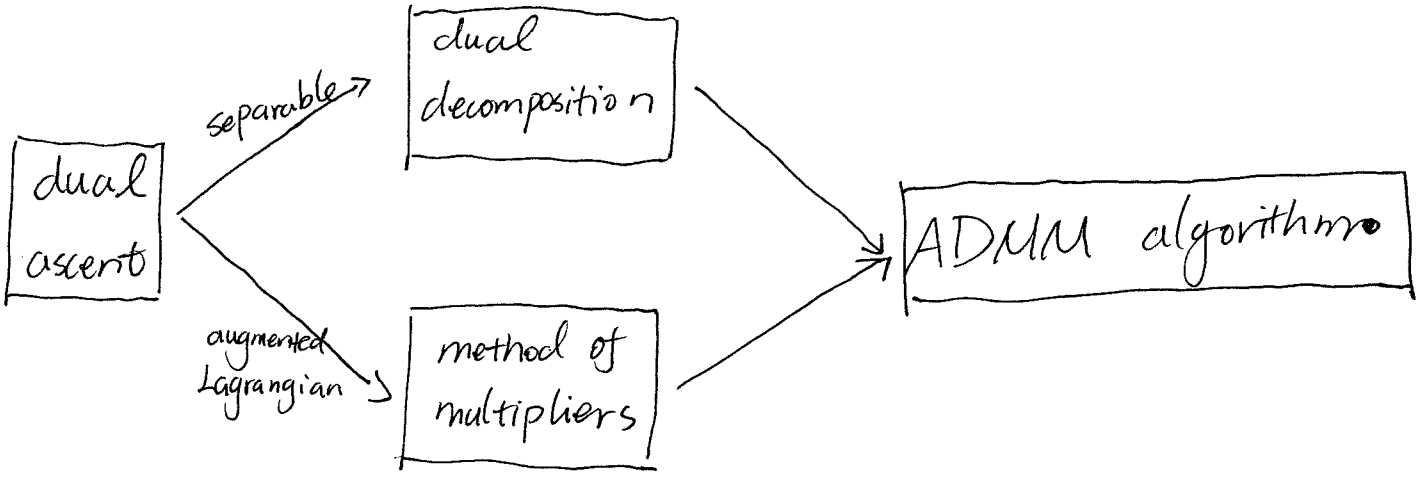
Introduction to Alternating Direction Method of Multipliers

Contents:

More specifically,

ADMM section contains: ADMM algorithm, two assumptions, convergence, and optimality conditions.

The **relationship** between dual ascent, dual decomposition, method of multipliers, and ADMM algorithms is:



Dual Ascent

Equality constrained convex optimization problem

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & Ax - b = 0 \end{aligned}$$

$x \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $f : \mathbb{R}^n \to \mathbb{R}$ is convex.

The Lagrangian is

$$L(x, \nu) = f(x) + \nu^T (Ax - b)$$

The dual function is

$$\begin{aligned} g(\nu) &= \inf_x L(x, \nu) \\ &= -f^*(-A^T\nu) - b^T\nu \end{aligned}$$

The dual problem is

$$\text{maximize} \quad g(\nu)$$

Assume strong duality holds and f is strictly convex,

$$x^* = \operatorname{argmin}_x L(x, \nu^*)$$

where ν^* is a dual optimal point. (see page 23 in first lecture notes.)

Dual ascent iteration consists

$$x^{k+1} = \arg\min_x L(x, v^k)$$

$$v^{k+1} = v^k + \alpha^k (Ax^{k+1} - b)$$

where $\alpha^k > 0$ is a step size.

* How to understand ?

① Given v^k, find x^{k+1} by minimizing $L(x, v^k)$

② Plug x^{k+1} into $L(x, v)$ to get the g function in step k.

$$g_k(v) = f(x^{k+1}) + v^T(Ax^{k+1} - b)$$

Take derivative w.r.t v

$$\nabla g_k(v) = Ax^{k+1} - b$$

③ Adjust v^k in the direction of $Ax^{k+1} - b$ with size α^k, i.e. $v^{k+1} = v^k + \alpha^k (Ax^{k+1} - b)$

④ With appropriate choice of α^k, the dual function increases in each step. i.e.

$$g_{k+1}(y^{k+1}) > g_k(y^k)$$

Separable Problem

A problem is separable if

$$f(x) = \sum_{i=1}^{N} f_i(x_i)$$

Where $X = (x_1 \cdots x_N)^T$, $x_i \in \mathbb{R}^{n_i}$ are subvectors of x.

And the matrix A can be partitioned as

$$A = [A_1 \cdots A_N]$$

so $Ax = \sum_{i=1}^{N} A_i x_i$.

Example.

minimize $x_1 + x_2$.

subject to $\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = b$

so $f(x) = x_1 + x_2 = \sum_{i=1}^{2} f_i(x_i)$, where $f_i(x_i) = x_i$.

$$Ax = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = a_1 x_1 + a_2 x_2 = \sum_{i=1}^{2} a_i x_i$$

Dual Decomposition

If the problem

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && Ax - b = 0 \end{aligned}$$

is separable and $f(x) = \sum_{i=1}^{N} f_i(x_i)$, $Ax = \sum_{i=1}^{N} A_i x_i$,

then the Lagrangian is

$$\begin{aligned} L(x, \nu) &= f(x) + \nu^T (Ax - b) \\ &= \sum_{i=1}^{N} f_i(x_i) + \nu^T \left(\sum_{i=1}^{N} A_i x_i - b \right) \\ &= \sum_{i=1}^{N} \left[f_i(x_i) + \nu^T \left(A_i x_i - \frac{b}{N} \right) \right] \\ &\triangleq \sum_{i=1}^{N} L_i(x_i, \nu) \end{aligned}$$

is separable. For each block i, dual ascent can be used in parallel. i.e.,

$$\begin{aligned} x_1^{k+1} &= \operatorname*{argmin}_{x_1} L_1(x_1, \nu^k) \\ &\vdots \\ x_N^{k+1} &= \operatorname*{argmin}_{x_N} L_N(x_N, \nu^k) \\ \nu^{k+1} &= \nu^k + \alpha^k (A x^{k+1} - b) \end{aligned}$$

* ν-update step is a "gather" step.

Augmented Lagrangians

For optimization ~~prob~~ problem

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && Ax - b = 0 \end{aligned}$$

the augmented Lagrangian is

$$L_\rho(x, \nu) = f(x) + \nu^T(Ax - b) + \frac{\rho}{2}\|Ax - b\|_2^2$$

where $\rho > 0$ is called the penalty parameter

* $L_0(x, \nu)$ is the standard Lagrangian
* The augmented Lagrangian can be viewed as the standard (unaugmented) Lagrangian with problem

$$\begin{aligned} &\text{minimize} && f(x) + \frac{\rho}{2}\|Ax - b\|_2^2 \\ &\text{subject to} && Ax - b = 0 \end{aligned}$$

which is "equivalent" to the original problem.

* Dual function: $g_\rho(\nu) = \inf_x L_\rho(x, \nu)$
* Benefit of adding quadratic penalty term is that $g_\rho(\nu)$ can be shown to be differentiable under rather mild conditions on the original problem.

Method of Multipliers

Apply dual ascent to the modified problem

$$\begin{aligned} \text{minimize} \quad & f(x) + \frac{\rho}{2}\|Ax-b\|_2^2 \\ \text{subject to} \quad & Ax - b = 0 \end{aligned}$$

yields the method of multipliers algorithm iteration:

$$x^{k+1} = \operatorname{argmin}_x L_\rho(x, v^k)$$

$$v^{k+1} = v^k + \rho(Ax^{k+1} - b)$$

* Compare to previous dual ascent iteration, in x-step ~~standard~~ $L(x, v^k)$ is replaced by $L_\rho(x, v^k)$ and in v-step, α^k is replaced by ρ, as step size.
* Method of multipliers converges even if f takes on value $\pm\infty$ or is not strictly convex.

Why use ρ as step size?

Recall the optimality conditions for the original problem are

primal feasibility: $Ax^* - b = 0$

dual feasibility: $\nabla f(x^*) + A^T u^* = 0$

By definition, x^{k+1} minimize $L_\rho(x, u^k)$, so

$$
\begin{aligned}
0 &= \nabla_x L_\rho(x, u^k)\big|_{x=x^{k+1}} \\
&= \nabla_x \Big\{ f(x) + \cancel{(A^T)} u^{kT}(Ax-b) + \frac{\rho}{2}\|Ax-b\|_2^2 \Big\}\Big|_{x=x^{k+1}} \\
&= \Big\{ \nabla_x f(x) + A^T u^k + \rho A^T(Ax-b) \Big\}\Big|_{x=x^{k+1}} \\
&= \nabla_x f(x^{k+1}) + A^T u^k + \rho A^T(Ax^{k+1}-b) \\
&= \nabla_x f(x^{k+1}) + A^T\left[u^k + \rho(Ax^{k+1}-b) \right] \\
&= \nabla_x f(x^{k+1}) + A^T u^{k+1}
\end{aligned}
$$

* By using ρ as step size, the dual feasibility condition always holds.
* As iteration proceeds, the primal residual $Ax^{k+1}-b$ converges to zero, so primal feasibility holds.
* Together guarantees the optimality conditions.

Alternating Direction Method of Multipliers (ADMM)

The ADMM solves problems in the form.

$$\begin{aligned} &\text{minimize} && f(x) + g(z) \\ &\text{subject to} && Ax + Bz = c \end{aligned}$$

$x \in \mathbb{R}^n$, $z \in \mathbb{R}^m$, $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$, $c \in \mathbb{R}^p$. f and g are convex.

The augmented Lagrangian is

$$L_\rho(x, z, v) = f(x) + g(z) + v^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2$$

Method of multipliers iterations

$$(x^{k+1}, z^{k+1}) = \underset{x,z}{\operatorname{argmin}}\, L_\rho(x, z, v^k)$$

$$v^{k+1} = v^k + \rho(Ax^{k+1} + Bz^{k+1} - c)$$

ADMM iterations

$$x^{k+1} = \underset{x}{\operatorname{argmin}}\, L_\rho(x, z^k, v^k)$$

$$z^{k+1} = \underset{z}{\operatorname{argmin}}\, L_\rho(x^{k+1}, z, v^k)$$

$$v^{k+1} = v^k + \rho(Ax^{k+1} + Bz^{k+1} - c)$$

* x and z are updated in an alternating fashion, which accounts for the term "Alternating Direction".

Assumptions

There are two assumptions that guarantee the convergence of the ADMM algorithm.

Assumption 1: The (extended-real-valued) functions $f: \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ and $g: \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ are closed, proper, and convex.

Explanation: using the language of epigraph, the function f satisfies assumption 1 iff its epigraph

$$\text{epi } f = \{(x,t) \in \mathbb{R}^n \times \mathbb{R} \mid f(x) \le t\}$$

is a closed nonempty convex set.

* Assumption 1 implies x and z steps are solvable (not necessary unique), which means $\exists x, z$ to minimize the augmented Lagrangian.

* f and g are not necessary to be differentiable, and the value can be $+\infty$.

Assumption 2 : The unaugmented Lagragian L_0 has a saddle point.

Explanation : It means that there exist (x^*, z^*, v^*), not necessary unique, for which

$$L_0(x^*, z^*, v) \le L_0(x^*, z^*, v^*) \le L_0(x, z, v^*)$$

holds for all x, z, v.

* $L_0(x^*, z^*, v^*) \ge \sup_v L_0(x^*, z^*, v)$

  $L_0(x^*, z^*, v^*) \le \inf_{x,z} L_0(x, z, v^*)$

  and with assumption 1, it follows $L_0(x^*, z^*, v^*)$ is finite for any saddle point (x^*, z^*, v^*).

* (x^*, z^*) is primal optimal, v^* is dual optimal

* strong duality, $p^* = d^*$, holds

* A, B, C can be anything, even not full rank.

Convergence

① Residual convergence

$$r^k = Ax^k + Bz^k - c \longrightarrow 0 \quad \text{as} \quad k \to \infty$$

② Objective convergence

$$f(x^k) + g(z^k) \longrightarrow p^* \quad \text{as} \quad k \to \infty$$

③ Dual variable convergence

$$y^k \longrightarrow y^* \quad \text{as} \quad k \to \infty$$

optimality Conditions

The necessary and sufficient optimality conditions for ADMM problem as

primal feasibility: $Ax^* + Bz^* - c = 0$

dual feasibility: $\begin{cases} 0 \in \partial f(x^*) + A^T v^* \\ 0 \in \partial g(z^*) + B^T v^* \end{cases}$

* ∂ denotes the subdifferential operator. (If f and g are differentiable, $\partial f, \partial g$ can be replaced by $\nabla f, \nabla g$.)
* By definition, z^{k+1} minimizes $L_\rho(x^{k+1}, z, v^{k+1})$. So

$$
\begin{aligned}
0 &\in \partial L_\rho(x^{k+1}, z, v^k)\big|_{z=z^{k+1}} \\
&= \partial \left\{ f(x^{k+1}) + g(z) + v^{kT}(Ax^{k+1} + Bz - c) + \frac{\rho}{2}\|Ax^{k+1} + Bz - c\|_2^2 \right\}_{z=z^{k+1}} \\
&= \left\{ \partial g(z) + B^T v^k + \rho B^T (Ax^{k+1} + Bz - c) \right\}\Big|_{z=z^{k+1}} \\
&= \partial g(z^{k+1}) + B^T v^k + \rho B^T (Ax^{k+1} + Bz^{k+1} - c) \\
&= \partial g(z^{k+1}) + B^T v^k + \rho B^T r^{k+1} \\
&= \partial g(z^{k+1}) + B^T \left[v^k + \rho \cdot r^{k+1} \right] \\
&= \partial g(z^{k+1}) + B^T v^{k+1}
\end{aligned}
$$

which means the second dual feasibility holds

* By definition, x^{k+1} minimizes $L_\rho(x, z^k, u^k)$, so

$$0 \in \partial L_\rho(x, z^k, u^k)\big|_{x=x^{k+1}}$$

$$= \partial\{f(x) + g(z^k) + u^{kT}(Ax + Bz^k - c) + \frac{\rho}{2}\|Ax + Bz^k - c\|_2^2\}\big|_{x=x^{k+1}}$$

$$= \partial f(x^{k+1}) + A^T u^k + \rho A^T(Ax^{k+1} + Bz^k - c)$$

$$= \partial f(x^{k+1}) + A^T u^k + \rho A^T(Ax^{k+1} + Bz^{k+1} - c + Bz^k - Bz^{k+1})$$

$$= \partial f(x^{k+1}) + A^T u^k + \rho A^T(Ax^{k+1} + Bz^{k+1} - c) + \rho A^T B(z^k - z^{k+1})$$

$$= \partial f(x^{k+1}) + A^T[u^k + \rho r^{k+1}] + \rho A^T B(z^k - z^{k+1})$$

$$= \partial f(x^{k+1}) + A^T u^{k+1} + \rho A^T B(z^k - z^{k+1})$$

So

$$\rho A^T B(z^{k+1} - z^k) \in \partial f(x^{k+1}) + A^T u^{k+1}$$

Denote the <u>dual residual</u>.

$$s^{k+1} = \rho A^T B(z^{k+1} - z^k)$$

and it can be shown that $s^{k+1} \to 0$ as $k \to \infty$, which means the first dual feasibility holds.

* $r^{k+1} = Ax^{k+1} + Bz^{k+1} - c$ is called <u>primal residual</u>.

* As ~~int~~ iteration proceeds, the primal residual $Ax^{k+1} + Bz^{k+1} - b \to 0$, so primal feasibility holds.

* Two conditions ~~to gether~~ guarantees that primal and dual optimal can be reached.

Solve Lasso with ADMM

The typical Lasso problem is

$$\text{minimize} \quad \|Ax-b\|_2^2 + \lambda\|x\|_1$$

where $\lambda > 0$ is a regularization parameter. It is an optimization problem without constraint, and most importantly the objective function is separable. Equivalently, the Lasso problem can be written as.

$$\begin{aligned} \text{minimize} \quad & \|Ax-b\|_2^2 + \lambda\|z\|_1 \\ \text{subject to} \quad & x - z = 0 \end{aligned}$$

which fits the form of problem that can be solved by ADMM. Let $f(x) = \|Ax-b\|_2^2$, $g(z) = \lambda\|z\|_1$, $A = B = I$, $c = 0$, we can transform the standard ADMM problem to be Lasso problem.

The augmented Lagrangian is

$$L_\rho(x, z, u) = \|Ax-b\|_2^2 + \lambda\|z\|_1 + u^T(x-z) + \frac{\rho}{2}\|x-z\|_2^2$$

In x-step, $x^{k+1} = \arg\min_x L_\rho(x, z^k, u^k)$. So,

$$\nabla_x L_\rho(x, z^k, u^k) = \nabla_x \left\{ \|Ax-b\|_2^2 + u^T x + \frac{\rho}{2}\|x - z^k\|_2^2 \right\}$$

$$= 2(A^TAx - A^Tb) + u^k + \rho(x - z^k)$$

$$= (2A^TA + \rho I)x - (2A^Tb - u^k + \rho z^k)$$

$$\equiv 0$$

$$\Rightarrow x^{k+1} = (2A^TA + \rho I)^{-1}(2A^Tb - u^k + \rho z^k)$$

* $2A^TA + \rho I$ is always invertible. (positive-definite)

In z-step: $z^{k+1} = \arg\min_z L_\rho(x^{k+1}, z, u^k)$.

$$= \arg\min_z \left\{ \lambda\|z\|_1 - u^T z + \frac{\rho}{2}\|x^{k+1} - z\|_2^2 \right\}$$

$$= \arg\min_z \sum_{i=1}^n \left\{ \lambda|z_i| - u_i^k z_i + \frac{\rho}{2}(x_i^{k+1} - z_i)^2 \right\}$$

For each z_i, $i = 1, \cdots, n$, the subdifferential is

$$\partial\left\{ \lambda|z_i| - u_i^k z_i + \frac{\rho}{2}(x_i^{k+1} - z_i)^2 \right\}$$

$$= \begin{cases} \lambda - u_i^k - \rho(x_i^{k+1} - z_i), & z_i > 0 \\ -\lambda - u_i^k - \rho(x_i^{k+1} - z_i), & z_i < 0 \\ \left[-\lambda - u_i^k - \rho(x_i^{k+1} - z_i),\ \lambda - u_i^k - \rho(x_i^{k+1} - z_i) \right], & z_i = 0 \end{cases}$$

when $z_i > 0$,

$$\lambda - v_i^k - \rho(x_i^{k+1} - z_i) = 0$$

$$\Rightarrow z_i = x_i^{k+1} - \frac{\lambda - v_i^k}{\rho} > 0 \Rightarrow x_i^{k+1} > \frac{\lambda - v_i^k}{\rho}$$

when $z_i < 0$,

$$-\lambda - v_i^k - \rho(x_i^{k+1} - z_i) = 0$$

$$\Rightarrow z_i = x_i^{k+1} + \frac{\lambda + v_i^k}{\rho} < 0 \Rightarrow x_i^{k+1} < -\frac{\lambda + v_i^k}{\rho}$$

when $z_i = 0$,

$$-\lambda - v_i^k - \rho(x_i^{k+1} - z_i) \le 0 \le \lambda - v_i^k - \rho(x_i^{k+1} - z_i)$$

$$\Rightarrow -\lambda - v_i^k - \rho x_i^{k+1} \le 0 \le \lambda - v_i^k - \rho x_i^{k+1}$$

$$\Rightarrow x_i^{k+1} \in \left[-\frac{\lambda + v_i^k}{\rho}, \frac{\lambda - v_i^k}{\rho}\right]$$

So,

$$z_i^{k+1} = \begin{cases} x_i^{k+1} - \frac{\lambda - v_i^k}{\rho}, & x_i^{k+1} > \frac{\lambda - v_i^k}{\rho} \\ x_i^{k+1} + \frac{\lambda + v_i^k}{\rho}, & x_i^{k+1} < -\frac{\lambda + v_i^k}{\rho} \\ 0, & x_i^{k+1} \in \left[-\frac{\lambda + v_i^k}{\rho}, \frac{\lambda - v_i^k}{\rho}\right] \end{cases}$$

The ADMM for Lasso problem is.

$$x^{k+1} = (2A^TA + \rho I)^{-1}(2A^Tb - v^k + \rho z^k)$$

$$z_i^{k+1} = \begin{cases} x_i^{k+1} - \frac{\lambda - v_i^k}{\rho}, & x_i^{k+1} > \frac{\lambda - v_i^k}{\rho} \\ x_i^{k+1} + \frac{\lambda + v_i^k}{\rho}, & x_i^{k+1} < -\frac{\lambda + v_i^k}{\rho} \\ 0, & x_i^{k+1} \in \left[-\frac{\lambda + v_i^k}{\rho}, \frac{\lambda - v_i^k}{\rho}\right] \end{cases}$$

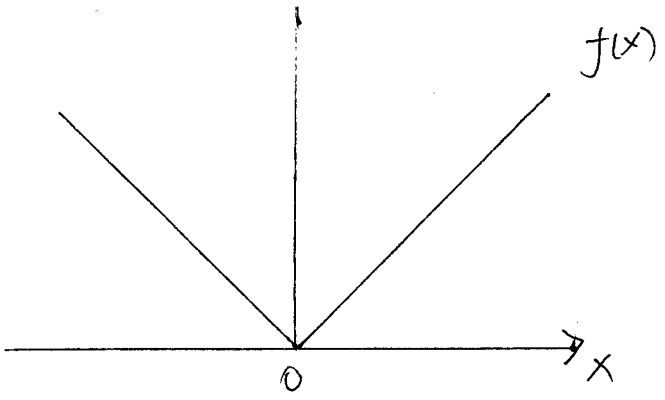
$$v^{k+1} = v^k + \rho(x^{k+1} - z^{k+1})$$

* How fast ?

- dense matrix $A \in R^{400,000 \times 8,000}$ (~30 GB data)
- loading data costs 30 sec
- factorization to 80 subsystems costs 5 min
- ADMM iterations cost 0.5 - 2 sec
- total time cost is 5 ~ 6 min

subdifferential "∂" and minimization

Case 1: $f(x) = |x|$



$$\partial f(x) = \begin{cases} 1 \\ -1 \\ [-1,1] \end{cases}, \quad \begin{matrix} x>0 \\ x<0 \\ x=0 \end{matrix}$$

Case 2: $f(x) = |x| + (x-a)^2$

$$\partial f(x) = \begin{cases} 1+(2x-2a) = 2x-2a+1, & x>0 \\ -1+(2x-2a) = 2x-2a-1, & x<0 \\ [2x-2a-1,\ 2x-2a+1], & x=0 \end{cases}$$

To find $\min f(x)$ is to find x, s.t. $0 \in \partial f(x)$

when $x>0$, $2x-2a+1=0 \Rightarrow x = \frac{2a-1}{2}$

$$\begin{cases} x>0 \\ x=\frac{2a-1}{2} \end{cases} \Rightarrow \frac{2a-1}{2} > 0 \Rightarrow a > \frac{1}{2}$$

when $x<0$, $2x-2a-1=0 \Rightarrow x = \frac{2a+1}{2}$

$$\begin{cases} x<0 \\ x=\frac{2a+1}{2} \end{cases} \Rightarrow \frac{2a+1}{2} < 0 \Rightarrow a < -\frac{1}{2}$$

when $x=0$, $2x-2a-1 \le 0 \le 2x-2a+1 \Rightarrow -2a-1 \le 0 \le -2a+1$

$$\Rightarrow -\frac{1}{2} \le a \le \frac{1}{2}.$$

So,

$$\underset{x}{\operatorname{argmin}} f(x) = \begin{cases} \frac{2a-1}{2}, & a > \frac{1}{2} \\ \frac{2a+1}{2}, & a < -\frac{1}{2} \\ 0, & a \in [-\frac{1}{2}, \frac{1}{2}] \end{cases}$$

* Check "Convex Analysis" by Rockafellar, chapter 23.