} \cdot x_0 \Rightarrow$ keeps oscillating

  If $\alpha > 2 \Rightarrow x_{k+1}$ diverges for $k \to \infty$

- Note: the algorithm does not converge when α is too large; it converges slowly if α is too small..

- Proof of convergence (for $\alpha > 0$ sufficiently small). Need to show that

1) $f(x_k)$ decreases across iterations
2) $\|x_k - x^*\|_2$ decreases sufficiently fast across iterations

  Typically, we need stronger structural properties of the function, in addition to convexity

- First approach

Lemma 1. *Assume f is continuously differentiable and $\exists L > 0$ such that*

$$\|\nabla f(x) - \nabla f(y)\|_2 \le L\|x-y\|_2, \ \forall x, y, \in \mathbb{R}^n \quad *$$

(gradient is Lipschitz continuous with parameter L) Then,

$$f(y) \le f(x) + \nabla f(x)^T(y-x) + \frac{L}{2}\|x-y\|_2^2, \ \forall x, y, \in \mathbb{R}^n$$

To see this, let $v = y - x$

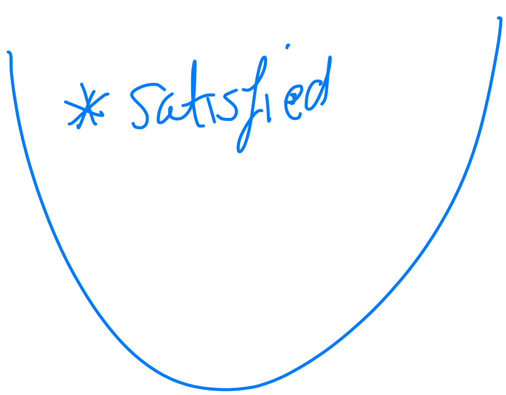
$$\Rightarrow f(y) = f(x) + \int_0^1 \nabla f(x+tv)^T v \, dt$$

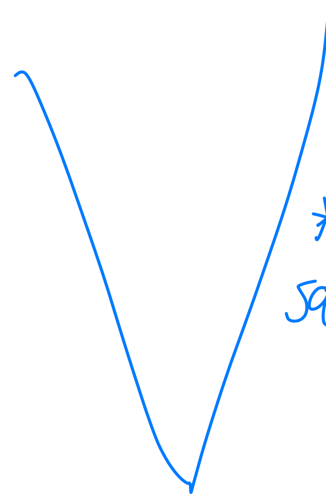
$$= f(x) + \nabla f(x)^T v + \int_0^1 [\nabla f(x+tv) - \nabla f(x)]^T v \, dt$$

$$\le f(x) + \nabla f(x)^T v + \int_0^1 \|\nabla f(x+tv) - \nabla f(x)\|_2 \|v\|_2 \, dt$$

$$\le f(x) + \nabla f(x)^T v + \int_0^1 L t \|v\|_2^2 \, dt$$

$$= f(x) + \nabla f(x)^T v + \frac{1}{2} L \|v\|_2^2$$





* not satisfied

•

Theorem 2. *Assume the same conditions as before hold; f is bounded below by f^*; and $0 < \alpha < 2/L$. Then $\nabla f(x_k) \to 0$ for $k \to \infty$.*

$$f(x_{n+1}) = f(x_n - \alpha \nabla f(x_n)) \leq$$

$$\leq f(x_n) - \alpha \nabla f(x_n)^T \nabla f(x_n) + \frac{1}{2} L \alpha^2 \|\nabla f(x_n)\|_2^2$$

$$= f(x_n) - \alpha \left(1 - \frac{\alpha L}{2}\right) \|\nabla f(x_n)\|_2^2$$

so that $f(x_n)$ is a non-increasing sequence.

Compute the sum to get:

$$f(x_n) - f(x_0) \leq -\alpha \left(1 - \frac{\alpha L}{2}\right) \underbrace{\sum_{k=0}^{n-1} \|\nabla f(x_k)\|_2^2}_{S_n}$$

S_n is a monotonically increasing sequence $\Rightarrow \lim S_n = +\infty$ or $\lim S_n = S^* < +\infty$

We have two cases:

1) $\lim S_n = +\infty \Rightarrow f(x_n) \to -\infty \Rightarrow$ contradiction
(f bounded below)

2) $\lim S_n = S^* < +\infty$
$\Rightarrow \|\nabla f(x_n)\| \to 0$ (since S_n is monotonically increasing)

- Norm approach:

  Lemma 3. *If f is convex, then*

  $$(\nabla f(x) - \nabla f(y))^T (x - y) \geq 0,\ \forall x, y, \in \mathbb{R}^n$$

  (this holds also if ∇ is a sub-gradient)

  A mapping that satisfies this condition is called "monotone mapping"

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

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

$\Downarrow$ Do the sum

$$0 \geq \left(\nabla f(x) - \nabla f(y)\right)^T (y-x)$$

-

Lemma 4. *If f is convex, differentiable, and its gradient is Lipschitz continuous with parameter L, i.e.*

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2, \ \forall x, y, \in \mathbb{R}^n$$

then

$$(\nabla f(x) - \nabla f(y))^T(x - y) \geq \frac{1}{L}\|\nabla f(x) - \nabla f(y)\|_2^2, \ \forall x, y, \in \mathbb{R}^n$$

Fix y. Let $g(x) = f(x) - f(y) - \nabla f(y)^T(x-y)$

We have $g(y) = 0$, $\nabla g(x) = \nabla f(x) - \nabla f(y)$, $\nabla g(y) = 0$

and g is convex $\Rightarrow$ x minimizes $g(y)$

Additionally, g has Lipschitz continuous gradient:

$$\|\nabla g(z) - \nabla g(x)\| = \|\nabla f(z) - \nabla f(x)\| \leq L\|x - z\|$$

Let $z = x - \frac{1}{L}\nabla g(x)$

Then we obtain,

$$0 \leq g(z) \leq g(x) + \nabla g(x)^T(z-x) + \frac{1}{2}L\|z-x\|^2 = f(x) - f(y) - \nabla f(y)^T(x-y)$$
$$- \frac{1}{L}\nabla g(x)^T\nabla g(x) + \frac{1}{2}L\frac{1}{L^2}\|\nabla g(x)\|^2$$

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

Interchange the role of x and y to get:

$$0 \leq f(y) - f(x) - \nabla f(x)^T(y-x) - \frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2$$

Compute the sum to get the desired result

This is a consequence of f convex, differentiable, and with continuous Lipschitz gradient (previous lemma)

- **Theorem 5.** *Assume that*

$$(\nabla f(x) - \nabla f(y))^T(x-y) \geq \frac{1}{L}\|\nabla f(x) - \nabla f(y)\|_2^2,\ \forall x, y, \in \mathbb{R}^n;$$

$0 < \alpha < 2/L$ *and* $\exists x^*$ *with* $\nabla f(x^*) = 0$. *Then, the sequence of points generated by*

$$x_{k+1} = x_k - \alpha \nabla f(x_k)$$

converges, and the limit x_∞ *satisfies* $\nabla f(x_\infty) = 0$.

$$\|x_{k+1} - x^*\|_2^2 = \|x_{k+1} - x_k + x_k - x^*\|_2^2$$

$$= \|\underbrace{x_{k+1} - x_k}_{-\alpha \nabla f(x_k)}\|_2^2 + \|x_k - x^*\|_2^2 + 2(\underbrace{x_{k+1} - x_k}_{-\alpha \nabla f(x_k)})^T(x_k - x^*)$$

$$= \alpha^2 \|\nabla f(x_k) - \nabla f(x^*)\|_2^2 + \|x_k - x^*\|_2^2 - 2\alpha \underbrace{[\nabla f(x_k) - \nabla f(x^*)][x_k - x^*]}_{\geq \frac{1}{L}\|\nabla f(x_k) - \nabla f(x^*)\|_2^2}$$

$$\leq -\alpha\left(\frac{2}{L} - \alpha\right)\|\nabla f(x_k)\|_2^2 + \|x_k - x^*\|_2^2$$

$\Rightarrow$ computing the sum $\sum_{k=0}^{n-1}$:

$$0 \leq \|x_n - x^*\|_2^2 \leq \|x_0 - x^*\|_2^2 - \alpha\left(\frac{2}{L} - \alpha\right)\overbrace{\sum_{k=0}^{n-1}\|\nabla f(x_k)\|_2^2}^{S_n}$$

$\Rightarrow$ S_n is a monotonically non-decreasing sequence, and

$$S_n \leq \frac{1}{\alpha\left(\frac{2}{L} - \alpha\right)}\|x_0 - x^*\|_2^2 \quad \forall n \Rightarrow \|\nabla f(x_n)\|_2 \to 0 \text{ as } n \to \infty$$

Note: $\nabla f(x_k) \to 0$, but x_k might still be unbounded

However, if x^* is a limit point of $\{x_k, k \geq 0\} \Longrightarrow \nabla f(x^*) = 0$

i.e. x^* is optimal:

x^* limit point $\Longleftrightarrow \exists$ subsequence $\{y_k = x_{t_k}, k \geq 0\}$ with $\lim_k y_k = x^*$

$\forall \varepsilon > 0 \ \exists \ k'_\varepsilon : \|y_k - x^*\|_2 \leq \frac{\varepsilon}{L} \quad \forall k \geq k'_\varepsilon$

$\Longrightarrow \|\nabla f(y_k) - \nabla f(x^*)\|_2 \leq L \|y_k - x^*\|_2 \leq \varepsilon \quad \forall \ k \geq k'_\varepsilon$

However, $\nabla f(x_k) \to 0 \Longleftrightarrow \forall \varepsilon > 0 \ \exists \ k''_\varepsilon$ s.t. $\|\nabla f(y_k)\|_2 \leq \varepsilon \ \forall \ k \geq k''_\varepsilon$

$$\Longrightarrow \quad \varepsilon^2 \geq \|\nabla f(y_k) - \nabla f(x^*)\|_2^2 \geq \left(\|\nabla f(y_k)\|_2 - \|\nabla f(x^*)\|_2 \right)^2 \quad \forall k \geq \max\{k'_\varepsilon, k''_\varepsilon\}$$

$$\Longrightarrow \quad \|\nabla f(x^*)\|_2 \leq \|\nabla f(y_k)\|_2 + \varepsilon \leq 2\varepsilon$$

Since this holds $\forall \varepsilon > 0 \Longrightarrow \nabla f(x^*) = 0$

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

- These results prove convergence to (one) optimal point x^*. However, they do not provide guarantees on how much time it takes to converge. To this end, we need stronger conditions (e.g., strong convexity).

Theorem 6. *If f is strongly convex with Lipschitz continuous gradient with parameter L,*

$$L\|x-y\|_2^2 \geq [\nabla f(x) - \nabla f(y)]^T(x-y) \geq \rho\|x-y\|_2^2, \ \forall x, y \in \mathbb{R}^n$$

for some $\rho > 0$ (note that we must have $\rho \leq L$), and $0 < \alpha < \frac{2\rho}{L^2}$, then x_k converges to x^* with linear rate. In particular,

$$\|x_k - x^*\| \leq \xi^k \|x_0 - x^*\|$$

where $\xi = \sqrt{1 + \alpha^2 L^2 - 2\alpha\rho} \in (0, 1)$.

Same as before:

$$\|x_{k+1}-x^*\|_2^2 = \|x_{k+1}-x_k+x_k-x^*\|_2^2$$

$$= \|\underbrace{x_{k+1}-x_k}_{-\alpha\nabla f(x_k)}\|_2^2 + \|x_k-x^*\|_2^2 + 2(\underbrace{x_{k+1}-x_k}_{-\alpha\nabla f(x_k)})^T(x_k-x^*)$$

$$= \alpha^2\underbrace{\|\nabla f(x_k)-\nabla f(x^*)\|_2^2}_{\leq L^2\|x_k-x^*\|_2^2} + \|x_k-x^*\|_2^2 - 2\alpha\underbrace{[\nabla f(x_k)-\nabla f(x^*)][x_k-x^*]}_{\geq \rho\|x_k-x^*\|_2^2}$$

$$\leq \underbrace{(\alpha^2L^2+1-2\alpha\rho)}_{\xi}\|x_k-x^*\|_2^2 \leq \xi^{k+1}\|x_0-x^*\|_2^2$$

- Scaled gradient descent algorithm:

$$x_{k+1} = x_k - \alpha \nabla f(x_k)$$

converges if the gradient of f is Lipschitz continuous with parameter L and $\alpha < 2/L$. The algorithm can be made faster by properly scaling the gradient by a positive definite matrix $P \succ 0$:

$$x_{k+1} = x_k - \alpha P \nabla f(x_k)$$

This algorithm converges if the gradient of f is Lipschitz continuous and $\alpha < \frac{2}{L\lambda_{\max}}$, where $\lambda_{\max}$ is the maximum eigenvalue of f.

To see this, note that the this is equivalent to a change of variables:

$y = \sqrt{P}^{-1} x \iff x = \sqrt{P} y$. Let $g(y) = f(\sqrt{P} y)$

$\min_y g(y)$ is a convex problem. $\nabla g(y) = \sqrt{P} \cdot \nabla f(\sqrt{P} y)$

Note that $\|\nabla g(y) - \nabla g(z)\|_2^2 = \left(\nabla f(\sqrt{P}y) - \nabla f(\sqrt{P}z)\right)^T P \left(\nabla f(\sqrt{P}y) - \nabla f(\sqrt{P}z)\right)$

$\leq \lambda_{\max} \|\nabla f(\sqrt{P}y) - \nabla f(\sqrt{P}z)\|_2^2 \leq \lambda_{\max} L^2 \|\sqrt{P}y - \sqrt{P}z\|_2^2$

$\leq \lambda_{\max}^2 L^2 \|y - z\|_2^2$

$\Rightarrow$ g has Lipschitz continous gradient with param. $\lambda_{\max} L$

$\Rightarrow$ GD converges with $\alpha < \frac{2}{\lambda_{\max} L}$ and updates

$y_{n+1} = y_n - \alpha \nabla g(y_n) = y_n - \alpha \sqrt{P} \nabla f(\sqrt{P} y_n)$

$\iff x_{n+1} = x_n - \alpha P \nabla f(x_n)$

- Example

$$\min_{x_1,x_2} \frac{1}{2}(x_1^2 + \rho x_2^2)$$

where $\rho \gg 1$

Standard GD

$$\nabla f(x) = \begin{pmatrix} x_1 \\ \rho x_2 \end{pmatrix} \Longrightarrow x_{n+1} = x_n - \alpha \begin{pmatrix} 1 & 0 \\ 0 & \rho \end{pmatrix} x_n$$

$$= \begin{pmatrix} 1-\alpha & 0 \\ 0 & 1-\rho\alpha \end{pmatrix} x_n = \begin{pmatrix} (1-\alpha)^{n+1} & 0 \\ 0 & (1-\rho\alpha)^{n+1} \end{pmatrix} x_0$$

$\Longrightarrow$ converges to 0 for $\alpha < \frac{2}{\rho}$ with linear rate $\min\{|1-\alpha|; |1-\rho\alpha|\}$

The best rate is obtained by minimizing over $\alpha < \frac{2}{\rho} \Longrightarrow \alpha = \frac{2}{\rho+1}$

In this case $x_{k+1} = \left(\frac{\rho-1}{\rho+1}\right)^{k+1} \begin{pmatrix} 1 & 0 \\ 0 & -1^{k+1} \end{pmatrix} x_0$ can be very slow when $\rho \gg 1$

Scaled GD: let $P = \begin{pmatrix} 1 & 0 \\ 0 & \rho^{-1} \end{pmatrix}$

$$\Longrightarrow x_{n+1} = x_n - \alpha P \begin{pmatrix} 1 & 0 \\ 0 & \rho \end{pmatrix} x_n = (1-\alpha) x_n = (1-\alpha)^{k+1} x_0$$

converges to 0 for $\alpha < 2$

Converges in a single iteration if $\alpha = 1$

- These algorithms can be generalized as follows:

$$x_{k+1} = x_k + d_k$$

where d_k is a descent direction:

$$\nabla f(x_k)^T d_k \leq -\epsilon \|\nabla f(x_k)\|_2^2, \ \epsilon > 0$$

Further, assume that

$$\|d_k\|_2 \leq M \|\nabla f(x_k)\|_2$$

Then, we can prove the following:

Theorem 7. *Assume d_k is a descent direction and $\epsilon > \frac{LM^2}{2}$. Assume f is bounded below. Then, if $\epsilon > \frac{LM^2}{2}$, $\nabla f(x_k) \to 0$ for $k \to \infty$.*

To see this,

$$f(x_{k+1}) = f(x_k + d_k) \leq f(x_k) + \nabla f(x_k)^T d_k + \frac{L}{2}\|d_k\|_2^2 \leq f(x_k) - \left(\epsilon - \frac{LM^2}{2}\right)\|\nabla f(x_k)\|_2^2$$

hence

$$f^* \leq f(x_{n+1}) \leq f(x_0) - \left(\epsilon - \frac{LM^2}{2}\right)\sum_{k=0}^{n}\|\nabla f(x_k)\|_2^2$$

hence we must have $\|\nabla f(x_k)\|_2 \to 0$ for $k \to \infty$.

Examples of descent directions:

– $d_k = -\alpha \nabla f(x_k)$ (standard gradient descent algorithm)

choose $\varepsilon = M = \alpha$ to get $\|d_k\|_2 = \alpha \|\nabla f(x_k)\|_2$; $\nabla f(x_k)^T d_k = -\alpha \|\nabla f(x_k)\|_2^2$

Converges if $\varepsilon > \frac{LM^2}{2} \iff \alpha > \frac{L\alpha^2}{2} \iff \alpha < \frac{2}{L}$

– $d_k = -\alpha P \nabla f(x_k), \ P \succ 0$ (scaled gradient descent algorithm)

Choose $\varepsilon = \lambda_{max}\alpha$ to get $\|d_k\|_2 \leq \alpha\lambda_{max}\|\nabla f(x_k)\|_2$; $\nabla f(x_k)^T d_k = -\alpha \nabla f(x_k)^T P \nabla f(x_k)$
$M = \lambda_{max}\alpha$

$\leq -\alpha\lambda_{max}\|\nabla f(x_k)\|_2^2$

Converges if : $\varepsilon > \frac{LM^2}{2} \iff \lambda_{max}\alpha > \frac{L\lambda_{max}^2\alpha^2}{2} \iff \alpha < \frac{2}{L\lambda_{max}}$

– Assume a strongly convex function $H(x) \succ \rho I,\ \forall x$, such that $H(x) \prec \lambda_{\max} I$. $d_k = -\alpha H(x)^{-1} \nabla f(x_k)$ (Newton algorithm)

Choose $\varepsilon = \frac{\alpha}{\lambda_{max}}$, $M = \frac{\alpha}{\rho} \Rightarrow \|d_k\|_2 = \alpha \sqrt{\nabla f(x_k)^T H(x)^{-2} \nabla f(x_k)} \leq \frac{\alpha}{\rho} \|\nabla f(x_k)\|_2$

$\nabla f(x_k)^T d_k = -\alpha \nabla f(x_k)^T H(x)^{-1} \nabla f(x_k) \leq -\frac{\alpha}{\lambda_{max}} \|\nabla f(x_k)\|_2^2$

Converges if $\varepsilon > \frac{LM^2}{2} \Longleftrightarrow \frac{\alpha}{\lambda_{max}} > \frac{L}{2} \frac{\alpha^2}{\rho^2} \Longleftrightarrow \alpha < \frac{2\rho^2}{L \lambda_{max}}$

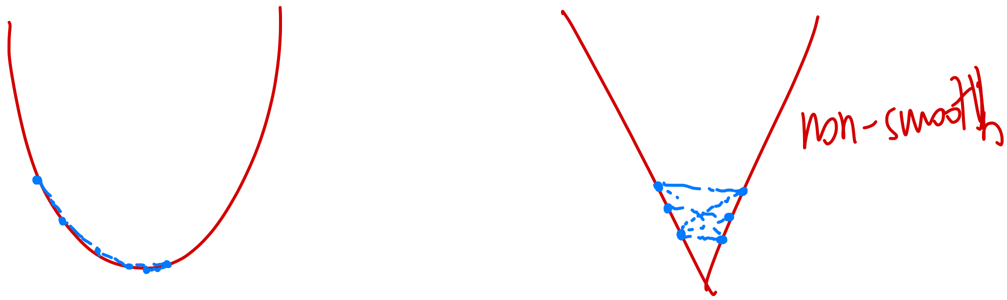
Note: Newton direction is the one that minimizes a second order Taylor approximation of the objective function

$$f(y) \simeq f(x_k) + \nabla f(x_k)^T (y - x_k) + \frac{1}{2}(y - x_k)^T H(x_k)(y - x_k)$$

minimized at

$$y^* - x_k = -H(x_k)^{-1} \nabla f(x_k)$$

- These proofs require the function to be smooth (Lipschitz continuous gradient)



What if this condition is not satisfied? We need to use sub-gradients. In this case, the standard gradient descent does not converge to the optimal point, but may keep oscillating:

Theorem 8. *Assume f is convex and its subgradients are bounded, $\|\nabla f(x)\|_2 \leq M$. Consider the subgradient descent algorithm*

$$x_{k+1} = x_k - \alpha \nabla f(x_k),$$

where $\nabla f(x)$ is a subgradient of f at x. Then, for any $\epsilon > 0$ and $\alpha < \cancel{\epsilon/M}$ $2\epsilon/M^2$, ~~there exists a time $K_{\epsilon,\alpha} < \infty$ such that~~ $\forall k \geq 0,\ \exists\, n \geq k$ s.t. $f(x_n) < f(x^*) + \epsilon$

$$\cancel{f(x_k) \leq f(x^*) + \epsilon, \ \forall k \geq K_{\epsilon,\alpha}},$$

i.e. x_k converges to an $\epsilon-$suboptimal point.

Let $\Omega_\delta = \{x : f(x) \leq f(x^*) + \delta\}$. Clearly, $x^* \in \Omega_\delta, \forall \delta \geq 0$

Let $d_n = \min_{x \in \Omega_0} \|x - x_n\|_2$ → distance from optimal set

$\operatorname{argmin}_{x \in \Omega_0} \|x - x_n\|_2$

We have: $d_{n+1}^2 \leq \|x^* - x_{n+1}\|_2^2 = \|x^* - x_n\|_2^2 + \|x_n - x_{n+1}\|_2^2 + 2[x^* - x_n]^T (x_n - x_{n+1})$

$$= d_n^2 + \alpha^2 \|\nabla f(x_n)\|_2^2 + 2\alpha (x^* - x_n)^T \nabla f(x_n)$$

From convexity: $\nabla f(x_n)^T (x^* - x_n) \leq f(x^*) - f(x_n)$

$$\Rightarrow d_{n+1}^2 \leq d_n^2 + \alpha^2 M^2 + 2\alpha \underline{\left(f(x^*) - f(x_n)\right)}$$

Now, consider different cases:

1) $x_k \notin \Omega_\varepsilon$

$$\Rightarrow f(x_k) > f(x^*) + \varepsilon \Rightarrow d_{k+1}^2 \leq d_k^2 - \underbrace{\left(2\alpha\varepsilon - \alpha^2 M^2\right)}_{>0 \text{ since } \alpha < \frac{2\varepsilon}{M^2}}$$

$\Rightarrow$ Eventually, $x_n \in \Omega_\varepsilon$, for some $n > k$

2) $x_k \in \Omega_\varepsilon \Rightarrow$ true with $n = k$

To guarantee convergence to the optimal point, we need to use a diminishing step-size.

Theorem 9. *Assume f is convex and its subgradients are bounded, $\|\nabla f(x)\|_2 \leq M$. Consider the subgradient descent algorithm*

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k),$$

where $\nabla f(x)$ is a subgradient of f at x. Then, if

$$\sum_k \alpha_k = \infty, \ \sum_k \alpha_k^2 < \infty,$$

then $x_k \to x^*$, where x^* has a sub-gradient $\nabla f(x^*) = 0$

Same as before, but with α_k instead of α:

$$d_{k+1}^2 \leq d_k^2 + \alpha_k^2 M^2 + 2\alpha_k \underbrace{\left(f(x^*) - f(x_k)\right)}_{\leq 0 \ \forall k}$$

$\Longrightarrow$ sum $\sum_{k=0}^{n-1}$ to get: $0 \leq d_n^2 \leq d_0^2 + \underbrace{\sum_{k=0}^{n-1} \alpha_k^2 M^2}_{< W < \infty} - 2\sum_{k=0}^{n-1} \alpha_k \left(f(x_k) - p^*\right)$

$$\Longrightarrow 0 \leq d_n^2 \leq d_0^2 + W - 2\sum_{k=0}^{n-1} \alpha_k \left(f(x_k) - p^*\right)$$

$\Longrightarrow \sum_{k=0}^{n-1} \alpha_k \left(f(x_k) - p^*\right) \leq \frac{d_0^2 + W}{2} \Longrightarrow$ convergent series

If $f(x_k)$ converges to something different from p^*

$\Longrightarrow$ the series diverges $\to$ contradiction

$\Longrightarrow f(x_k) \to p^*$ as $k \to \infty$

III. Constrained optimization algorithms

- Solve $\min f(x),$ s.t. $x \in \mathcal{F}$, f convex, $\mathcal{F}$ is convex

  Optimality condition is

$$f'(x^*; x - x^*) \geq 0, \ \forall x \in \mathcal{F}$$

  where $x^* \in \mathcal{F}$

  When f is differentiable, the optimality condition becomes

$$\nabla f(x^*)^T (x - x^*) = 0, \ \forall x \in \mathcal{F}$$

- However, the normal gradient descent algorithm does not work any more because the new x_{k+1} might fall outside of $\mathcal{F}$
- Three solutions to this problem:

1) Associate a penalty to constraint violation: choose convex $g(x)$ such that

$$g(x) = 0, \ x \in \mathcal{F}$$

$$g(x) > 0, \ x \notin \mathcal{F}$$

   and solve the unconstrained problem

$$\min f(x) + \beta g(x)$$

   The solution will approach the original constrained problem as $\beta \to \infty$

2) Interior point method: choose $g(x)$ such that $g(x) \to \infty$ as x approaches the boundary of $\mathcal{F}$ from inside; then, minimize

$$\min f(x) + \beta g(x)$$

   as before; due to the barrier, the optimal solution is in the interior of $\mathcal{F}$; as $\beta \to 0$, the optimal solution tends to the solution of the unconstrained problem

3) Projection method: after each update, project x_{k+1} back to its feasible set:

$$[x_{k+1}]^+ = \arg\min_{x \in \mathcal{F}} \|x - x_{k+1}\|_2$$

In the first two cases, the problem is converted to an unconstrained problem; we can then use gradient based algorithms; however, it may be difficult to ensure the Lipschitz continuity of the gradient.

IV. Projection and gradient projection algorithm

- Define the projection

$$[x]^+ = \arg\min_{y \in \mathcal{F}} \|y - x\|_2$$

Example: $\mathcal{F} \equiv \otimes_i [a_i, b_i]$ (projection onto a box)

$$\min_{y \in \mathcal{F}} \|y - x\|_2^2 = \min_{y \in \mathcal{F}} \sum_i (y_i - x_i)^2$$

$$= \sum_i \min_{y_i \in [a_i, b_i]} (y_i - x_i)^2 = \begin{cases} x_i & \text{if } x_i \in [a_i, b_i] \\ b_i & \text{if } x_i > b_i \\ a_i & \text{if } x_i < a_i \end{cases}$$

- Projection theorem (Bertsekas&Tsitsiklis,P.211)

Theorem 10.

1) *$\forall x$, $\exists$ a unique $z \in \mathcal{F}$ that minimizes $\|y - x\|_2$ over all $y \in \mathcal{F}$; hence, $[x]^+$ is uniquely defined.*
2) *$z = [x]^+$ if and only if $(y - z)^T(x - z) \leq 0, \ \forall y \in \mathcal{F}$*
3) *The mapping $p(x) = [x]^+$ is continuous and non-expansive, i.e.*

$$\|p(x) - p(y)\|_2 \leq \|x - y\|_2, \ \forall x, y \in \mathbb{R}^n$$

1) Assume $\mathcal{F}$ is convex (not true if $\mathcal{F}$ is not convex)

$\min_{y \in \mathcal{F}} \frac{1}{2}\|y - x\|_2^2$ is a convex optimization problem

Optimality conditions are:

$\nabla f(y^*)^T(y - y^*) \geq 0 \ \forall y \in \mathcal{F}$

$\Longleftrightarrow (y^* - x)^T(y - y^*) \geq 0 \ \forall y \in \mathcal{F}$ (this proves (2) as well)

Assume this condition is verified under two points, y_1^*, y_2^* with $y_1^* \neq y_2^*, \in \mathcal{F}$

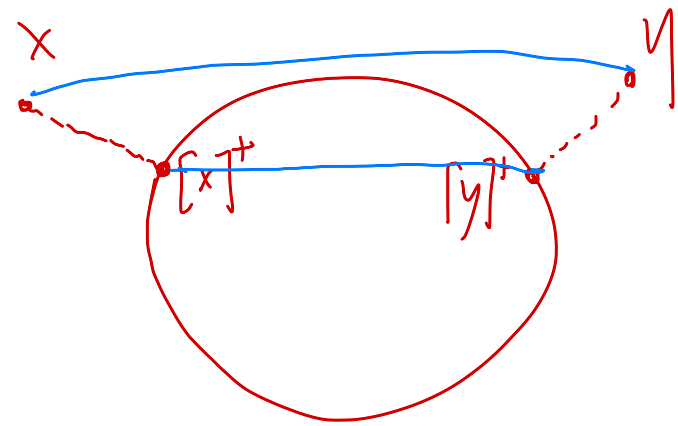
$\Longleftrightarrow (y_i^* - x)^T(y - y_i^*) \geq 0 \ \forall y \in \mathcal{F}, \forall i \in \{1, 2\}$

$\Longrightarrow (y_1^* - x)^T(y_2^* - y_1^*) \geq 0$ and $(y_2^* - x^T)(y_1^* - y_2^*) \geq 0$

$\Longrightarrow$ (sum them) $(y_1^* - y_2^*)(y_2^* - y_1^*) \geq 0 \Longleftrightarrow -\|y_2^* - y_1^*\|_2^2 \geq 0$

$\Longleftrightarrow y_1^* = y_2^* \Longrightarrow$ contradiction!

3) $\|p(x)-p(y)\|_2 \le \|x-y\|_2$



From part (b): $(u-p(x))^T(x-p(x)) \le 0 \quad \forall u \in F$

$\Rightarrow (p(y)-p(x))^T(x-p(x)) \le 0$

Similarly: $(p(x)-p(y))^T(y-p(y)) \le 0$

$\Rightarrow$ sum to get:

$$(p(y)-p(x))^T(p(y)-p(x)+x-y) \le 0$$

$$\Leftrightarrow \|p(y)-p(x)\|_2^2 \le (p(y)-p(x))^T(y-x)$$

$$\le \|p(y)-p(x)\|_2 \|y-x\|_2$$

(continuity also follows)

- <u>Gradient projection algorithm</u>

$$x_{k+1} = [x_k - \alpha \nabla f(x_k)]^+$$

Lemma 11. *Assume f is convex and differentiable. Then $x^* = \arg\min_{x \in \mathcal{F}} f(x)$ if and only if*

$$x^* = [x^* - \alpha \nabla f(x^*)]^+,$$

i.e. x^* is a fixed point of the gradient projection algorithm.

$x^* = \arg\min_{x \in \mathcal{F}} f(x) \Longleftrightarrow$ (optimality conditions):

$$\nabla f(x^*)^T (x - x^*) \geq 0 \quad \forall x \in \mathcal{F}$$

1) Assume $x^* = [x^* - \alpha \nabla f(x^*)]^+ = P(x^* - \alpha \nabla f(x^*))$

$\Longrightarrow$ From optimality of projection:

$$\left(x - P(x^* - \alpha \nabla f(x^*))\right)^T \left(x^* - \alpha \nabla f(x^*) - P\left(x^* - \alpha \nabla f(x^*)\right)\right) \leq 0 \quad \forall x \in \mathcal{F}$$

$\Longleftrightarrow (x - x^*)^T (-\alpha \nabla f(x^*)) \leq 0 \quad \forall x \in \mathcal{F} \Longleftrightarrow \nabla f(x^*)^T (x - x^*) \geq 0 \quad \forall x \in \mathcal{F}$

$\Longrightarrow x^*$ optimal

2) Assume $x^* \neq [x^* - \alpha \nabla f(x^*)]^+ \Longrightarrow \exists\, x \in \mathcal{F}: (x - x^*)(x^* - \alpha \nabla f(x^*) - x^*) > 0$

$\Longleftrightarrow \alpha \nabla f(x^*)^T (x - x^*) < 0 \Longrightarrow x^*$ suboptimal

$\Longrightarrow$ Solving unconstrained problem equivalent to finding fixed point of $x^* = [x^* - \alpha \nabla f(x^*)]^+$

Theorem 12. *If f is convex, with Lipschitz continuous gradient with parameter L, there exists some x^* such that $x^* = [x^* - \alpha\nabla f(x^*)]^+$, and $0 < \alpha < 2/L$, then x_k converges and its limit minimizes $f(x)$ over $\mathcal{F}$.*

From projection we have

$$(y - x_{n+1})^T (x_n - \gamma\nabla f(x_n) - x_{n+1}) \leq 0 \quad \forall y \in \mathcal{F}$$

in particular for $y = x_n$

$$\gamma (x_{n+1} - x_n)^T \nabla f(x_n) \leq -\|x_{n+1} - x_n\|_2^2$$

$$\Longrightarrow f(x_{n+1}) \leq f(x_n) + \nabla f(x_n)^T (x_{n+1} - x_n) + \frac{L}{2}\|x_{n+1} - x_n\|_2^2$$

$$\leq f(x_n) - \underbrace{\left(\frac{1}{\gamma} - \frac{L}{2}\right)}_{>0} \|x_{n+1} - x_n\|_2^2$$

$$\Longrightarrow f(x_n) \leq f(0) - \left(\frac{1}{\gamma} - \frac{L}{2}\right) \sum_{n=0}^{n-1} \|x_{n+1} - x_n\|_2^2 \Longrightarrow \lim \|x_{n+1} - x_n\|_2^2 = 0$$

$\Longrightarrow$ converges to a limit point such that $x^* = [x^* - \alpha\nabla f(x^*)]^+$

If further f is strongly convex, we have the following linear convergence result

Theorem 13. *If f is strongly convex with Lipschitz continuous gradient with parameter L,*

$$L\|x-y\|_2^2 \geq [\nabla f(x) - \nabla f(y)]^T(x-y) \geq \rho\|x-y\|_2^2, \ \forall x, y \in \mathbb{R}^n$$

for some $\rho > 0$ (note that we must have $\rho \leq L$), $x^* = \arg\min_{x\in\mathcal{F}} f(x)$ (unique since f is strongly convex), and $0 < \alpha < \frac{2\rho}{L^2}$, then x_k converges to x^* with linear rate. In particular,

$$\|x_k - x^*\| \leq \xi^k \|x_0 - x^*\|$$

where $\xi = \sqrt{1+\alpha^2L^2 - 2\alpha\rho} \in (0,1)$.

$$\|x_{k+1} - x^*\|_2^2 = \|P(x_k - \alpha\nabla f(x_k)) - P(x^* - \alpha\nabla f(x^*))\|_2^2$$

$$\leq \|x_k - x^* - \alpha(\nabla f(x_k) - \nabla f(x^*))\|_2^2$$

$$= \|x_k - x^*\|_2^2 + \alpha^2 \overbrace{\|\nabla f(x_k) - \nabla f(x^*)\|_2^2}^{\leq L^2\|x_k - x^*\|_2^2}$$

$$\underbrace{-2\alpha(x_k - x^*)^T(\nabla f(x_k) - \nabla f(x^*))}_{\geq \rho\|x-y\|_2^2}$$

$$\leq \|x_k - x^*\|_2^2 (1 + \alpha^2L^2 - 2\alpha\rho) \leq (1 - \alpha^2L^2 - 2\alpha\rho)^{k+1} \|x_0 - x^*\|_2^2$$

- Scaled gradient projection algorithm: similar to the unconstrained case, we can define the scaled version of the algorithm

$$x_{k+1} = [x_k - \alpha P \nabla f(x_k)]^+ \quad , P > 0$$

However, in this case, we need to take special case at the projection operation. To see this, treat the scaled algorithm as a change of variables:

$y = \sqrt{P}^{-1} x \Leftrightarrow x = \sqrt{P} y$. Let $g(y) = f(\sqrt{P} y)$, $Y = \{\sqrt{P}^{-1} x, x \in F\}$

$\Rightarrow \min_{y \in Y} g(y)$ is convex optimization problem.

$$\nabla g(y) = \sqrt{P} \cdot \nabla f(\sqrt{P} y)$$

$$\Rightarrow \quad y_{k+1} = [y_k - \alpha \nabla g(y_k)]^+ = \arg\min_{y \in Y} \| y_k - \alpha \nabla g(y_k) - y \|_2^2$$

$$= \arg\min_{y \in Y} \| \sqrt{P}^{-1} x_k - \alpha \sqrt{P} \nabla f(x_k) - y \|_2^2$$

$$= \arg\min_y \left(x_k - \alpha P \nabla f(x_k) - \sqrt{P} y \right)^T P^{-1} \left(x_k - \alpha P \nabla f(x_k) - \sqrt{P} y \right)$$

$$= \arg\min_y \| x_k - \alpha P \nabla f(x_k) - \sqrt{P} y \|_{P^{-1}}^2$$

$\Rightarrow$ Going back to original domain $x = \sqrt{P} y$

$$x_{k+1} = \sqrt{P} y_{k+1} = \arg\min_{x \in X} \| x_k - \alpha \nabla f(x_k) - x \|_{P^{-1}}^2$$

(P may also be a function of time t, see Bertsekas)

- Projection in the dual

  - In general, the projection operation can be difficult to carry out if the constraints set is in a complex form.

  - However, projection is easy in the dual domain, since the constraint set is always a quadrant. In addition, the subgradient has a simple form.

- Primal problem:

$$\min f_0(x)$$
$$\text{s.t. } f_i(x) \leq 0, \ \forall i$$
$$Ax = b$$

Lagrangian:

$$L(x, \lambda, \nu) = f_0(x) + \sum_i \lambda_i f_i(x) + \nu^T(Ax - b), \ \lambda \geq 0$$

Dual function

$$g(\lambda, \nu) = \min_x L(x, \lambda, \nu)$$

the minimization of the Lagrangian is unconstrained, hence it can be accomplished using a standard unconstrained gradient descent algorithm.

Dual problem

$$\max g(\lambda, \nu)$$
$$\text{s.t. } \lambda \geq 0$$

This can be solved using the gradient projection algorithm.

The subgradient of g at $(\lambda^{(k)}, \nu^{(k)})$ is given by

$$\nabla g(\lambda^{(k)}, \nu^{(k)}) = [f_1(x^{(k)}), \ldots, f_m(x^{(k)}), Ax - b]$$

where

$$x^{(k)} = \arg\min_x L(x, \lambda^{(k)}, \nu^(k))$$

To show that this is indeed a subgradient, need to show that

$$g(\lambda, \nu) \leq g(\lambda^{(k)}, \nu^{(k)}) + \nabla g(\lambda^{(k)}, \nu^{(k)})^T([\lambda; \nu] - [\lambda^{(k)}; \nu^{(k)}]), \ \forall \lambda \geq 0, \forall \nu$$

(note that g is concave)

In fact we have

$$
\begin{aligned}
& g(\lambda^{(k)}, \nu^{(k)}) + \nabla g(\lambda^{(k)}, \nu^{(k)})^T([\lambda; \nu] - [\lambda^{(k)}; \nu^{(k)}]) \\
&= L(x^{(k)}, \lambda^{(k)}, \nu^{(k)}) + \sum_i f_i(x^{(k)})(\lambda_i - \lambda_i^{(k)}) + (\nu - \nu^{(k)})^T(Ax^{(k)} - b) \\
&= L(x^{(k)}, \lambda, \nu) \geq \min_x L(x, \lambda, \nu) = g(\lambda, \nu).
\end{aligned}
$$

As a result, the gradient projection algorithm for the dual is of the following simple form:

$$\lambda_i^{(k+1)} = [\lambda_i^{(k)} + \alpha_k f_i(x^{(k)})]^+$$

$$\nu^{(k+1)} = \nu^{(k)} + \alpha_k(Ax^{(k)} - b)$$

(possibly, diminishing step-size if not differentiable)

- Example: waterfilling in fading channels

$$\max_{p} \sum_{g} \mathbb{P}(g) \ln(1 + gp(g))$$

$$\text{s.t. } 0 \leq p \leq P_{\max}$$

$$\sum_{g} \mathbb{P}(g) p(g) \leq \bar{P}$$

We found previously that $p(g) = \left(\frac{1}{\lambda} - \frac{1}{g}\right)^{+}$ → projection into $[0, P_{\max}]$

Let $\lambda^{(0)} > 0$ (initialization)

$$\Rightarrow \frac{\partial g(\lambda)}{\partial \lambda} = \sum_{g} \mathbb{P}(g) \left(\frac{1}{\lambda} - \frac{1}{g}\right)^{+} - \bar{P}$$

$$\Rightarrow \lambda^{(n+1)} = \left[\lambda^{(n)} + \alpha \left(\sum_{g} \mathbb{P}(g) \left(\frac{1}{\lambda} - \frac{1}{g}\right)^{+} - \bar{P} \right) \right]^{+}$$

→ projection into $[0, +\infty]$

Interpretation: 1) if power constrain satisfied with equality

$\Rightarrow$ keep $\lambda^{(n+1)} = \lambda^{(n)}$

2) If constraint violated

$\Rightarrow$ increase $\lambda^{(n+1)} > \lambda^{(n)} \Rightarrow p(g)$ decreases

3) If constraint satisfied strictly

$\Rightarrow$ decrease $\lambda^{(n+1)} \leq \lambda^{(n)} \Rightarrow p(g)$ increases

- Example: utility maximization of a single resource

$$\max_x \sum_i U_i(x_i)$$

$$\text{s.t. } x \geq 0$$

$$\sum_i x_i \leq R$$

Let $U_i(x_i) = \frac{x_i^{\alpha_i}}{1-\alpha_i}\beta_i$, $\alpha_i \in (0,1)$, $\beta_i > 0$ (as an example)

$$\Rightarrow L(x,\lambda) = -\sum_i U_i(x_i) + \lambda\left(\sum_i x_i - R\right), \; x \geq 0, \; \lambda \geq 0$$

$$g(\lambda) = \min_{x \geq 0} L(x,\lambda) = \sum_i \min_{x_i \geq 0} -U_i(x_i) + \lambda\left(x_i - \frac{R}{N}\right)$$

$$\Rightarrow \text{Solve } U_i'(x_i) = \lambda \Leftrightarrow \frac{\alpha_i x_i^{\alpha_i - 1}}{1-\alpha_i}\beta_i = \lambda$$

$$\Leftrightarrow x_i^*(\lambda) = \left[\frac{\lambda(1-\alpha_i)\alpha_i}{\beta_i}\right]^{\frac{-1}{1-\alpha_i}} \quad (\text{satisfies } x_i^* > 0)$$

Algorithm:
- Initialize $\lambda^{(0)} > 0$
- Compute $x_i^*(\lambda^{(k)})$

$$\Rightarrow \lambda^{(k+1)} = \Big[\lambda^k + \alpha \underbrace{\Big(\sum_i x_i^*(\lambda^{(k)}) - R\Big)}_{\frac{dg(\lambda)}{d\lambda}}\Big]^+$$

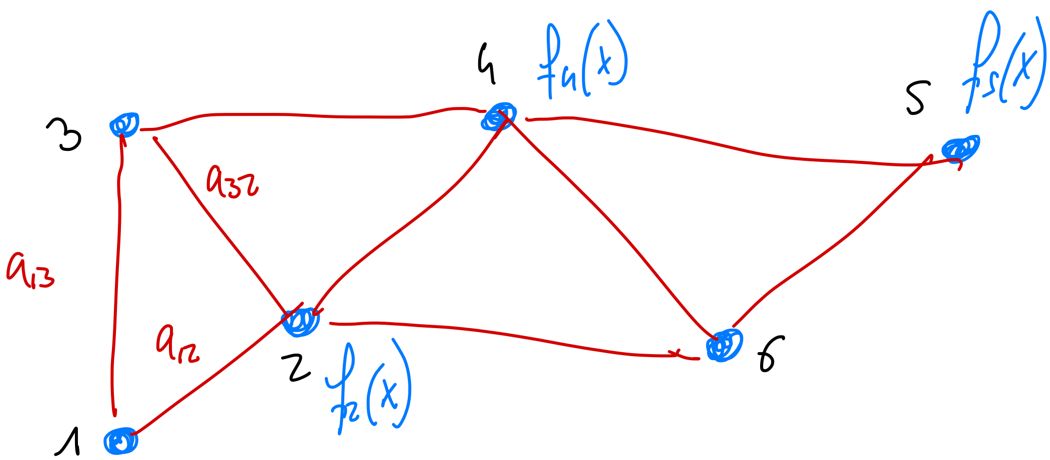
$\nearrow$ projection into $\mathbb{R}_+$

If $\sum_i x_i^*(\lambda^{(k)}) > R \Rightarrow$ increase λ, decrease x_i^*

If $\sum_i x_i^*(\lambda^{(k)}) < R \Rightarrow$ decrease λ, increase x_i^*

- Example: distributed optimization over a network

$$* \quad \min_x \sum_i f_i(x)$$, $f_i : \mathbb{R} \to \mathbb{R}$, f_i concave



- Network as undirected graph with weight matrix $A = [a_{ij}]_{\forall i j}$
- Assume A is stochastic and symmetric: $A \cdot \mathbb{1} = \mathbb{1}$, $A^T = A$, $a_{ij} \geq 0 \ \forall i,j$ and full rank (sufficient to have undirected graph with one communicating class)
- Each node owns a local function $f_i(x)$ but needs to solve cooperatively

$*$

- To this end, each node has a local variable x_i and communicates with the neighbors to solve $*$ in a distributed fashion

$$\min_{x \in \mathbb{R}} \sum_i f_i(x) = \min_{x \in \mathbb{R}^n} \sum_i f_i(x_i) \quad \text{s.t. } x_i = x_j, \forall j \in N_i, \forall i$$

neighbors of i

$$= \min_{x \in \mathbb{R}^n} \sum_i f_i(x_i) \quad \text{s.t. } x_i = \sum_j a_{ij} x_j, \forall i$$

$$= \min_{x \in \mathbb{R}^n} \sum_i f_i(x_i) \quad \text{s.t. } (I - A)x = 0$$

(note that x must be eigenvector of A associated to eigenvalue 1, i.e. $x = \alpha \cdot \mathbb{1}$)

$$\min_{x \in \mathbb{R}^n} \sum_i f_i(x_i) \Rightarrow L(x,v) = \sum_i f_i(x_i) + v^T(I-A)x$$

$$\text{s.t. } (I-A)x = 0$$

$\Rightarrow$ min at x such that: $f_i'(x_i) + [v^T(I-A)]_i = 0$

$$\Leftrightarrow f_i'(x_i) + v_i - \sum_{j \in N_i} v_j a_{ji} = 0 \Rightarrow \text{Determine } x_i^*(v_i; v_j, \forall j \in N_i)$$

$\Rightarrow$ Algorithm: start from $v^{(0)}$ (e.g. $v_i^{(0)} = 0 \ \forall i$)

Node i computes $x_i^*(v_i^{(0)}; v_j^{(0)}, \forall j \in N_i) = x_i^{(0)}$,

communicates its local variable to the neighbors,

and $$\frac{\partial g(v)}{\partial v_i} = [(I-A)x^{(0)}]_i = \underbrace{x_i^{(0)} - \overbrace{\sum_{j \in N_i} a_{ij} x_j^{(0)}}^{\text{communication with neighbors}}}_{\text{consensus disagreement}}$$

and communicates it with the neighbors N_i

Upon receiving $\frac{\partial g(v)}{\partial v_j}$, $\forall j \in N_i$ from the neighbors, node i

computes $v_j^{(1)} = v_j^{(0)} + \alpha \frac{\partial g(v)}{\partial v_j}$ which enables it to

solve $x_i^{(1)}$.

So on, the algorithm proceeds for $k \geq 1$

Interior point methods

$$\min f_0(x)$$
$$\text{s.t. } f_i(x) \le 0 \quad \forall i = 1 \dots m$$
$$Ax = b$$

$\Rightarrow$ log-barrier function

$$\phi_i(x) = -\ln(-f_i(x))$$

$\phi_i(x)$ is convex since $\nabla \phi_i(x) = -\frac{\nabla f_i(x)}{f_i(x)}$

$$\nabla^2 \phi_i(x) = \frac{\nabla f_i(x) \nabla f_i(x)^T}{f_i(x)^2} + \frac{\nabla^2 f_i(x)}{-f_i(x)} \succeq 0$$

$\Rightarrow$ solve instead:

$$\min f_0(x) - \frac{1}{t} \sum_i \ln(-f_i(x)), \quad t > 0$$
$$\text{s.t. } Ax = b$$

with domain: $D = \{ x : x \in D(f_0) ; x \in D(f_i) \; \forall i ; f_i(x) < 0 \}$

D is convex

$\Rightarrow$ convex optimization problem

Minimizer $x^*(t)$ is called central path. As $t \to \infty$, $x^*(t) \to x^*$

$$\min \ f_0(x) - \frac{1}{t}\sum_i \ln(-f_i(x)), \quad t>0$$
$$\text{s.t. } Ax=b$$

KKT conditions are: $\exists\, v$ s.t. $f_i(x)<0 \ \forall i$ and

$$\nabla f_0(x) + \sum_i \frac{-1}{t f_i(x)} \nabla f_i(x) + A^T v = 0, \quad Ax=b$$

Now, let $\lambda_i^*(t) = \frac{-1}{t f_i(x)}$ $\Rightarrow$ we can rewrite KKT as: $(x^*(t), \lambda^*(t), v^*(t))$ solve

$$\nabla f_0(x) + \sum_i \lambda_i \nabla f_i(x) + A^T v = 0$$

$$\lambda > 0$$

$$f_i(x) < 0 \ \forall i, \quad Ax=b$$

$$\lambda_i f_i(x) = -\frac{1}{t}$$ (modified complementary slackness conditions for original problem)

$\Rightarrow$ we get a set of "modified KKT conditions"

Note that, since $x^*(t)$ solves $\nabla f_0(x) + \sum_i \lambda_i^*(t) \nabla f_i(x) + A^T v^*(t) = 0$

$\Rightarrow$ it is primal feasible and it minimizes the Lagrangian of the original problem

$$L(x, \lambda^*(t), \nu^*(t)) = f_0(x) + \sum_i \lambda_i^*(t) f_i(x) + \nu^*(t)^T (Ax - b)$$

$$\Rightarrow f_0(x^*(t)) \geq p^* = \max_{\lambda \geq 0, \nu} g(\lambda, \nu) \geq g(\lambda^*(t), \nu^*(t)) = L(x^*(t), \lambda^*(t), \nu^*(t))$$

$$= f_0(x^*(t)) - \frac{m}{t}$$

$\Rightarrow$ $f(x^*(t)) \leq p^* + \frac{m}{t} \Rightarrow x^*(t)$ is an ε-optimal solution with $\varepsilon = \frac{m}{t}$

$\Rightarrow$ Given the desired accuracy ε, choose $t = \frac{m}{\varepsilon}$ and find $x^*(t)$. Challenges: when t is large (good accuracy) problem becomes ill conditioned:

Hessian of objective function is: $\nabla^2 f_0(x) + \frac{1}{t}\sum_i \left[\frac{\nabla f_i(x) \nabla f_i(x)^T}{f_i(x)^2} + \frac{\nabla^2 f_i(x)}{-f_i(x)}\right]$

$\rightarrow \infty$ when $f_i(x) \simeq 0$

$\Rightarrow$ solution: Barrier method

1) start with $t \sim$ small, solve $x^*(t)$, let $\mu > 1$ and $x = x^*(t)$

2) let $t := \mu t$ and initialize algorithm with x

$\Rightarrow$ solve $x^*(t)$ and let $x = x^*(t)$

Repeat this step until $t \geq \frac{m}{\varepsilon}$

To find $x^*(t)$ you may use any algorithm

- In your descent algorithm, you need to make sure to <u>always</u> satisfy $f_i(x) < 0 \ \forall i$: include an additional step in your line search algorithm to optimise the step-size, and to make sure the step-size is small enough to not violate the feasibility constraint

- <u>Interior point primal-dual method</u>: it is an alternative approach that attempts to solve directly the modified KKT conditions:

$$\nabla f_0(x) + \sum_i \lambda_i \nabla f_i(x) + v^T(Ax-b) = 0$$

$$Ax = b, \quad f_i(x) < 0 \ \forall i$$

$$\lambda_i f_i(x) = -\frac{1}{t}$$

Let $f(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{bmatrix}$ and $Df(x) = \begin{bmatrix} \nabla f_1(x)^T \\ \vdots \\ \nabla f_m(x)^T \end{bmatrix}$

$$r_t(x,\lambda,v) = \begin{pmatrix} \nabla f_0(x) + Df(x)^T \cdot \lambda + A^T v \\ -\operatorname{diag}(f(x))\lambda - \frac{1}{t}\mathbb{1} \\ Ax - b \end{pmatrix} \begin{matrix} r_{dual} \\ r_{central} \\ r_{pri} \end{matrix}$$

Solving the modified KKT conditions amount to solving $r_t(x,\lambda,v) = 0$ with $f_i(x) < 0 \ \forall i$

Let $y = (x, \lambda, \nu)$ with $f_i(x) < 0 \; \forall i$, $\lambda > 0$ be the current point in the algorithm

We want to find $\Delta y = (\Delta x, \Delta \lambda, \Delta \nu)$ such that

$$r_t(y + \Delta y) \approx 0$$

To this end, first order Taylor approx:

$$r_t(y + \Delta y) \approx r_t(y) + \sum_j \frac{\partial r_t(y)}{\partial y_j} \Delta y_j \qquad r_t(y) + Dr_t(y) \Delta y$$

$$\Longrightarrow \quad \Delta y = -Dr_t(y)^{-1} \cdot r_t(y)$$

and the algorithm update becomes

$$y_{k+1} = y_k + \gamma_k Dr_t(y_k)^{-1} r_t(y_k)$$

(Where γ_k is chosen so as to guarantee $f_i(x_{k+1}) < 0 \; \forall i$ and $\lambda_{k+1} > 0$)

In particular: $\left[\dfrac{\partial r_t(y)}{\partial x_j}\right]_j = \begin{bmatrix} \nabla^2 f_0(x) + \sum_i \lambda_i \nabla^2 f_i(x) \\ -\mathrm{diag}(\lambda) Df(x) \\ A \end{bmatrix}$

$$\left[\frac{\partial r_t(y)}{\partial \lambda_j}\right]_j = \begin{bmatrix} Df(x)^T \\ -\mathrm{diag}(f(x)) \\ 0 \end{bmatrix}; \quad \left[\frac{\partial r_t(y)}{\partial \nu_j}\right]_j = \begin{bmatrix} A^T \\ 0 \\ 0 \end{bmatrix}$$

Hence Δy solves:

$$\overbrace{\begin{bmatrix} \nabla^2 f_0(x) + \sum_i \lambda_i \nabla^2 f_i(x) & Df(x)^T & A^T \\ -\mathrm{diag}(\lambda) Df(x) & -\mathrm{diag}(f(x)) & 0 \\ A & 0 & 0 \end{bmatrix}}^{Dr_t(y)} \Delta y$$

$$= - \underbrace{\begin{bmatrix} \nabla f_0(x) + Df(x)^T \cdot \lambda + A^T \nu \\ -\mathrm{diag}(f(x)) \lambda - \frac{1}{t} \mathbb{1} \\ Ax - b \end{bmatrix}}_{r_t(y)}$$

Algorithm outline

Let: surrogate duality gap: $-f(x)^T\lambda$

1) Init. x with $f_i(x) < 0$, $\lambda < 0$, v
2) Set $t = \mu \cdot \frac{m}{\hat{\eta}}$; determine Δy

and $y_{k+1} = y_k + \alpha_k \cdot \Delta y$

3) continue until $\|r_{pri}\|_2 \leq \varepsilon$, $\|r_{dual}\|_2 \leq \varepsilon$, $\hat{\eta} \leq \varepsilon$

We know show that residual decreases for small step size α_k:

$$\lim_{\alpha \to 0} \frac{\|r_t(y + \alpha \Delta y)\|_2^2 - \|r_t(y)\|_2^2}{\alpha} = 2 \left[\underbrace{D r_t(y) \Delta y}_{-r_t(y)} \right]^T \cdot r_t(y)$$

$$= -2 \|r_t(y)\|_2^2 < 0$$

$$\Longrightarrow \frac{d}{d\alpha} \|r_t(y + \alpha \Delta y)\|_2^2 \Big|_{\alpha \to 0} = -2 \|r_t(y)\|_2^2$$

$\Longrightarrow$ $\|r_t(y)\|_2$ strictly decreases for a sufficiently small step size α

(even though Δy might not be a descent direction)