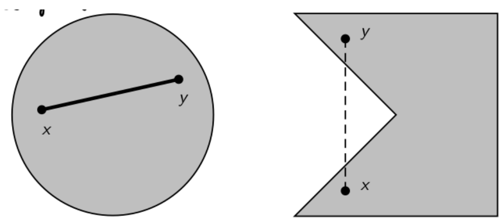
Insiemi convessi

Un insieme è convesso se contiene tutte le convex combination di qualsiasi due punti in C

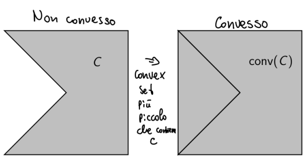
$$\underbrace{\text{convex combination}} \downarrow \quad \alpha x + \beta y \quad \text{con} \quad \alpha+\beta = 1,\ \alpha \geq 0,\ \beta \geq 0$$



convex set   non - convex set

Convex hull

Dato un insieme C, è il più piccolo insieme convesso che contiene C



Operazioni che preservano la complessità

1) Somma e differenza

2) Intersezione

3) Clousure e interior

4) Affine function (funzioni di primo grado), $f: \mathbb{R}^N \to \mathbb{R}^M$ t.c $f(x) = Ax + b$, applicate all'insieme

Oss: l'unione non la preserva

Coni

Un insieme C è un cono se $\alpha x \in C$, $\forall x \in C$ e $\alpha \geq 0$ } stiamo considerando le half lines

1) Non tutti i coni sono convessi → Esempio: Cono composto dagli assi

↑
Cone

2) Non tutti i coni sono poliedroni → Esempio: $\{x_3 \geq \sqrt{x_1^2 + x_2^2},\ x \in \mathbb{R}^3\}$

Remember: $P = \{x : Ax \leq b\}$ è un poliedro e il suo recession cone, cioè l'insieme delle direzioni che dato un qualsiasi punto del poliedro si sposta all'interno di esso, è:

$$\text{rec}(P) := \{d : \ x + \alpha d \in P \quad \text{for any } x \in P,\ \alpha \geq 0\}.$$

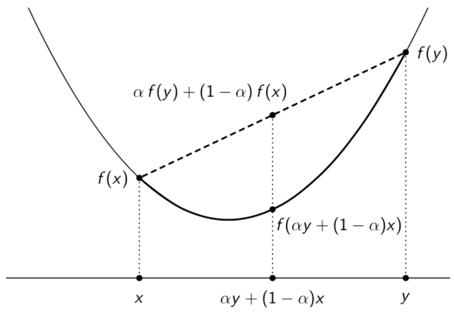
Il recession cone è lui stesso un poliedro.

Funzioni Convesse

Dato un insieme C convesso, una funzione $f: C \to \mathbb{R}$ è convessa se:

$$f(\alpha y + (1-\alpha)x) \leq \alpha f(y) + (1-\alpha) f(x) \qquad \forall\, x, y \in C, x \neq y, \forall\, \alpha \in (0,1)$$

Graficamente:



Dati due punti x, y, la funzione è minore del segmento che congiunge x e y.

Questo deve valere per una qualsiasi coppia x, y della funzione.

OSS: f è concava se $-f$ è convessa.

1) Strictly convex

Dato un insieme C convesso, una funzione $f: C \to \mathbb{R}$ è strictly convex se:

$$f(\alpha y + (1-\alpha)x) < \alpha f(y) + (1-\alpha) f(x) \qquad \forall\, x, y \in C, x \neq y, \forall\, \alpha \in (0,1)$$

↳ non è $\leq$

In questa definizione non vengono incluse, ad esempio, le funzioni lineari.

2) Strongly Convex

Dato un insieme C convesso, una funzione $f: C \to \mathbb{R}$ è strongly convex se:

$$f(\alpha y + (1-\alpha)x) \leq \alpha f(y) + (1-\alpha) f(x) - \frac{\tau}{2}\alpha(1-\alpha)\|y - x\|_2^2 \quad , \quad \forall\, x, y \in C, x \neq y, \forall\, \alpha \in (0,1)$$

Teo: f è strongly convex sse $\exists\, \tau > 0$ t.c. $f(x) - \frac{\tau}{2}\|x\|_2^2$ è convesso

Attenzione:

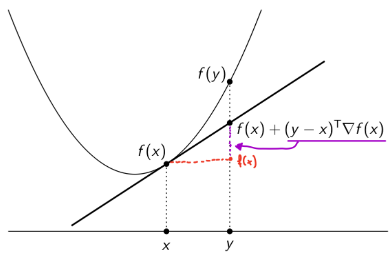
1) Strongly convex $\Rightarrow$ strictly convex $\Rightarrow$ convex
2) Convex $\not\Rightarrow$ strictly convex (Ad esempio per le funz. lineari)
3) Strictly convex $\not\Rightarrow$ strongly convex

Definizione tramite First order Conditions (derivata prima)

Sia C convesso, $f: C \to \mathbb{R}$ differenziabile continua, f è convessa se e solo se:

$$f(y) \geq f(x) + (y-x)^T \nabla f(x) \qquad \forall\, x, y \in C.$$

Graficamente:



la tangente è al di sotto del grafico

1) Strictly Convex : $$f(y) \overset{\text{solo } >}{>} f(x) + (y-x)^T \nabla f(x) \qquad \forall\, x, y \in C, \text{with } x \neq y.$$

2) Strongly Convex

Esiste un $\tau > 0$ t.c $$f(y) \geq f(x) + (y-x)^T \nabla f(x) + \underbrace{\boxed{\frac{\tau}{2}\|y-x\|_2^2}}_{\text{termine positivo}} \qquad \forall\, x, y \in C.$$

Definizione tramite Second order Conditions (derivata seconda)

Sia C convesso, $f: C \to \mathbb{R}$ twice differenziabile continua, f è convessa se e solo se:

$$v^T \nabla^2 f(x) v \geq 0 \qquad \forall\, v \neq 0,$$ o equivalentemente the eigenvalues of $\nabla^2 f(x)$ are ≥ 0.

cioè la Hessian Matrix $\nabla^2 f(x)$ è positiva semidefinita

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 x_n} \\ & & \vdots \\ & & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}$$

1) Strictly Convex: $\nabla^2 f(x)$ è $\underbrace{\text{positiva definita}}_{\text{Autovalori } > 0 \text{, non } \geq}$ $\forall x \in C$ o equivalentemente $$v^T \nabla^2 f(x) v > 0 \qquad \forall\, v \neq 0,$$

2) Strongly convex: Esiste $\tau > 0$ t.c $\nabla^2 f(x) - \tau I$ è positivo semidefinito $\forall x \in C$, cioè:

$$v^T \nabla^2 f(x) v \geq \tau \|v\|_2^2 \qquad \forall\, v \neq 0,$$ o equivalentemente gli autovalori di $\nabla^2 f(x)$ sono $\geq$ di τ

Examples

$f(x) = c^\top x$ is both convex and concave
$f(x) = \frac{1}{2}x^\top Qx + c^\top x$ is

- convex iff Q is positive semidefinite
- strongly convex iff Q is positive definite
- concave iff Q is negative semidefinite
- strongly concave iff Q is negative definite

$f(x) = e^{ax}$ for any $a \in \mathbb{R}$ is strictly convex, but not strongly convex
$f(x) = \log(x)$ is strictly concave, but not strongly concave
$f(x) = x^a$ with $x > 0$ is strictly convex if $a > 1$ or $a < 0$. Is it strongly convex?
$f(x) = x^a$ with $x > 0$ is strictly concave if $0 < a < 1$
$f(x) = \|x\|$ is convex, but not strictly convex
$f(x) = \max\{x_1, \ldots, x_n\}$ is convex, but not strictly convex

Operazioni che preservano la convessità

1) Se f è convessa e $\alpha > 0 \Rightarrow \alpha f$ è convessa

2) Se f_1, f_2 sono convesse $\Rightarrow f_1 + f_2$ è convessa

3) Se f è convessa $\Rightarrow f(Ax+b)$ è convessa

4) Se $f_1, \ldots, f_m$ è convessa $\Rightarrow f(x) = \max\{f_1(x), \ldots, f_m(x)\}$ è convessa

5) Se $\{f_i\}_{i \in I}$ è una famiglia di funzioni convesse $\Rightarrow f(x) = \sup_{i \in I} f_i(x)$ è convessa

6) Siano $f: \mathbb{R}^n \to \mathbb{R}$ e $g: \mathbb{R} \to \mathbb{R}$

- If f is convex and g is convex and nondecreasing, then $g \circ f$ is convex.
- If f is concave and g is convex and nonincreasing, then $g \circ f$ is convex.

- If f is concave and g is concave and nondecreasing, then $g \circ f$ is concave.
- If f is convex and g is concave and nonincreasing, then $g \circ f$ is concave.

$\Rightarrow$

Examples

- If f is convex, then $\overset{g}{e}^{\overbrace{f(x)}^{f}}$ is convex
- If f is concave and positive, then $\overset{g}{\log} \overbrace{f(x)}^{f}$ is concave
- If f is convex, then $-\log(-f(x))$ is convex on $\{x : \; f(x) < 0\}$
- If f is concave and positive, then $\dfrac{1}{f(x)}$ is convex
- If f is convex and nonnegative, then $f(x)^p$ is convex for all $p \geq 1$

7) Dato $f: \mathbb{R}^n \to \mathbb{R}$ e $\alpha \in \mathbb{R}$, l'insieme $S_\alpha(f) = \{x \in \mathbb{R}^n : \; f(x) \leq \alpha\}$ è l'α-sublevel set di f.

Se f è convessa $\Rightarrow S_\alpha(f)$ è convessa $\forall \alpha \in \mathbb{R}$

OSS: Non è vero il contrario, ad esempio x^3 è una funz. non concava con sublevel concavi

Quasi-Convex functions

Dato un convex set $C \subseteq \mathbb{R}^n$, una funzione $f: C \to \mathbb{R}$ è quasi convex se l'α-sublevel sets sono convessi $\forall \alpha \in \mathbb{R}$.

Examples

- $f(x) = \sqrt{|x|}$ is quasiconvex on $\mathbb{R}$
- $f(x_1, x_2) = x_1 x_2$ is quasiconcave on $\{x \in \mathbb{R}^2 : \; x_1 > 0, x_2 > 0\}$
- $f(x) = \log x$ is quasiconvex and quasiconcave
- $f(x) = \operatorname{ceil}(x) = \inf\{z \in \mathbb{Z} : \; z \geq x\}$ is quasiconvex and quasiconcave

Optimization Problem in standard form

$$\begin{cases} \min \ f(x) \\ g(x) \le 0 \\ h(x) = 0 \end{cases}$$

dove:
- $f: \mathbb{R}^n \to \mathbb{R}$ è la funzione obiettivo
- $g(x) = (g_1(x), \dots, g_m(x))$ sono le inequality constraints functions
- $h(x) = (h_1(x), \dots, h_p(x))$ sono le equality constraints functions

Domain (D): Insieme di punti dove tutte le funzioni sono definite

Feasible Region (Ω): Insieme di punti, nel dominio, che rispettano tutti i constraints

OSS: Consideremo solo problemi di minimizzazione, essendo che:

$$\max \{ f(x) \} = -\min \{ f(x) \}$$

obiettivo: Trovare x, che rispetta i vincoli, t.c minimizza la funzione obiettivo

Ottimo Globale e locale

Un valore è ottimo se $v^* = \inf\{f(x) : x \in \Omega\}$

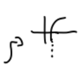
- $v^* \in \mathbb{R}$ se il problema è limitato inferiormente
- $v^* = -\infty$ se il problema non è limitato inferiormente
- $v^* = +\infty$ se il problema è infeasible, cioè $\Omega = \emptyset$

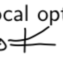
1) Ottimo Globale: È un punto $x^* \in \Omega$ t.c $f(x^*) \le f(x)$ $\forall x \in \Omega$

$\longrightarrow$ $\min\{f(x) : x \in \Omega\}$ denota l'insieme dei minimi globali

2) Ottimo locale: È un punto $x^* \in \Omega$ t.c $f(x^*) \le f(x)$ $\forall x \in \Omega \cap B(x^*, R)$ per qualche $R > 0$

Examples




- $f(x) = \log(x)$, $v^* = -\infty$, no optimal solution
- $f(x) = x^3 - 3x$, $v^* = -\infty$, $x^* = 1$ is a local optimum
- $f(x) = e^x$, $v^* = 0$, no optimal solution 
- $f(x) = x\log(x)$, $v^* = -1/e$, $x^* = 1/e$ is a global optimum
- $f(x) = 3x^4 - 8x^3 - 6x^2 + 24x + 19$, $v^* = 0$, $x^* = -1$ is a global optimum and $\tilde{x} = 2$ is a local optimum

Convex Optimization Problems

Un optimization problem è convesso se:

$$\begin{cases} \min\ f(x) \\ g(x) \le 0 \\ h(x) = 0 \end{cases}$$

→ $f(x)$ convex
→ $g(x)$ convex
→ $h(x)$ affine functions (lineari) ⇒ Vogliamo che siano sia convex che concave

OSS: Se il problema non è convesso, potrebbe essere possibile riformularlo per renderlo tale.

$$\begin{cases} \min\ x_1^2 + x_2^2 \\ x_1/(1+x_2^2) \le 0 \\ \underbrace{(x_1+x_2)^2 = 0}_{\text{Not an affine function}} \end{cases} \text{ is NOT convex,} \qquad \Rightarrow \qquad \begin{cases} \min\ x_1^2 + x_2^2 \\ x_1 \le 0 \\ x_1 + x_2 = 0 \end{cases} \text{ is convex.}$$

Not convex

teorema 1: Nei convex optimization problem, la feasible region (Ω) è un convex set. Posso quindi muovermi da un feasible point a un altro passando da un segmento che li congiunge.

teorema 2: Nei convex optimization problem, ogni ottimo locale è un ottimo globale

Esistenza di un ottimo globale (teorema di Weierstrass)

Se la funzione obiettivo f è continua e Ω è chiusa e limitata, allora esiste un ottimo globale

Corollario 1: Se tutte le funzioni f, g_i, h_j sono continue, il dominio D è chiuso e la feasible region Ω è limitata, allora esiste un ottimo globale.

Corollario 2: Se la funzione obiettivo f è continua, la feasible region Ω è chiusa, ma non per forza limitata, e esiste un $\alpha \in \mathbb{R}$ t.c. l' α-sublevel set

$$S_\alpha(f) = \{x \in \Omega : \ f(x) \le \alpha\}$$

è non empty e limitato, allora esiste un ottimo globale.

Questo perché vogliamo minimizzare f e se troviamo un sublevel limitato, possiamo considerare come feasible region il sub level essendo che non siamo interessati ai punti $> \alpha$ (stiamo minimizzando)

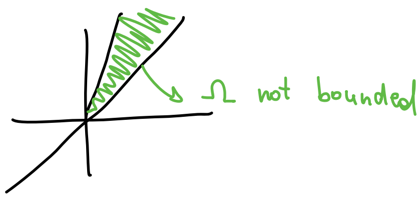
Esempio:

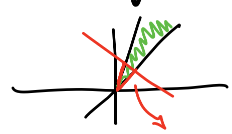
$$\begin{cases} \min \ e^{x_1+x_2} \\ x_1 - x_2 \le 0 \\ -2x_1 + x_2 \le 0 \end{cases} \quad \Rightarrow \quad S_2(f) = \{ x \in \Omega : f(x) \le 2 \}$$

se prendiamo $\alpha = 2$

$$\downarrow$$

$$e^{x_1+x_2} \le 2 \Rightarrow x_1 + x_2 \le \log 2$$





vogliamo minimizzare f sul triangolo il quale è bounded e closed

Corollario 3: Se la funzione obiettivo f è continua e coercive (cioè $\lim_{\|x\| \to \infty} f(x) = +\infty$) e la feasible region Ω è chiusa, allora esiste un ottimo globale.

Questo perché, se f è coercive $\rightarrow$ Qualsiasi sublevel set è limitato

Corollario 4

1) Se f è strongly convex e Ω è chiusa $\Rightarrow$ esiste un ottimo globale
2) Se f è strongly convex e Ω è chiusa e convex $\rightarrow$ l'ottimo globale esiste ed è unico

Esistenza di un ottimo globale nei quadratic problems

$$\begin{cases} \min \ \frac{1}{2} x^T Q x + c^T x \\ Ax \le b \end{cases}$$

, sia il recession cone di Ω $\quad rec(\Omega) = \{ d : Ad \le 0 \}$

Per il teorema di Eaves, esiste un ottimo globale sse le seguenti condizioni tengono:

1) $d^T Q d \ge 0 \quad \forall d \in rec(\Omega)$

2) $d^T(Qx+c) \geq 0 \quad \forall x \in \Omega$ e $\forall d \in rec(\Omega)$ t.c $d^TQd = 0$

Casi speciali:

1) Se $Q=0$ (il problema è lineare) allora il problema ha un ottimo globale sse $d^Tc \geq 0 \quad \forall d \in rec(\Omega)$.

Questo perché:

a) $d^TQd \geq 0$ → sempre vera perché $Q=0$

b) $d^T(Qx+c) \geq 0 = d^Tc \geq 0$ perché $Q=0$ e $d^TQd=0 \ \forall d$

2) Q è semidefinite, a e b sono soddisfatte

a) $d^TQd \geq 0$ → sempre perché $Q>0$

b) $d^T(Qx+c) \geq 0 = 0 \geq 0$ perché se $Q>0 \Rightarrow d^TQd=0$ solo se $d=0$

3) Se Ω è limitata, a e b sono soddisfatte

perché se Ω è limitata, il recession cone è composto solo da $d=0$, quindi

a) $d^TQd \geq 0$ → è soddisf.

b) $d^T(Qx+c) \geq 0$ → è soddisf.

Optimality Conditions basate sul first order (derivata prima)

Unconstrained Problems

Nel caso di problemi senza vincoli $\min\{f(x) : x \in \mathbb{R}^n\}$.

1) Se x è un ottimo locale → $\nabla f(x^*) = 0$

2) Se f è convesso → x^* è un ottimo globale sse $\nabla f(x^*) = 0$

Constrained Problems

Quando analizziamo problemi con vincoli, otteniamo che non è detto che $\nabla f(x) = 0$ nell'ottimo globale, questo perché deve essere presa in considerazione la relazione tra il gradiente e la feasible region.

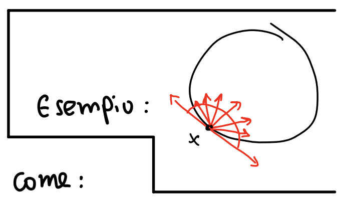
Esempio :

Example.

$$\begin{cases} \min \ x_1 + x_2 \\ x_1^2 + x_2^2 - 4 \le 0 \end{cases}$$

$\Omega = B(0,2)$, global optimum is $x^* = (-\sqrt{2}, -\sqrt{2})$, $\nabla f(x^*) = (1,1)$.

Definiamo quindi il tangent cone.



Tangent Cone

Dato $x \in \Omega$, il tangent cone a Ω in x è definito come:

$$T_\Omega(x) = \left\{ d \in \mathbb{R}^n : \exists \{z_k\} \subset \Omega, \ \exists \{t_k\} > 0, \ z_k \to x, \ t_k \to 0, \ \lim_{k \to \infty} \frac{z_k - x}{t_k} = d \right\}$$

cioè l'insieme di tutte le direzioni, uscenti da x, verso un qualsiasi altro punto della feasible region.

Condizione di Ottimalità

Se x^* è un ottimo locale $\Rightarrow$ $d^T \nabla f(x^*) \ge 0, \quad \forall \, d \in T_\Omega(x^*)$.

questo perché :

1) Se il prodotto fosse negativo, vorrebbe dire che in una delle direzioni che ci fa rimanere nella feasible region Ω, la funzione decresce

2) Se invece il prodotto è ≥ 0 $\forall d$ che ci fa rimanere nella feasible region Ω, allora abbiamo trovato un minimo locale perché la funz. non decresce in nessuna direzione.

Problema : usare il tangent cone nella pratica essendo difficile da trovare.

Proprietà del tangent cone

$T_\Omega(x)$ è relazionato con le proprietà geometriche di Ω.

Qual è la relazione tra $T_\Omega(x)$ e i constrains g e h che definiscono Ω?

First order fasible direction cone

Dato $x \in \Omega$, l'insieme $A(x) = \{ i : g_i(x) = 0 \}$ denota l'insieme di inequality constraints che sono attivi su x.

L'insieme $D(x)$ è chiamato First order fasible direction cone su x

$$D(x) = \left\{ d \in \mathbb{R}^n : \begin{array}{ll} d^T \nabla g_i(x) \leq 0 & \forall\, i \in \mathcal{A}(x), \\ d^T \nabla h_j(x) = 0 & \forall\, j = 1, \dots, p \end{array} \right\}$$

Generalmente $T_\Omega(x) \neq D(x)$, la relazione che abbiamo è la seguente:

TEO: $T_\Omega(x) \subseteq D(x) \quad \forall x \in \Omega$

Abadie Constraints Qualification (ACQ)

Quando $T_\Omega(x) = D(x)$ diciamo che la Abadie Constraints Qualification (ACQ) è vera in x (abbiamo dei buoni vincoli dal punto di vista matematico)

In generale, ACQ non è vera in nessuna $x \in \Omega$, ma ci sono dei casi in cui è vera.

Condizioni sufficenti per ACQ

1) Affine constraints: Se g e h sono affine per tutte le funzioni, allora ACQ è vera $\forall x \in \Omega$

2) Slater Condition: Se g sono convesse e h affini e esiste $\bar{x} \in \text{int}(D)$ t.c $g(\bar{x}) < 0$ e $h(\bar{x}) = 0$, allora ACQ è vera $\forall x \in \Omega$

interno (esclusi gli estremi)

3) Linear indipendence of the gradients of active constraints:

Se $\bar{x} \in \Omega$ e i vettori $\begin{cases} \nabla g_i(\bar{x}) & \text{for } i \in \mathcal{A}(\bar{x}), \\ \nabla h_j(\bar{x}) & \text{for } j = 1, \dots, p \end{cases}$ sono indipendenti, allora ACQ è vera $\forall \bar{x} \in \Omega$

Karush - Kuhn - Tucker Theorem

La proprietà ACQ è importante perché è quella su cui si basa il KKT theorem.
Il KKT è importante perché ci permette di scrivere la ricerca del ottimo locale come un sistema di equality e disequality constraints, invece di usare la definizione di tangent cone.

Teorema

Se x^* è un ottimo locale e ACQ è vera in x^*, allora esiste un $\lambda^* \in \mathbb{R}^m$ e un $\mu^* \in \mathbb{R}^p$ t.c (x^*, λ^*, μ^*) soddisfano il KTT system.
I vettori λ^* e μ^* sono chiamati "moltiplicatori". λ^* per g const. e μ^* per gli h const.

$$\begin{cases} \nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla g_i(x^*) + \sum_{j=1}^{p} \mu_j^* \nabla h_j(x^*) = 0 & \Rightarrow \text{Generalizzazione della condizione sugli stationary points} \\ \lambda_i^* g_i(x^*) = 0 \quad \forall\, i = 1, \ldots, m & \Rightarrow \text{Complementary conditions} \begin{cases} \text{Se il vincolo è attivo } (=0) \Rightarrow \lambda \neq 0 \\ \text{Se il vincolo è inattivo } (\neq 0) \Rightarrow \lambda = 0 \end{cases} \\ \lambda^* \geq 0 & \\ \left.\begin{array}{l} g(x^*) \leq 0 \\ h(x^*) = 0 \end{array}\right\} \text{vincoli} & \end{cases}$$

Quando si cerca l'ottimo locale dobbiamo determinare non solo x^* ma anche λ^* e μ^*.

Attenzione:

1) Possiamo applicare il KTT system sse l'ACQ è vera in x^*
2) KTT da le condizioni necessarie per l'ottimalità, ma non le sufficenti.
   Non è detto che le soluzioni del KTT system siano local optimum.

KKT theorem per problemi convessi

Se l'optimization problem è convesso e (x^*, λ^*, μ^*) sono la soluzione del KTT system, allora x^* è un ottimo globale.
In questo caso il KTT system da sia una condizione sufficente che necessaria.

KKT theorem per problemi non convessi

La soluzione del KKT system è un candidato per essere un ottimo locale ma dobbiamo analizzare la derivata seconda per essere sicuri che lo sia effettivamente

Critical Cone

È un sottoinsieme del first-order feasible direction cone ed è definito, data una soluzione del KTT system (x^*, λ^*, μ^*) come:

First order direction cone

$$C(x^*, \lambda^*, \mu^*) = \left\{ d \in \mathbb{R}^n : \begin{array}{ll} d^T \nabla g_i(x^*) = 0 & \forall\, i \in \mathcal{A}(x^*) \text{ con } \lambda_i^* > 0 \\ d^T \nabla g_i(x^*) \leq 0 & \forall\, i \in \mathcal{A}(x^*) \text{ con } \lambda_i^* = 0 \\ d^T \nabla h_j(x^*) = 0 & \forall\, j = 1, \ldots, p \end{array} \right\}$$ oppure $$C(x^*, \lambda^*, \mu^*) = \{ d \in D(x^*) :\ d^T \nabla f(x^*) = 0 \}$$

Cioè l'insieme delle direzioni dove la funzione f può decrescere. (non sappiamo se la f in quel punto cresce o no)

Second order necessary optimality condition

Sia la Lagrangian function definita come $L(x, \lambda, \mu) := f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=1}^{p} \mu_j h_j(x).$

Definiamo la condizione necessaria:

Assumiamo che (x^*, λ^*, μ^*) risolve il KKT system e i gradienti dei vincoli attivi in x^* siano linearmente indipendenti.

Se x^* è un minimo locale allora

$$L(x, \lambda, \mu) := f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=1}^{p} \mu_j h_j(x).$$

$$d^T \nabla^2_{xx} L(x^*, \lambda^*, \mu^*)\, d \geq 0 \qquad \forall\, d \in C(x^*, \lambda^*, \mu^*),$$

dove $\nabla^2_{xx} L(x^*, \lambda^*, \mu^*)$ è l'hessian Matrix di $L(\cdot, \lambda^*, \mu^*)$ in x^*.

Per avere la condizione sufficiente per l'ottimalità, abbiamo bisogno che:

se (x^*, λ^*, μ^*) risolve il KKT system e $d^T \nabla^2_{xx} L(x^*, \lambda^*, \mu^*)\, d > 0 \quad \forall\, d \in C(x^*, \lambda^*, \mu^*), d \neq 0,$

allora x^* è un ottimo locale. (abbiamo bisogno che la disuguaglianza sia stretta)

Lagrangian duality

$L(x, \lambda, \mu) := f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=1}^{p} \mu_j h_j(x)$

Sia l'optimization problem (P) = $\begin{cases} \min\ f(x) \\ g_i(x) \leq 0 & i = 1, \ldots, m \\ h_j(x) = 0 & j = 1, \ldots, p \end{cases}$ e L la lagrangian function

Dato $\lambda \geq 0$ e $\mu \in \mathbb{R}^p$, il problema $\begin{cases} \min\ L(x, \lambda, \mu) \\ x \in \mathcal{D} \end{cases}$ è chiamato Lagrangian

Relaxation del problema (P) e $\varphi(\lambda, \mu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \mu)$ è la lagrangian dual function.

La dual function φ ha le seguenti caratteristiche:

1) È sempre concava

2) Può essere $-\infty$ in alcuni punti

3) Può non essere differenziabile in alcuni punti

TEO: $\forall \lambda \geq 0$ e $\mu \in \mathbb{R}^p$, abbiamo che $\varphi(\lambda, \mu) \leq v(P)$

$\hookrightarrow$ la dual function ci da un lower bound dell'optimal value del problema

Lagrangian Dual Problem

Il problema (D) $\begin{cases} \max \varphi(\lambda, \mu) \\ \lambda \geq 0 \end{cases}$ è chiamato langrangian dual problem di (P).

Vogliamo quindi trovare il miglior lower bound per $v(P)$.

Attenzione: Il dual problem (D) è sempre un convex problem, anche se (P) non è convesso!

OSS:

1) Il duale del duale è il primale

2) Il duale mantiene il grado del primale
   (lineare $\to$ lineare, quadratico $\to$ quadratico)

TEO (weak duality): $\forall$ problema (P), $v(D) \leq v(P)$

$\hookrightarrow$ Se $v(D) = v(P)$ viene detto "Strong Duality".

Se (P) è convesso, esiste un ottimo globale x^* e ACQ è vera in x^*, allora:

1) KKT multiplier (λ^*, μ^*) associati a x^* sono ottimi globali del duale (D), e viceversa.

2) $v(D) = v(P)$

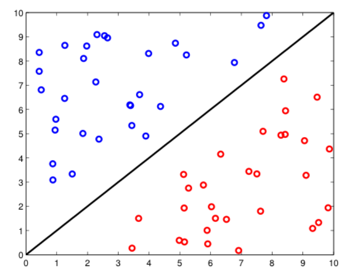
Support Vector Machine

SVM è un binary classification method.

linear SVM

Siano A e B due set "finiti" con i label conosciuti, 1 per i punti in A e −1 per i punti in B.

$A \cup B$ = Training set



Assumiamo che A e B siano linearmente separabili, cioè esiste un hyperpiano $H = \{ x \in \mathbb{R}^n : w^T x + b = 0 \}$ t.c.:

$$w^T x^i + b > 0 \qquad \forall\, x^i \in A,$$
$$w^T x^j + b < 0 \qquad \forall\, x^j \in B.$$

Per determinare la classe di un nuovo dato x, usiamo la decision function:

$$f(x) = \text{sign}(w^T x + b) = \begin{cases} 1 & \text{if } w^T x + b > 0, \\ -1 & \text{if } w^T x + b < 0. \end{cases}$$

1) Qual è la condizione necessaria e sufficiente per far sì che A e B siano linearmente indip, cioè esiste un H che li separa?

L'intersezione dei convex hull di A e B è l'insieme vuoto

$$\text{conv}(A) \cap \text{conv}(B) = \emptyset$$

2) Ci sono più hyperplane possibili (∞) se A e B sono linearmente separabili, quale hyperplane scegliamo?

L'hyperplane che sia il più lontano possibile dal training point più vicino.

Margin of Separation

Se H è l'hyperplane, il margin of separation di H è definito come la minima distanza tra H e $A \cup B$, cioè

$$\rho(H) = \min_{x \in A \cup B} \frac{|w^T x + b|}{\|w\|}.$$

SVM problem and dual

GOAL: Vogliamo quindi trovare l' hyperplane con il massimo margin of separation.

$\hookrightarrow$ Questo è equivalente alla risoluzione del seguente problema quadratico convesso:

$$\begin{cases} \min\limits_{w,b} \ \frac{1}{2}\|w\|^2 \\ w^T x^i + b \geq 1 \quad \forall\, x^i \in A \\ w^T x^j + b \leq -1 \quad \forall\, x^j \in B \end{cases} \quad (1)$$

$\rightarrow$ MATLAB FORM

$$\frac{1}{2}\|w\|^2 = \frac{1}{2}(w^T, b) \cdot \underbrace{\left(\quad\right)}_{Q} \binom{w}{b} + \underbrace{(\quad)}_{c} \binom{w}{b}$$

$$\begin{array}{l} \text{sel } A \rightarrow \\ \\ \text{sel } B \rightarrow \end{array} \left(\quad \rule{3em}{0.4pt} \quad \right) \binom{w}{b} \leq \underbrace{\left(\quad \right)}_{\text{const.}}$$

Possiamo riscrivere il problema, considerando $\ell = |A \cup B|$ e definendo, $\forall x^i \in A \cup B$, un label: $y^i = \begin{cases} 1 & \text{se } x^i \in A \\ -1 & \text{se } x^i \in B \end{cases}$

in questo modo: $\begin{cases} \min\limits_{w,b} \ \frac{1}{2}\|w\|^2 \\ 1 - y^i(w^T x^i + b) \leq 0 \qquad \forall\, i = 1, \ldots, \ell \end{cases} \quad (2)$

OSS: è facile considerare il lagrangian dual del problema (2) e viene usato anche per l'SVM non lineare.

Dual Problem

la lagrangian function è

$$\begin{aligned} L(w, b, \lambda) &= \frac{1}{2}\|w\|^2 + \sum_{i=1}^{\ell} \lambda_i \left[1 - y^i(w^T x^i + b)\right] \\ &= \underbrace{\frac{1}{2}\|w\|^2 - \sum_{i=1}^{\ell} \lambda_i y^i w^T x^i}_{\text{Dipendono da } w} - \underbrace{b \sum_{i=1}^{\ell} \lambda_i y^i}_{\text{dipende da } b} + \underbrace{\sum_{i=1}^{\ell} \lambda_i}_{\text{non dipende da nessuno dei due}} \end{aligned}$$

consideriamo due casi:

1) Se $\sum_{i=1}^{\ell} \lambda_i y^i \neq 0$, allora $\min_{w,b} L(w,b,\lambda) = -\infty$

2) Se $\sum_{i=1}^{\ell} \lambda y^i = 0$, L non dipende da b, L è strongly convex e

$\arg\min_{w} L(w,b,\lambda)$ è dato dallo stationary point (unique)

$$\nabla_w L(w,b,\lambda) = w - \sum_{i=1}^{\ell} \lambda_i y^i x^i = 0$$

quindi, la dual function è:

$$\varphi(\lambda) = \begin{cases} -\infty & \text{if } \sum_{i=1}^{\ell} \lambda_i y^i \neq 0 \\ -\frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} y^i y^j (x^i)^T x^j \lambda_i \lambda_j + \sum_{i=1}^{\ell} \lambda_i & \text{if } \sum_{i=1}^{\ell} \lambda_i y^i = 0 \end{cases}$$

$X = (y^1 x^1, \dots, y^\ell x^\ell)$

$$\begin{cases} \min \frac{1}{2} \lambda^T \underbrace{\boxed{X^T X}}_{Q} \lambda \underbrace{\boxed{- e^T}}_{c} \lambda \\ \underbrace{(y^1, \dots, y^\ell)}_{Aeq} (\lambda) = \underbrace{0}_{Beq} \\ \lambda \geq \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}}_{LB} \end{cases}$$

MATLAB FORM

e il dual problem è

$$\begin{cases} \max_{\lambda} -\frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} y^i y^j (x^i)^T x^j \lambda_i \lambda_j + \sum_{i=1}^{\ell} \lambda_i \\ \sum_{i=1}^{\ell} \lambda_i y^i = 0 \\ \lambda \geq 0 \end{cases}$$

oppure

$$\begin{cases} \max_{\lambda} -\frac{1}{2} \lambda^T X^T X \lambda + e^T \lambda \\ \sum_{i=1}^{\ell} \lambda_i y^i = 0 \\ \lambda \geq 0 \end{cases}$$

dove: $n \times \ell$ matrix $X = (y^1 x^1, y^2 x^2, \dots, y^\ell x^\ell)$ and the vector $e^T = (1, \dots, 1)$.

Abbiamo quindi:

1) Il dual problem è un convex quadratic problem

2) I dual constraints sono più semplici dei primal constraints

3) Il dual problem ha optimal solutions e sono date da ogni KTT multiplier λ^* associato al primal optimum (w^*, b^*)

a) Se $\lambda_i^* > 0$, allora x^i è detto support vector (Importanti per determinare il migliore H)

b) Se λ^* è un dual optimum, allora

$$w^* = \sum_{i=1}^{\ell} \lambda_i^* y^i x^i.$$

c) b^* è ottenuto usando la condizione complementare $\lambda_i^* \left[1 - y^i((w^*)^\top x^i + b^*)\right] = 0;$, infatti, $\forall i$ $\lambda_i > 0$ allora:

$$b^* = \frac{1}{y^i} - (w^*)^\top x^i.$$

d) la decision function è data da:

$$f(x) = \text{sign}((w^*)^\top x + b^*).$$

Cosa succede se A e B non sono linearmente separabili?

Abbiamo che il sistema lineare $1 - y^i(w^\top x^i + b) \le 0 \quad i = 1, \dots, \ell$ non ha soluzioni.

Introduciamo una slack variable $\xi_i \ge 0$ e consideriamo il problema rilassato per considerare gli outliers:

$$\begin{aligned} &1 - y^i(w^\top x^i + b) \le \xi_i && i = 1, \dots, \ell \\ &\xi_i \ge 0 && i = 1, \dots, \ell \end{aligned}$$

Se x^i è missclassificato, allora $\xi_i > 1$.

La somma $\sum_{i=1}^{\ell} \xi_i$ è un limite superiore del numero di punti missclassificati.

Aggiungiamo alla obj. function il termine $C \sum_{i=1}^{\ell} \xi_i$, dove $C > 0$ è un parametro.

linear SVM with sotf margin

$$\begin{cases} \min\limits_{w,b,\xi} \; \dfrac{1}{2}\|w\|^2 + C \sum\limits_{i=1}^{\ell} \xi_i \\ 1 - y^i(w^\top x^i + b) \le \xi_i & \forall\, i = 1, \dots, \ell \\ \xi_i \ge 0 & \forall\, i = 1, \dots, \ell \end{cases}$$

Il dual problem è

$$\begin{cases} \max\limits_{\lambda} \; -\frac{1}{2}\sum\limits_{i=1}^{\ell}\sum\limits_{j=1}^{\ell} y^i y^j (x^i)^\top x^j \lambda_i \lambda_j + \sum\limits_{i=1}^{\ell} \lambda_i \\ \sum\limits_{i=1}^{\ell} \lambda_i y^i = 0 \\ 0 \le \lambda_i \le \underline{C} \qquad i = 1, \dots, \ell \end{cases}$$

→ A differenza del duale senza soft margin, abbiamo solo un limite superiore su λ

Se λ^* è un ottimo per il duale, allora $w^* = \sum\limits_{i=1}^{\ell} \lambda_i^* y^i x^i$.

Dobbiamo trovare b^* t.c $0 < \lambda_i^* < C$ e usando le condizioni complementari

$$\begin{cases} \lambda_i^* \left[1 - y^i((w^*)^\top x^i + b^*) - \xi_i^*\right] = 0 \\ (C - \lambda_i^*)\xi_i^* = 0 \end{cases}$$

otteniamo che:

$$b^* = \frac{1}{y^i} - (w^*)^\top x^i.$$

Non linear SVM

Nel caso di un modello SVM non lineare, abbiamo il problema:

$$\begin{cases} \min\limits_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum\limits_{i=1}^{\ell} \xi_i \\ 1 - y^i(w^\top \phi(x^i) + b) \le \xi_i & \forall\, i = 1, \dots, \ell \\ \xi_i \ge 0 & \forall\, i = 1, \dots, \ell \end{cases}$$

w is a vector in a high dimensional space (maybe infinite variables)

e il suo duale:

$$\begin{cases} \max\limits_{\lambda} \; -\frac{1}{2}\sum\limits_{i=1}^{\ell}\sum\limits_{j=1}^{\ell} y^i y^j \phi(x^i)^\top \phi(x^j) \lambda_i \lambda_j + \sum\limits_{i=1}^{\ell} \lambda_i \\ \sum\limits_{i=1}^{\ell} \lambda_i y^i = 0 \\ 0 \le \lambda_i \le C \qquad \forall\, i = 1, \dots, \ell \end{cases}$$

Nel dual problem non abbiamo w che può → avere dimensionalità infinita, ma abbiamo tante incognite quanti sono i punti nel training set

dove $\phi : \mathbb{R}^n \to H$ è una funzione che mappa i punti in uno spazio con maggiore dimensionalità.

1) Risolviamo il duale
2) Calcoliamo $w^* = \sum\limits_{i=1}^{\ell} \lambda_i^* y^i \phi(x^i)$

3) Usiamo ogni λ_i^* t.c $0 < \lambda_i^* < C$ per trovare b^*:

$$y^i \left[\sum_{j=1}^{\ell} \lambda_j^* y^j \phi(x^j)^\top \phi(x^i) + b^* \right] - 1 = 0$$

4) Decision function:

$$f(x) = \text{sign}((w^*)^\top \phi(x) + b^*) = \text{sign}\left(\sum_{i=1}^{\ell} \lambda_i^* y^i \phi(x^i)^\top \phi(x) + b^* \right)$$

Come possiamo vedere, nel duale non abbiamo bisogno di sapere l'immagine dei punti ma il loro prodotto scalare.

$\lambda^* \rightarrow$ know $\phi(x^i)^\top \phi(x^j)$
$\phi(x^i)^\top \phi(x)$
$b^* \rightarrow$ know $\phi(x^i)^\top \phi(x^j)$

Quindi anche se l'immagine può essere in infinite dimensioni, il prodotto è facile da calcolare.

Kernel Functions

Una funzione $k : R^n \times R^n$ è chiamata kernel se esiste $\phi : R^n \to H$ t.c

$$k(x, y) = \langle \phi(x), \phi(y) \rangle,$$

Esempi:

- $k(x, y) = x^\top y$
- $k(x, y) = (x^\top y + 1)^p$, with $p \geq 1$ (polynomial)
- $k(x, y) = e^{-\gamma \|x-y\|^2}$ (Gaussian)
- $k(x, y) = \tanh(\beta x^\top y + \gamma)$, with suitable β and γ

Teo: Se $k : R^n \times R^n \to R$ è un kernel e $x^1, \ldots, x^\ell \in R^n$, allora la matrice K definita in questo modo $K_{ij} = k(x^i, x^j)$ è semidefinita positiva.

Possiamo quindi definire il duale come:

$$\begin{cases} \max\limits_{\lambda} \ -\frac{1}{2} \sum\limits_{i=1}^{\ell} \sum\limits_{j=1}^{\ell} y^i y^j k(x^i, x^j) \lambda_i \lambda_j + \sum\limits_{i=1}^{\ell} \lambda_i \\ \sum\limits_{i=1}^{\ell} \lambda_i y^i = 0 \\ 0 \leq \lambda_i \leq C \qquad i = 1, \ldots, \ell \end{cases}$$

1) Si sceglie k

2) Si trovano le optimal solutions λ^* del duale

3) Si sceglie i t.c $0 < \lambda_i^* < C$ e b^*:

$$b^* = \frac{1}{y^i} - \sum_{j=1}^{\ell} \lambda_j^* y^j k(x^i, x^j)$$

4) Decision Function:

$$f(x) = \text{sign}\left(\sum_{i=1}^{\ell} \lambda_i^* y^i k(x^i, x) + b^* \right)$$

la separating surface $f(x) = 0$ è:

1) lineare nello spazio delle features

2) Non-lineare nello spazio degli input

Regression Problem

Dato un training set, vogliamo trovare una funzione che rappresenta la miglior approssimazione a essi.

Polynomial Regression

Nel caso polinomiale, vogliamo trovare un polinomio p di grado $n-1$, con $n \leq \ell$.

$$p = z_1 + z_2 x + \dots + z_n x^{n-1}$$

dove $z_1, \dots, z_n$ sono i coefficenti.

Il residual è il vettore $r \in \mathbb{R}^\ell$ t.c $r_i = p(x_i) - y_i$ con $i = 1, \dots, n$

cioè la differenza tra il valore reale e il valore trovato dal polinomio.

GOAL: Vogliamo trovare i coeff. z di p t.c $\|r\|$ è minimo, cioè

$$\begin{cases} \min \ \|\overbrace{Az}^{p} - y\| \\ z \in \mathbb{R}^n \end{cases} \quad \text{dove} \quad A = \begin{pmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_\ell & x_\ell^2 & \dots & x_\ell^{n-1} \end{pmatrix} \qquad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_\ell \end{pmatrix}$$

Per qualsiasi norma, noi consideremo $\|\cdot\|_2$, $\|\cdot\|_1$ e $\|\cdot\|_\infty$, $f(z) = \|Az - y\|$ è convessa.

Questo perché la norma di un affine function è una convex function.

Quindi il problema è un unconstrained problem, convesso.

caso $\|\cdot\|_2$ (least square approx):

Quadratic unconstrained convex problem

$$\begin{cases} \min \ \frac{1}{2}\|Az - y\|_2^2 = \frac{1}{2}(Az - y)^T(Az - y) = \frac{1}{2}z^T A^T A z - z^T A^T y + \frac{1}{2}y^T y \\ z \in \mathbb{R}^n \end{cases}$$

dove $\underbrace{\text{rank}(A) = n}_{\text{colonne lin. indip.}}$ e quindi $A^T A$ positive definite

di conseguenza, la soluzione ottimale è unica ed è lo stationary point della obj function, cioè:

la soluzione del sistema è : $A^T A z = A^T y$

Caso $\|\cdot\|_1$ (linear Programming problem):

$$\begin{cases} \min \ \|Az - y\|_1 = \sum_{i=1}^{\ell} |A_i z - y_i| \\ z \in \mathbb{R}^n \end{cases} = \begin{cases} \min\limits_{z,u} \ \sum_{i=1}^{\ell} u_i \\ u_i = |A_i z - y_i| \\ \quad = \max\{A_i z - y_i, y_i - A_i z\} \end{cases} = \begin{cases} \min\limits_{z,u} \ \sum_{i=1}^{\ell} u_i \\ u_i \geq \max\{A_i z - y_i, y_i - A_i z\} \end{cases}$$

rilassiamo il constraint

↓

non cambiamo l'insieme delle opt. solutions ma solo quello del feasible sol.

$$= \begin{cases} \min\limits_{z,u} \ \sum_{i=1}^{\ell} u_i \\ u_i \geq A_i z - y_i & \forall\, i = 1, \dots, \ell \\ u_i \geq y_i - A_i z & \forall\, i = 1, \dots, \ell \end{cases}$$

Caso $\|\cdot\|_\infty$ (linear Prog. problem)

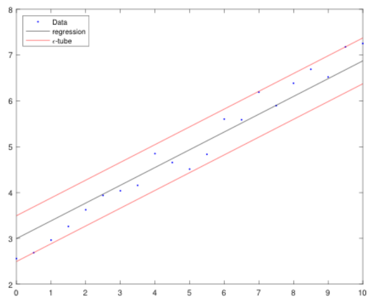
$$\begin{cases} \min \ \|Az - y\|_\infty = \max\limits_{i=1,\dots,\ell} |A_i z - y_i| \\ z \in \mathbb{R}^n \end{cases} = \begin{cases} \min \ u \\ u = \max\limits_{i=1,\dots,\ell} |A_i z - y_i| \end{cases} \approx \begin{cases} \min\limits_{z,u} \ u \\ u \geq A_i z - y_i & \forall\, i = 1, \dots, \ell \\ u \geq y_i - A_i z & \forall\, i = 1, \dots, \ell \end{cases}$$

ε - SV regression

È un modello di regressione lineare basato su SVM.

Vogliamo trovare un funzione che:

1) Ha al massimo un deviazione ε dai target y_i di tutti i training data
2) È il più "piatta" possibile



⇒ Contiene tutti i punti

Consideriamo la affine function $f(x) = w^T x + b$ e un tolerance parameter ε.

Vogliamo risolvere il seguente problema:

$$\begin{cases} \min\limits_{w,b} \frac{1}{2}\|w\|^2 \\ y_i \le w^T x_i + b + \varepsilon \qquad \forall\, i = 1,\dots,\ell \\ y_i \ge w^T x_i + b - \varepsilon \qquad \forall\, i = 1,\dots,\ell \end{cases}$$

Problema: Se ε troppo piccolo, il modello potrebbe non essere possibile.

ε - SV regression con slack variables

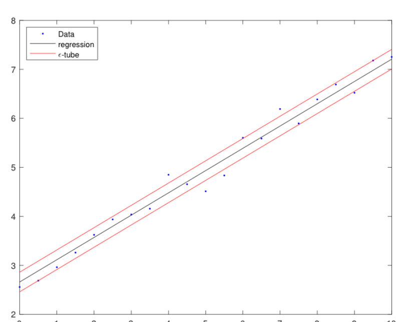
L' SVM model viene esteso con le slack variables ξ^+ e ξ^- per rilassare i vincoli

$$\begin{cases} \min\limits_{w,b,\xi^+,\xi^-} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i^+ + \xi_i^-) \\ y_i \le w^T x_i + b + \varepsilon + \xi_i^+ \qquad \forall\, i = 1,\dots,\ell \\ y_i \ge w^T x_i + b - \varepsilon - \xi_i^- \qquad \forall\, i = 1,\dots,\ell \\ \xi^+ \ge 0 \\ \xi^- \ge 0 \end{cases}$$

⇒ Il parametro C da il trade-off tra la flatness di f e la tolleranza a deviazioni $> \varepsilon$

↓ Duale

$$\begin{cases} \max\limits_{\lambda^+,\lambda^-} -\frac{1}{2}\sum_{i=1}^{\ell}\sum_{j=1}^{\ell}(\lambda_i^+ - \lambda_i^-)(\lambda_j^+ - \lambda_j^-)(x_i)^T x_j \\ \qquad -\varepsilon\sum_{i=1}^{\ell}(\lambda_i^+ + \lambda_i^-) + \sum_{i=1}^{\ell} y_i(\lambda_i^+ - \lambda_i^-) \\ \sum_{i=1}^{\ell}(\lambda_i^+ - \lambda_i^-) = 0 \\ \lambda_i^+ \in [0, C] \\ \lambda_i^- \in [0, C] \end{cases}$$



1) Il duale è un convex quadratic problem

2) I vincoli sono più facili

3) Se $\lambda_i^+ > 0$ o $\lambda_i^- > 0$ $\Rightarrow$ x_i è un support vector

4) Se (λ^+, λ^-) è un dual optimum, allora

$$w = \sum_{i=1}^{\ell} (\lambda_i^+ - \lambda_i^-) x_i,$$

5) b è ottenuto usando le complementary conditions:

$$\begin{aligned}
&\lambda_i^+ \left[\varepsilon + \xi_i^+ - y_i + w^\mathsf{T} x_i + b\right] = 0\\
&\lambda_i^- \left[\varepsilon + \xi_i^- + y_i - w^\mathsf{T} x_i - b\right] = 0\\
&\xi_i^+ (C - \lambda_i^+) = 0\\
&\xi_i^- (C - \lambda_i^-) = 0
\end{aligned}$$

a) Se c'è qualche i t.c $0 < \lambda_i^+ < C$ $\Rightarrow$ $b = y_i - w^\mathsf{T} x_i - \varepsilon;$

b) Se c'è qualche i t.c $0 < \lambda_i^- < C$ $\Rightarrow$ $b = y_i - w^\mathsf{T} x_i + \varepsilon.$

Non-linear SVM

Per generare una non linear regression function usiamo il kernel.

Primal problem:

$$\begin{cases}
\min \ \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{\ell} (\xi_i^+ + \xi_i^-) \\
y_i \le w^\mathsf{T} \phi(x_i) + b + \varepsilon + \xi_i^+ & \forall\, i = 1, \dots, \ell \\
y_i \ge w^\mathsf{T} \phi(x_i) + b - \varepsilon - \xi_i^- & \forall\, i = 1, \dots, \ell
\end{cases}$$

$\Rightarrow$ Possibe ∞ variabili

w is a vector in a high dimensional space (maybe infinite variables)

$\downarrow$ Duale

Dual problem:

$$\begin{cases}
\max\limits_{(\lambda^+, \lambda^-)} \ -\frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} (\lambda_i^+ - \lambda_i^-)(\lambda_j^+ - \lambda_j^-) \phi(x_i)^\mathsf{T} \phi(x_j) \\
\qquad - \varepsilon \sum_{i=1}^{\ell} (\lambda_i^+ + \lambda_i^-) + \sum_{i=1}^{\ell} y_i (\lambda_i^+ - \lambda_i^-) \\
\sum_{i=1}^{\ell} (\lambda_i^+ - \lambda_i^-) = 0 \\
\lambda_i^+, \lambda_i^- \in [0, C]
\end{cases}$$

$\Rightarrow$ 2ℓ variabili

1) Scegliere il kernel

2) Risolvere il duale (trovare λ^+, λ^-)

3) trovare b:

a) $b = y_i - \varepsilon - \sum_{j=1}^{\ell} (\lambda_j^+ - \lambda_j^-) k(x_i, x_j),$ for some i s.t. $0 < \lambda_i^+ < C$

b) $b = y_i + \varepsilon - \sum_{j=1}^{\ell} (\lambda_j^+ - \lambda_j^-) k(x_i, x_j),$ for some i s.t. $0 < \lambda_i^- < C$

4) Regression function

$$f(x) = \sum_{i=1}^{\ell} (\lambda_i^+ - \lambda_i^-) k(x_i, x) + b$$

Essa è :

a) lineare nel feature space

b) Non lineare nell' input space

Clustering

Dato un insieme S di patterns e un intero k, trovare una partizione di S in k subset $S_1 \dots S_k$ (clusters) che sono omogenei e ben separati.

Assumiamo che i patterns sono vettori $p_1, \dots, p_\ell \in \mathbb{R}^n$.

Consideriamo una distanza d tra i vettori $\Rightarrow$ 1) $d(x,y) = \|x-y\|_2^2$ = least sq. dist.

o 2) $d(x,y) = \|x-y\|_1$ = Manatthan dist.

Per ogni cluster S_J introduciamo un centroide $x_J \in \mathbb{R}^n$ e ogni pattern è associato al centroide più vicino.

GOAL: Trovare k centroidi in modo da minimizzare la somma delle distanze tra ogni pattern e il centroide più vicino.
$$\begin{cases} \min \sum_{i=1}^{\ell} \min_{j=1,\dots,k} d(p_i, x_j) \\ x_j \in \mathbb{R}^n \qquad \forall\, j = 1, \dots, k \end{cases}$$
$\Rightarrow$ Problema: Questa obj f non è solitamente convessa.

Caso $\|\cdot\|_2$:

L' optimization problem diventa il seguente
$$\begin{cases} \min \sum_{i=1}^{\ell} \min_{j=1,\dots,k} \|p_i - x_j\|_2^2 \\ x_j \in \mathbb{R}^n \qquad \forall\, j = 1, \dots, k \end{cases}$$

1) Se k=1 (un cluster), il problema è convesso e senza constraints

(1) $$\begin{cases} \min \sum_{i=1}^{\ell} \|p_i - x\|_2^2 = \sum_{i=1}^{\ell} (x-p_i)^T (x-p_i) \\ x \in \mathbb{R}^n \end{cases}$$
$\Rightarrow$ global optimum
$$x = \frac{\sum_{i=1}^{\ell} p_i}{\ell}$$
(mean or baricenter)

2) Se $k>1$, il problema non è convesso e non è smooth

$$\begin{cases} \min\limits_{x} \sum\limits_{i=1}^{\ell} \min\limits_{j=1,\dots,k} \|p_i - x_j\|_2^2 \\ x_j \in \mathbb{R}^n \qquad \forall\, j = 1,\dots,k \end{cases} \qquad (2)$$

TEO: Il problema (2) è equivalente al seguente nonconvex smooth problem

$$\begin{cases} \min\limits_{x,\alpha} \sum\limits_{i=1}^{\ell} \sum\limits_{j=1}^{k} \alpha_{ij} \|p_i - x_j\|_2^2 \\ \sum\limits_{j=1}^{k} \alpha_{ij} = 1 \quad \forall\, i = 1,\dots,\ell \\ \alpha_{ij} \geq 0 \qquad \forall\, i = 1,\dots,\ell,\ j = 1,\dots,k \\ x_j \in \mathbb{R}^n \qquad \forall\, j = 1,\dots,k. \end{cases} \qquad (3)$$

In questa forma, se fissiamo x o α il problema rispetto alla variabile fissata, diventa facile da risolvere.

Su questo concetto si basa il k-means algorithm.

K-means algorithm

1) Se viene fissato x_j, allora il problema (3) è decomponibile in ℓ linear ← LP problem semplici.

$\forall\, i = 1 \dots \ell$, la soluzione ottimale è:

$$\alpha_{ij}^* = \begin{cases} 1 & \text{if } j \text{ is the first index s.t. } \|p_i - x_j\|_2 = \min\limits_{h=1,\dots,k} \|p_i - x_h\|_2 \\ & \quad (x_j \text{ is the first closest centroid to } p_i), \\ 0 & \text{otherwise.} \end{cases}$$

2) Se α_{ij} è fissato, allora (3) è decomponibile in k semplici convex QP problems (quadratic ↑ QP)

simili al problema con $k=1$.

$\forall\, j = 1 \dots k$, la soluzione ottimale è:

$$x_j^* = \frac{\sum\limits_{i=1}^{\ell} \alpha_{ij} p_i}{\sum\limits_{i=1}^{\ell} \alpha_{ij}}$$

Il k-means algo consiste nell'alternare la minimizzazione di $f(x,\alpha)=\sum_{i=1}^{\ell}\sum_{j=1}^{k}\alpha_{ij}\|p_i-x_j\|_2^2$ rispetto al fix delle var. x e α.

1) Inizializzazione:

a) Settare $t=0$

b) Scegliere i centroid $x_1^0,\dots,x_k^0$ e assegnare i pattern ai cluster:

$\forall i=1\dots\ell$

$$\alpha_{ij}^0=\begin{cases}1 & \text{if } j \text{ is the first index s.t. } \|p_i-x_j^0\|_2=\min_{h=1,\dots,k}\|p_i-x_h^0\|_2\\ 0 & \text{otherwise.}\end{cases}$$

2) Aggiornare i centroid: $\forall j=1\dots k$, calcola il mean

$$x_j^{t+1}=\left(\sum_{i=1}^{\ell}\alpha_{ij}^t p_i\right)\Big/\left(\sum_{i=1}^{\ell}\alpha_{ij}^t\right).$$

3) Aggiornare i clusters: $\forall i=1\dots\ell$, calcola

$$\alpha_{ij}^{t+1}=\begin{cases}1 & \text{if } j \text{ is the first index s.t. } \|p_i-x_j^{t+1}\|_2=\min_{h=1,\dots,k}\|p_i-x_h^{t+1}\|_2\\ 0 & \text{otherwise.}\end{cases}$$

4) Stopping criterion:

se $f(x^{t+1},\alpha^{t+1})=f(x^t,\alpha^t) \Rightarrow$ Stop

altrimenti $\hookrightarrow t=t+1 \Rightarrow$ step 1

TEO: L'algoritmo si ferma dopo un numero finito di iterazioni in una soluzione (x^*,α^*) del KKT system del problema (3) t.c.:

1) $f(x^*,\alpha^*)\le f(x^*,\alpha), \qquad \forall\,\alpha\ge 0 \text{ s.t. } \sum_{j=1}^{k}\alpha_{ij}=1 \quad \forall\, i=1,\dots,\ell,$

2) $f(x^*,\alpha^*)\le f(x,\alpha^*), \qquad \forall\, x\in\mathbb{R}^{kn}.$

ma non garantisce un ottimo globale!

Caso $\|\cdot\|_1$:

Il problema diventa così:

$$\begin{cases} \min \sum_{i=1}^{\ell} \min_{j=1,\dots,k} \|p_i - x_j\|_1 \\ x_j \in \mathbb{R}^n \qquad \forall\, j = 1,\dots,k \end{cases}$$

1) Se $k=1$: È convesso (non quadratico) decomponibile in n convex problem in una variabile.

$$\begin{cases} \min \displaystyle\sum_{i=1}^{\ell} \|p_i - x\|_1 = \sum_{i=1}^{\ell}\sum_{h=1}^{n} |x_h - (p_i)_h| = \sum_{h=1}^{n} \underbrace{\sum_{i=1}^{\ell} |x_h - (p_i)_h|}_{f_h(x_h)} \\ x \in \mathbb{R}^n \end{cases} \tag{4}$$

Possiamo decomporre il problema in n convex problem con una variabile

Dati ℓ numeri reali $a_1,\dots,a_\ell$, qual è la soluzione ottimale per:

$$\begin{cases} \min \displaystyle\sum_{i=1}^{\ell} |x - a_i| = f(x) \\ x \in \mathbb{R} \end{cases}$$

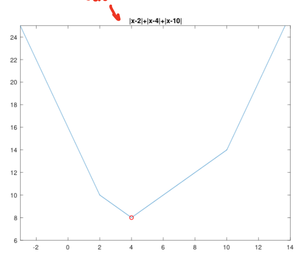
$f(x)$ può essere scritta nel seguente modo:

$$f(x) = \begin{cases} -\ell x + \sum_{i=1}^{\ell} a_i & \text{if } x < a_1 \\ (2-\ell)x + \sum_{i=2}^{\ell} a_i - a_1 & \text{if } x \in [a_1, a_2] \\ \dots & \dots \\ (2r-\ell)x + \sum_{i=r+1}^{\ell} a_i - \sum_{i=1}^{r} a_1 & \text{if } x \in [a_r, a_{r+1}] \\ \dots & \dots \\ (\ell-2)x + a_\ell - \sum_{i=1}^{\ell-1} a_i & \text{if } x \in [a_{\ell-1}, a_\ell] \\ \ell x - \sum_{i=1}^{\ell} a_i & \text{if } x > a_\ell \end{cases}$$

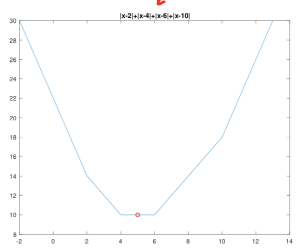
e l'ottimo globale è il median$(a_1,\dots,a_\ell)$, cioè:

$$\begin{cases} a_{(\ell+1)/2} & \text{if } \ell \text{ is odd,} \\ \dfrac{a_{\ell/2} + a_{1+\ell/2}}{2} & \text{if } \ell \text{ is even.} \end{cases}$$

odd



even



2) Se $k>1$, allora il problema è non convesso e non smooth:

$$\begin{cases} \min\limits_{x} \sum\limits_{i=1}^{\ell} \min\limits_{j=1,\dots,k} \|p_i - x_j\|_1 \\ x_j \in \mathbb{R}^n \qquad \forall\, j = 1,\dots,k \end{cases} \qquad (5)$$

Il problema (5) è equivalente al seguente

$$\begin{cases} \min\limits_{x,\alpha} \sum\limits_{i=1}^{\ell}\sum\limits_{j=1}^{k} \alpha_{ij}\|p_i - x_j\|_1 \\ \sum\limits_{j=1}^{k} \alpha_{ij} = 1 \quad \forall\, i = 1,\dots,\ell \\ \alpha_{ij} \geq 0 \qquad \forall\, i = 1,\dots,\ell,\ j = 1,\dots,k \\ x_j \in \mathbb{R}^n \qquad \forall\, j = 1,\dots,k. \end{cases}$$

K-median Algorithm

Il k-median algo si basa sulle seguenti proprietà del problema (6):

1) Se x_j sono fissati, allora (6) è scomponibile in ℓ semplici linear problem.

$\forall i = 1 \dots \ell$, la soluzione ottimale è

$$\alpha_{ij}^* = \begin{cases} 1 & \text{if } j \text{ is the first index s.t. } \|p_i - x_j\|_1 = \min\limits_{h=1,\dots,k} \|p_i - x_h\|_1 \\ & \quad (x_j \text{ is the first closest centroid to } p_i), \\ 0 & \text{otherwise.} \end{cases}$$

2) Se $\alpha_{ij} \in \{0,1\}$ sono fissati, allora (6) è scomponibile in k semplici convex problem.

$\forall j\ 1 \dots k$, la soluzione ottimale è:

$$x_j^* = \text{median}(p_i : \ \alpha_{ij} = 1).$$

Il k-median algo consiste nell'alternare la minimizzazione di $f(x,\alpha) = \sum\limits_{i=1}^{\ell}\sum\limits_{j=1}^{k} \alpha_{ij}\|p_i - x_j\|_1$ rispetto a x e α.

1) Inizializzazione

a) Setta $t=0$

b) Scegli i centroid $x_1^0, \dots, x_k^0 \in \mathbb{R}^n$ e assegna i pattern ai cluster.

$\forall i = 1 \dots \ell$

$$\alpha_{ij}^0 = \begin{cases} 1 & \text{if } j \text{ is the first index s.t. } \|p_i - x_j^0\|_1 = \min_{h=1,\dots,k} \|p_i - x_h^0\|_1 \\ 0 & \text{otherwise.} \end{cases}$$

2) Update centroids: $\forall j = 1 \dots k$ calcola $x_j^{t+1} = \text{median}(p_i : \ \alpha_{ij}^t = 1)$.

3) update clusters: $\forall i = 1 \dots \ell$ calcola $\alpha_{ij}^{t+1} = \begin{cases} 1 & \text{if } j \text{ is the first index s.t. } \|p_i - x_j^{t+1}\|_1 = \min_{h=1,\dots,k} \|p_i - x_h^{t+1}\|_1 \\ 0 & \text{otherwise.} \end{cases}$

4) Stopping criterion: Se $f(x^{t+1}, \alpha^{t+1}) = f(x^t, \alpha^t) \Rightarrow$ Stop

Altrimenti $t = t+1 \Rightarrow$ Vai allo step 1

TEO: Se k-median algo si ferma dopo un numero finito di iterazioni in un punto stazionario (x^*, α^*) del problema (6) t.c

$$\begin{aligned} & f(x^*, \alpha^*) \le f(x^*, \alpha), && \forall\, \alpha \ge 0 \text{ s.t. } \sum_{j=1}^k \alpha_{ij} = 1 \quad \forall\, i = 1, \dots, \ell, \\ & f(x^*, \alpha^*) \le f(x, \alpha^*), && \forall\, x \in \mathbb{R}^{kn}. \end{aligned}$$

Att. L'algo k-median non garantisce di trovare l'ottimo globale (perché se consideriamo entrambe le var. invece di fissarle, il problema non è convesso)

Methods for Unconstrained optimization problems

Gradient method

Sia un unconstrained problem: $\min_{x \in \mathbb{R}^n} f(x).$

Siamo in un qualsiasi punto x^k della funzione e scegliamo la direzione $d^k = -\nabla f(x^k)$ che fa sì che la funzione decresca il più possibile tra le altre direzioni (Perché ha l'angolo maggiore rispetto al gradiente)

Algorithm (Quadratic Case)

1) Scegliamo $x^0 \in \mathbb{R}^n$, settiamo $k = 0$

2) Fin quando $\nabla f(x_k) \neq 0$ (non siamo in un punto stazionario)

a) Scegliamo la direzione $d^k = -\nabla f(x^k)$ (dove f decresce maggiormente)

b) Vogliamo minimizzare f solo sull'half plane dettato dalla direzione, quindi calcoliamo una soluzione ottimale t_k (scalare) del problema

$$\min_{t>0} f(x^k + t\,d^k)$$

step size

c) Aggiornare x e k

i) $x^{k+1} = x^k + t_k\,d^k$

ii) $k = k+1$

Example. $f(x) = x_1^2 + 2\,x_2^2 - 3\,x_1 - 2\,x_2$, starting point $x^0 = (2,1)$.

f da minimizzare

$\nabla f(x^0) = (1,2)$, $d^0 = (-1,-2)$, $f(x^0 + td^0) = 9\,t^2 - 5\,t - 2$, $t_0 = 5/18$,

Derivata di f nel punto x^0

$-\nabla f(x^0)$

stationary point della derivata min. globale

step size

$$x^1 = (2,1) - \frac{5}{18}(1,2) = \left(\frac{31}{18}, \frac{4}{9}\right).$$

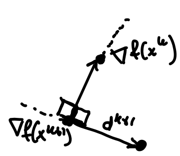
calcolo il nuovo x^{k+1}

OSS: In generale, quando abbiamo una $f = \frac{1}{2}x^TQx + c^Tx$ con Q positive definite matrix abbiamo che:

1) $f(x^k + td^k) = \frac{1}{2}(d^k)^TQ\,d^k\,t^2 + (g^k)^Td^k\,t + f(x^k)$, dove $g^k = \nabla f(x^k) = Q\,x^k + c$.

2) lo step size è uguale a: $t_k = -\frac{(g^k)^Td^k}{(d^k)^TQ\,d^k}$.

Convergenza (Good News)



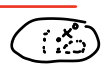
Proposizioni:

1) $(d^k)^T d^{k+1} = 0$ $\forall k$, cioè la direzione successiva è ortogonale alla precedente.

2) Se la sequenza $\{x^k\}$ converge in x^*, allora $\nabla f(x^*) = 0$

TEO: Se f è coerciva, la sequenza $\{x^k\}$ è bounded e qualsiasi dei cluster point sono punti stazionari

$\lim_{\|x\|} f(x) = \infty$



Corollario: Se f è coercive e convex, $\{x^k\}$ è limitata e qualsiasi dei cluster point sono minimi globali di f

Corollario (importante): Se f è strongly convex, $\{x^k\}$ converge a un minimo globale di f, qualsiasi sia x^0

Convergence rate (bad news)

Le sequenze generate si comportano a zig-zag.

TEO (Error Bound)

Se $f(x) = \frac{1}{2} x^T Q x + c^T x$, con Q positive definite matrix e x^* è un global minimum di f, allora la sequenza $\{x^k\}$ soddisfa la seguente disuguaglianza:

$$\|x^{k+1} - x^*\|_Q \leq \left(\frac{\frac{\lambda_n}{\lambda_1} - 1}{\frac{\lambda_n}{\lambda_1} + 1} \right) \|x^k - x^*\|_Q, \quad \forall\, k \geq 0, \quad \text{(linear convergence)}$$

where $\|x\|_Q = \sqrt{x^T Q x}$ and $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ are the eigenvalues of Q.

max autovalore

min autovalore

Se $\lambda_n / \lambda_1 >> 1$, allora il ratio $\left(\frac{\frac{\lambda_n}{\lambda_1} - 1}{\frac{\lambda_n}{\lambda_1} + 1} \right) \simeq 1$ e la convergenza può essere lenta

Gradient per f non quadratiche (Armijo inexact line search)

Quando f non è quadratica, l'exact step size può essere computazionalmente costoso da calcolare.

Set $\alpha, \gamma \in (0,1)$ and $\bar{t} > 0$. Choose $x^0 \in \mathbb{R}^n$, set $k = 0$.

while $\nabla f(x^k) \neq 0$ **do**
  $d^k = -\nabla f(x^k)$
  $t_k = \bar{t}$
  while $f(x^k + t_k d^k) > f(x^k) + \alpha\, t_k\, (d^k)^T \nabla f(x^k)$ **do**
    $t_k = \gamma t_k$
  end
  $x^{k+1} = x^k + t_k d$, $k = k + 1$
end

Conjugate Gradient Method

La direzione d non è più solo $-\nabla f(x)$ ma è una combinazione tra il gradiente calcolato nell'iterazione corrente e la direzione calcolata nell'iterazione precedente.

Quadratic Case

Nel caso quadratico abbiamo che:

1) $f(x) = \frac{1}{2} x^\top Q x + c^\top x$, where Q is positive definite.

2) $g = \nabla f(x) = Qx + c$.

La direzione è data da:

$$d^k = \begin{cases} -g^0 & \text{if } k = 0, \\ -g^k + \beta_k d^{k-1} & \text{if } k \geq 1, \end{cases}$$ dove $b_k = (d^k)^T Q d^{k-1} = 0$

$\downarrow$

b_k t.c d^k e d^{k-1} siano conjugate rispetto a Q.

OSS:

1) b_k è facile da calcolare: $\beta_k = \frac{(g^k)^\top Q d^{k-1}}{(d^{k-1})^\top Q d^{k-1}}$

2) Se calcoliamo l'exact line search, d^k è un direzione decrescente

3) Lo step size $t_k = -\frac{(g^k)^\top d^k}{(d^k)^\top Q d^k}$

Algorithm

Choose $x^0 \in \mathbb{R}^n$, set $g^0 = Q x^0 + c$, $k := 0$
while $g^k \neq 0$ **do**
  if $k = 0$ **then** $d^k = -g^k$
  else $\beta_k = \frac{(g^k)^\top Q d^{k-1}}{(d^{k-1})^\top Q d^{k-1}}$, $d^k = -g^k + \beta_k d^{k-1}$
  end
  $t_k = -\frac{(g^k)^\top d^k}{(d^k)^\top Q d^k}$
  $x^{k+1} = x^k + t_k d^k$, $g^{k+1} = Q x^{k+1} + c$, $k = k + 1$
end

Convergenza

Proposizione: Se non troviamo un min globale dopo k iterazioni, allora: gradienti $\{g^0, \dots, g^k\}$ sono ortogonali e le direzioni $\{d^0, \dots, d^k\}$ sono conjugate

TEO: 1) CG method trova il min glob. in max n iterazioni

2) Se Q ha r autovalori distinti $\Rightarrow$ CG trova il minimo globale in massimo r iterazioni.

Convergence Rate

TEO: Se $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ sono autovalori di Q, allora i seguenti bounds sono veri:

1) $$\|x^k - x^*\|_Q \leq 2 \left(\frac{\sqrt{\frac{\lambda_n}{\lambda_1}} - 1}{\sqrt{\frac{\lambda_n}{\lambda_1}} + 1} \right)^k \|x^0 - x^*\|_Q, \qquad \forall\, k \geq 0,$$

2) $$\|x^k - x^*\|_Q \leq \left(\frac{\lambda_{n-k+1} - \lambda_1}{\lambda_{n-k+1} + \lambda_1} \right) \|x^0 - x^*\|_Q, \qquad \forall\, k \geq 0.$$

Non linear functions

Choose $x^0 \in \mathbb{R}^n$, set $k := 0$

while $\nabla f(x^k) \neq 0$ **do**
  if $k = 0$ **then** $d^k = -\nabla f(x^k)$
  else $\beta_k = \frac{\|\nabla f(x^k)\|^2}{\|\nabla f(x^{k-1})\|^2}$, $\quad d^k = -\nabla f(x^k) + \beta_k\, d^{k-1}$
  end
  Compute the step size t_k
  $x^{k+1} = x^k + t_k\, d^k$, $k = k + 1$
end

Proposizioni:

1) Exact line search, d^k è la descent direction

2) Inexact line search

$$\begin{cases} f(x^k + t_k\, d^k) \leq f(x^k) + \alpha\, t_k\, \nabla f(x^k)^\top d^k, \\ |\nabla f(x^k + t_k\, d^k)^\top d^k| \leq -\beta\, \nabla f(x^k)^\top d^k, \end{cases} \qquad (1)$$

with $0 < \alpha < \beta < 1/2$, then d^k is a descent direction.

TEO: Se f è coercive $\Rightarrow$ se t_k soddisfa (1), genera una sequenza $\{x^k\}$ t.c

$$\liminf_{k \to \infty} \|\nabla f(x^k)\| = 0.$$

Newton Method

Se è possibile calcolare la derivata seconda, possiamo usare il newton Method

Vogliamo trovare il punto stazionario $\nabla f(x) = 0$

Ad ogni iterazione k, facciamo una linear approximation di $\nabla f(x)$ in x^k, cioè

$$\nabla f(x) \simeq \nabla f(x^k) + \nabla^2 f(x^k)(x - x^k),$$

e il nuovo x^{k+1} è la soluzione del sistema lineare

$$\nabla f(x^k) + \nabla^2 f(x^k)(x - x^k) = 0.$$

Algorithm

Choose $x^0 \in \mathbb{R}^n$, set $k = 0$
while $\nabla f(x^k) \neq 0$ **do**
  Solve the linear system $\nabla^2 f(x^k) d^k = -\nabla f(x^k)$
  $x^{k+1} = x^k + d^k$, $k = k + 1$
end

Convergenza

Se x^* è un minimo locale di f e $\nabla^2 f(x^*)$ è positive definite, allora esiste $\delta > 0$ t.c $\forall x^0 \in B(x^*, \delta)$ la sequenza $\{x^k\}$ converge in x^* e per quale $C > 0$, $\bar{k} > 0$:

$$\|x^{k+1} - x^*\| \leq C \|x^k - x^*\|^2 \qquad \forall\, k > \bar{k}, \qquad \text{(quadratic convergence)}$$

OSS:

✓ 1) la convergenza è quadratica (veloce)

× 2) Per la convergenza richiede che x^0 sia "vicino all' ottimo x^*", se x^0 è lontano non è garantita la convergenza

× 3) Ad ogni iterazione dobbiamo calcolare sia $\nabla f(x^k)$ che $\nabla^2 f(x^k)$

Newton Method con line search

Se f è strongly convex, possiamo assicurare la convergenza globalmente (anche da punti lontani) combinando la direzione data dal newton method con lo step size dato dall' Armijo search.

Set $\alpha, \gamma \in (0, 1)$, $\bar{t} > 0$. Choose $x^0 \in \mathbb{R}^n$, set $k = 0$
while $\nabla f(x^k) \neq 0$ **do**
  [search direction] Solve the linear system $\nabla^2 f(x^k) d^k = -\nabla f(x^k)$
  $t_k = \bar{t}$
  while $f(x^k + t_k d^k) > f(x^k) + \alpha\, t_k\, (d^k)^\mathsf{T} \nabla f(x^k)$ **do**
    $t_k = \gamma t_k$
  end
  $x^{k+1} = x^k + t_k d^k$, $k = k + 1$
end

Convergenza

Se f è strongly convex, $\{x^k\}$ converge al global min per qualsiasi x^0.

In più, se $\alpha \in (0, 1/2)$ e $\bar{t} = 1$ $\Rightarrow$ la convergenza è quadratica

Quasi - Newthon Methods

Ad ogni iterazione $\left(\nabla^2 f(x^k)\right)^{-1}$ è approssimata da una matrice positive definite H_k

Algorithm

Choose $x^0 \in \mathbb{R}^n$, a positive definite matrix H_0, $k = 0$

while $\nabla f(x^k) \neq 0$ **do**
  $d^k = -H_k \nabla f(x^k)$
  Compute step size t_k
  $x^{k+1} = x^k + t_k d^k$
  update H_{k+1}
  $k = k + 1$
end

Come aggiornare H_k?

1) Davidon - Fletcher - Powell method:

$$H_{k+1} = H_k + \frac{p^k (p^k)^\top}{(p^k)^\top g^k} - \frac{H_k g^k (g^k)^\top H_k}{(g^k)^\top H_k g^k},$$

dove : a) $p^k = x^{k+1} - x^k$

b) $g^k = \nabla f(x^{k+1}) - \nabla f(x^k)$

2) Broyden - Fletcher - Goldfarb - Shanno (BFGS)

$$H_{k+1} = H_k + \left(1 + \frac{(g^k)^\top H_k g^k}{(p^k)^\top g^k}\right) \frac{p^k (p^k)^\top}{(p^k)^\top g^k} - \frac{p^k (g^k)^\top H_k + H_k g^k (p^k)^\top}{(p^k)^\top g^k}.$$

Derivative - Free Methods

Sono metodi che non usano le derivate ma fanno un sample dell' obj function in un numero finito di punti ad ogni iterazione.

Positive Basis : è un insieme di vettori $\{v^1, \dots, v^p\}$ t.c :

1) Ogni $x \in \mathbb{R}^n$ è una conic combination di $v^1, \dots, v^p$, cioè esistono

$$\boxed{\alpha_1, \dots, \alpha_p \geq 0 \quad \text{t.c} \quad x = \sum_{i=1}^{p} \alpha_i v^i}$$

Linear comb. con tutti i parametri ≥ 0 (stanno nel cono dei due vettori)

2) $\forall i = 1, \dots, p$, v^i non è una conic combination degli altri $v^1, \dots, v^p$

Proposizione: Se $\{v^1, \dots, v^p\}$ è una positive basis, allora $\forall w \in \mathbb{R}^n \setminus \{0\}$ esiste $i \in \{1, \dots, p\}$ t.c $w^T v^i < 0$

Cioè possiamo usare come direzione decrescente un vettore v^i della positive basis.

Algorithm

Choose starting point $x^0 \in \mathbb{R}^n$, step size $t_0 > 0$, $\beta \in (0,1)$, tolerance $\varepsilon > 0$ and a positive basis D. Set $k = 0$.

while $t_k > \varepsilon$ **do**
  Order the poll set $\{x^k + t_k d, \quad d \in D\}$
  Evaluate f at the poll points following the chosen order
  If there is a poll point s.t. $f(x^k + t_k d) < f(x^k)$
    then $x^{k+1} = x^k + t_k d$, $t_{k+1} = t_k$ (successful iteration)
  else $x^{k+1} = x^k$, $t_{k+1} = \beta t_k$ (step size reduction)
  end
  $k = k + 1$
end

Teorema

Assumiamo che $\forall v^i \in D$, $v^i \in \mathbb{Z}^n$.

Se f è coercive e continuosly differentiable, $\{x^k\}$ ha un cluster point x^* t.c $\nabla f(x^*) = 0$

OSS:

1) L' assunzione $\forall v^i \in D$, $v^i \in \mathbb{Z}^n$ può essere rimossa se accettiamo iterazioni nuove se soddisfano una "sufficent" decrease condition

$$f(x^{k+1}) \leq f(x^k) - t_k^2.$$

2) Se $f(x^{k+1}) \leq f(x^k + t_k d) \quad \forall\, d \in D$, allora ogni cluster point di $\{x^k\}$ è uno stationary point di f e $\lim_{k \to \infty} \|\nabla f(x^k)\| = 0$.

Methods for Constrained optimization Problems

Sia P un constrained Problem con linear equality constraints

$$\begin{cases} \min \ f(x) \\ Ax = b \end{cases}$$

- dove f è strongly convex e ha la deriv. seconda
- A è una matrice $p \times n$ con $\text{rank}(A) = p$

Questo problema è equivalente all' unconstrained problem.

Scrivi $A = (A_B, A_N)$ con $\det(A_B) \neq 0$, allora $Ax = b$ è equivalente a

$$A_B x_B + A_N x_N = b \quad \Longrightarrow \quad x_B = A_B^{-1}(b - A_N x_N),$$

quindi:

$$\begin{cases} \min \ f(x) \\ Ax = b \end{cases} \quad \text{is equivalent to} \quad \begin{cases} \min \ f(A_B^{-1}(b - A_N x_N), x_N) \\ x_N \in \mathbb{R}^{n-p} \end{cases}$$

Active set method (quadratic)

Sia un problema quadratico

$$\begin{cases} \min \ \frac{1}{2} x^T Q x + c^T x \\ Ax \leq b \end{cases}$$

dove

- Q è positivo definito
- $\forall x \in \Omega$ i vettori $\{A_i : A_i x = b_i\}$ sono lin. indipendenti

L' active set method risolve ad ogni iterazione un problema quadratico con sol: equality constraints

0) Scegli un feasible point x^0, imposta $W^0 = \{i : A_i x^0 = b_i\}$ (working set) e $k = 0$

1) Trova la soluzione ottimale per il problema y^k

$$\begin{cases} \min \ \frac{1}{2} x^T Q x + c^T x \\ A_i x = b_i \qquad \forall \ i \in W_k \end{cases}$$

2) Se $y^k \neq x^k$ vai allo step 3

Altrimenti vai allo step 4

3) Se y^k è feasible $\Rightarrow t_k = 1$

a) Altrimenti $\quad t_k = \min \left\{ \dfrac{b_i - A_i x^k}{A_i(y^k - x^k)} : \ i \notin W_k, \ A_i(y^k - x^k) > 0 \right\}$

b) $x^{k+1} = x^k + t_k(y^k - x^k)$

c) $W_{k+1} = W_k \cup \{i \notin W_k : \ A_i x^{k+1} = b_i\}$,

d) $k = k + 1$

e) Vai allo step 1

4) Calcola i KKT multipliers μ^k che fanno riferimento a y^k

Se $\mu^k \geq 0 \Rightarrow$ STOP

Altrimenti a) $x^{k+1} = x^k$

b) $\mu_j^k = \min_{i \in W_k} \mu_i^k$

c) $W_{k+1} = W_k \setminus \{j\}$

d) $k = k + 1$

e) Vai allo step 1

Esempio

Example. Solve the problem

$$\begin{cases} \min \ \frac{1}{2}(x_1 - 3)^2 + (x_2 - 2)^2 \\ -2x_1 + x_2 \leq 0 \\ x_1 + x_2 \leq 4 \\ -x_2 \leq 0 \end{cases}$$

by means of the active-set method starting from $x^0 = (0, 0)$.

The working set $W_0 = \{1, 3\}$ hence $y^0 = x^0$; KKT multipliers are $\mu_1^0 = -3/2$, $\mu_3^0 = -11/2$. The new point $x^1 = x^0$ with $W_1 = \{1\}$; y^1 is the optimal solution of

$$\begin{cases} \min \ \frac{1}{2}(x_1 - 3)^2 + (x_2 - 2)^2 \\ -2x_1 + x_2 = 0 \end{cases}$$

which is equivalent to

$$\begin{cases} \min \ \frac{9}{2}x_1^2 - 11x_1 \\ x_1 \in \mathbb{R} \end{cases}$$

thus $y^1 = (11/9, 22/9)$ which is feasible, therefore $x^2 = y^1$ and $W_2 = \{1\}$. We already know that $y^2 = x^2$; the KKT multiplier is $\mu_1^2 = -8/9$, hence $x^3 = x^2$ and $W_3 = \emptyset$.

The optimal solution $y^3 = (3, 2)$ is not feasible, the step size is

$$t_3 = \min\left\{ \frac{b_2 - A_2 x^3}{A_2(y^3 - x^3)}, \frac{b_3 - A_3 x^3}{A_3(y^3 - x^3)} \right\} = \min\left\{ \frac{1}{4}, \frac{11}{2} \right\} = \frac{1}{4},$$

$x^4 = x^3 + t_3(y^3 - x^3) = (5/3, 7/3)$ and $W_4 = \{2\}$. The optimal solution $y^4 = (7/3, 5/3)$ is feasible, hence $x^5 = y^4$ and $W_5 = \{2\}$. Finally, $y^5 = x^5$ and $\mu_2^5 = 2/3 > 0$, thus x^5 is the global minimum of the original problem.

Penality Method (general problem)

Sia P un constrained problem

$$\begin{cases} \min \ f(x) \\ g_i(x) \leq 0 \qquad \forall \ i = 1, \dots, m \end{cases}$$

definiamo la quadratic penality function

$$p(x) = \sum_{i=1}^{m} (\max\{0, g_i(x)\})^2$$ => Positiva solo quando il vincolo $g_i(x)$ non è rispettato.

e consideriamo successivamente l' unconstrained penalized problem

$$\begin{cases} \min\ f(x) + \dfrac{1}{\varepsilon} p(x) := p_\varepsilon(x) \\ x \in \mathbb{R}^n \end{cases}$$ dove $$p_\varepsilon(x) \begin{cases} = f(x) & \text{if } x \in \Omega \\ > f(x) & \text{if } x \notin \Omega \end{cases}$$

0) Imposta $\varepsilon_0 > 0$, $\tau \in (0,1)$, $k=0$

1) Trova la soluzione ottimale x^k del problema penalizzato P_{ε_k}

2) Se $x^k \in \Omega$ => STOP

Altrimenti a) $\varepsilon_{k+1} = \tau \varepsilon_k$

b) $k = k+1$

c) Torna al passo 1

Proposizioni

- If f, g_i are continuously differentiable, then p_ε is continuously differentiable and $\nabla p_\varepsilon(x) = \nabla f(x) + \frac{2}{\varepsilon} \sum_{i=1}^{m} \max\{0, g_i(x)\} \nabla g_i(x)$
- If f and g_i are convex, then p_ε is convex
- Any (P_ε) is a relaxation of (P), i.e., $v(P_\varepsilon) \le v(P)$ for any $\varepsilon > 0$
- If $0 < \varepsilon_1 < \varepsilon_2$, then $v(P_{\varepsilon_2}) \le v(P_{\varepsilon_1})$
- If x^*_ε solves (P_ε) and $x^*_\varepsilon \in \Omega$, then x^*_ε is optimal also for (P)

Teoremi

- If f is coercive, then the sequence $\{x^k\}$ is bounded and any of its cluster points is an optimal solution of (P).
- If $\{x^k\}$ converges to x^*, then x^* is an optimal solution of (P).
- If $\{x^k\}$ converges to x^* and the gradients of active constraints at x^* are linear independent, then x^* is an optimal solution of (P) and the sequence of vectors $\{\lambda^k\}$ defined as

$$\lambda_i^k := \frac{2}{\varepsilon_k} \max\{0, g_i(x^k)\}, \qquad i = 1, \dots, m$$

converges to a vector λ^* of KKT multipliers associated to x^*.

Exact Penality Method (variante)

È una variante del panality method che non considera il quadrato.

Sia P un convex constrained problem

$$\begin{cases} \min\ f(x) \\ g_i(x) \le 0 \qquad \forall\, i = 1, \dots, m \end{cases}$$

definiamo la linear penalty function

$$\tilde{p}(x) = \sum_{i=1}^{m} \max\{0, g_i(x)\}.$$

allora il penalized problem è unconstrained, convex e nonsmooth

$$\begin{cases} \min \ f(x) + \dfrac{1}{\varepsilon}\,\widetilde{p}(x) \\ x \in \mathbb{R}^n \end{cases} \qquad (\tilde{P}_{\varepsilon_k})$$

Non abbiamo bisogno di una sequenza $\varepsilon_k \to 0$ per approssimare una optimal solution di P, questo ci permette di evitare problemi numerici.

0) Imposta $\varepsilon_0 > 0$, $\gamma \in (0,1)$, $k=0$

1) Trova una soluzione ottimale x^k del problema penalizzato $\tilde{P}_{\varepsilon_k}$

2) Se $x^k \in \Omega \Rightarrow$ Stop

Altrimenti $\varepsilon_{k+1} = \gamma\,\varepsilon_k$, $k = k+1$, vai allo step 1

Proposizione

Suppose that there exists an optimal solution x^* of (P) and λ^* is a KKT multipliers vector associated to x^*. Then, the sets of optimal solutions of (P) and $(\widetilde{P}_\varepsilon)$ coincide provided that $\varepsilon \in (0, 1/\|\lambda^*\|_\infty)$.

Teorema

The exact penalty method stops after a finite number of iterations at an optimal solution of (P).

Barrier Methods

Sia il problema

$$\begin{cases} \min \ f(x) \\ g(x) \leq 0 \end{cases}$$

dove

- f, g_i convessi e hanno deriv. seconda
- Non ci sono punti isolati in Ω
- Esiste una soluzione ottimale, cioè f è coercive o Ω è bounded
- La slater constraint qualification holds, cioè

  $\exists\, \bar{x}$ t.c $\bar{x} \in \operatorname{dom}(f)$ e $g_i(\bar{x}) < 0 \quad \forall i = 1 \dots m$

quindi la strong duality holds.

Special cases: linear problem, convex quadratic problem

Sotto queste condizioni il problema è equivalente all' unconstrained problem

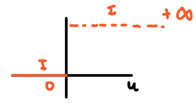
$$\begin{cases} \min \ f(x) + \sum_{i=1}^{m} I_-(g_i(x)) \\ x \in \mathbb{R}^n \end{cases}$$

dove

$$I_-(u) = \begin{cases} 0 & \text{if } u \leq 0 \\ +\infty & \text{if } u > 0 \end{cases}$$

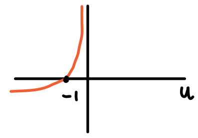
all' interno di Ω

All' esterno di Ω



$\downarrow$ Indicator function di $\mathbb{R}_-$, non è ne finita ne differenziabile la indicator function I_- può essere approssimata dalla smooth convex function

$$u \to \underline{-\varepsilon \log(-u)} \quad \text{dove } \varepsilon > 0$$



e l'approssimazione migliora come $\varepsilon \to 0$

Quindi possiamo approssimare l'unconstrained problem con

$$\begin{cases} \min \ f(x) - \varepsilon \sum_{i=1}^{m} \log(-g_i(x)) \\ x \in \text{int}(\Omega) \end{cases}$$

dove

$$B(x) = -\sum_{i=1}^{m} \log(-g_i(x))$$

is called logarithmic barrier function. It has the following properties:

- $\text{dom}(B) = \text{int}(\Omega)$
- B is convex
- B is smooth with

$$\nabla B(x) = -\sum_{i=1}^{m} \frac{1}{g_i(x)} \nabla g_i(x)$$

$$\nabla^2 B(x) = \sum_{i=1}^{m} \frac{1}{g_i(x)^2} \nabla g_i(x) \nabla g_i(x)^T + \sum_{i=1}^{m} \frac{1}{-g_i(x)} \nabla^2 g_i(x)$$

Se x^*_ε è la soluzione ottimale, allora:

$$\nabla f(x^*_\varepsilon) + \sum_{i=1}^{m} \frac{\varepsilon}{-g_i(x^*_\varepsilon)} \nabla g_i(x^*_\varepsilon) = 0$$

definiamo $\lambda^*_\varepsilon = \left(\frac{\varepsilon}{-g_1(x^*_\varepsilon)}, \dots, \frac{\varepsilon}{-g_m(x^*_\varepsilon)} \right) > 0 \Rightarrow L(x, \lambda^*_\varepsilon) = f(x) + \sum_{i=1}^{m} (\lambda^*_\varepsilon)_i g_i(x)$

$\downarrow$

convessa e $\nabla_x L(x^*_\varepsilon, \lambda^*_\varepsilon) = 0$

allora:

$$f(x^*_\varepsilon) \geq v(P) \geq \varphi(\lambda^*_\varepsilon) = \min_x L(x, \lambda^*_\varepsilon) = L(x^*_\varepsilon, \lambda^*_\varepsilon) = f(x^*_\varepsilon) - \underbrace{m\varepsilon}_{\text{optimality gap}}$$

Algorithm

0. Setto $\delta > 0$, $\tau < 1$, $\varepsilon_1 > 0$ e $k = 1$

   Scelgo $x^0 \in \text{int}(\Omega)$

1. Trovo la sol. ottimale x^k di:

$$\begin{cases} \min \ f(x) - \varepsilon_k \sum_{i=1}^{m} \log(-g_i(x)) \\ x \in \text{int}(\Omega) \end{cases}$$

   usando x^{k-1} come starting point

2. Se $m \varepsilon_k < \delta \Rightarrow$ Stop

   Se $\varepsilon_{k+1} = \tau \varepsilon_k$, $k = k+1$, vai allo step 1

La scelta di r implica un tradeoff:

a) piccolo $\Rightarrow$ meno outer iterations, più inner iterations

b) viceversa

Come trovare $x^0 \in \text{int}(\Omega)$?

Consider the auxiliary problem

$$\begin{cases} \min\limits_{x,s} \ s \\ g_i(x) \le s \end{cases}$$

- Take any $\tilde{x} \in \mathbb{R}^n$, find $\tilde{s} > \max\limits_{i=1,\dots,m} g_i(\tilde{x})$
  [$(\tilde{x}, \tilde{s})$ is in the interior of the feasible region of the auxiliary problem]
- Find an optimal solution (x^*, s^*) of the auxiliary problem using a barrier method starting from $(\tilde{x}, \tilde{s})$
- If $s^* < 0$ then $x^* \in \text{int}(\Omega)$
  else $\text{int}(\Omega) = \emptyset$

Multiobjective Optimization

In un multiobjective optimization problem, la obj function f è un vettore di p elementi $f(x) = [f_1(x), \dots, f_p(x)]$

Ci possono essere obiettivi che vanno in conflitto, abbiamo quindi bisogno di definire un ordine.

Pareto order

Dati $x, y \in \mathbb{R}^p$, diciamo che

$$x \ge y \iff x_i \ge y_i \quad \forall i = 1 \dots p$$

Questa relazione è un partial order in $\mathbb{R}^p$ ma non totale.

Ideal minimum: $x \in A$ t.c $y \ge x$ $\forall y \in A$

Minimum: $x \in A$ t.c $\nexists\, y \in A, y \ne x$ t.c $x \ge y$

Weak minimum: $x \in A$ t.c $\nexists\, y \in A, y \ne x$ t.c $x > y$

Proposizione

$IMin(A) \subseteq Min(A) \subseteq WMin(A)$.

If $IMin(A) \ne \emptyset$, then $IMin(A) = Min(A) = \{\bar{x}\}$.

Example. $A = \{x \in \mathbb{R}^2 : x_1 \ge 0, x_2 \ge 0, x_1 + x_2 \le 1\}$.
$IMin(A) = Min(A) = \{(0,0)\}$, $WMin(A) = \{x \in A : x_1 = 0 \text{ or } x_2 = 0\}$.

Example. $B = \{x \in \mathbb{R}^2 : 0 \le x_1 \le 2, 0 \le x_2 \le 2, x_1 + x_2 \ge 1\}$.
$IMin(B) = \emptyset$, $Min(B) = \{x \in B : x_1 + x_2 = 1\}$,
$WMin(B) = \{x \in B : x_1 = 0 \text{ or } x_2 = 0 \text{ or } x_1 + x_2 = 1\}$.

Dato un multiobjective opt. problem

$$\begin{cases} \min \; f(x) = (f_1(x), f_2(x), \dots, f_p(x)) \\ x \in \Omega \end{cases} \qquad (P)$$

abbiamo:

- $x^* \in \Omega$ is a Pareto ideal minimum of (P) if $f(x^*)$ is an Pareto ideal minimum of $f(\Omega)$, i.e., $f(x) \geq f(x^*)$ for any $x \in \Omega$.
- $x^* \in \Omega$ is a Pareto minimum of (P) if $f(x^*)$ is a Pareto minimum of $f(\Omega)$, i.e., if there is no $x \in \Omega$ such that

$$\begin{array}{ll} f_i(x^*) \geq f_i(x) & \text{for any } i = 1, \dots, p, \\ f_j(x^*) > f_j(x) & \text{for some } j \in \{1, \dots, p\}. \end{array}$$

- $x^* \in \Omega$ is a Pareto weak minimum of (P) if $f(x^*)$ is a Pareto weak minimum of $f(\Omega)$, i.e., if there is no $x \in \Omega$ such that

$$f_i(x^*) > f_i(x) \qquad \text{for any } i = 1, \dots, p.$$

Example. Consider

$$\begin{cases} \min \; (x_1 - x_2, \; -2x_1 + x_2) \\ x_1 \leq 1 \\ -x_1 \leq 0 \\ -x_1 + x_2 \leq 2 \\ 2x_1 - x_2 \leq 0 \end{cases} \qquad (P)$$

The image $f(\Omega) = \{(y_1, y_2) : \; y_1 = x_1 - x_2, \; y_2 = -2x_1 + x_2, \; x \in \Omega\}$.
We obtain $x_1 = -y_1 - y_2$ and $x_2 = -2y_1 - y_2$, hence
$f(\Omega) = \{(y_1, y_2) : \; -y_1 - y_2 \leq 1, \; y_1 + y_2 \leq 0, \; -y_1 \leq 2, \; -y_2 \leq 0\}$.
$IMin(f(\Omega)) = \emptyset$. $Min(f(\Omega)) = \{y \in f(\Omega) : \; -y_1 - y_2 = 1\}$, thus

$$\{\text{minima of (P)}\} = \{x \in \Omega : \; -x_1 + x_2 + 2x_1 - x_2 = 1\} = \{x \in \Omega : \; x_1 = 1\}.$$

$WMin(f(\Omega)) = \{y \in f(\Omega) : \; -y_1 - y_2 = 1 \text{ or } y_1 = -2 \text{ or } y_2 = 0\}$, thus

$$\{\text{weak minima of (P)}\} = \{x \in \Omega : \; x_1 = 1 \text{ or } x_1 - x_2 = -2 \text{ or } -2x_1 + x_2 = 0\}.$$

Existence Results

Per verificare l'esistenza abbiamo due teoremi

1) Generalized Weierstrass theorem

Se f_i è continua $\forall i = 1 \dots p$ e Ω è chiuso e limitato, allora esiste il minimo per (P)

2) Se f_i è continua $\forall i = 1 \dots p$ e Ω è chiuso e esistono $v \in \mathbb{R}$ e $j \in \{1 \dots p\}$ t.c il sublevel set $\{x \in \Omega : f_j(x) \leq v\}$ è non vuoto e limitato allora esiste il minimo di P

Corollario

Se f_i è continua $\forall i = 1 \dots p$, Ω è chiusa e f_j è coercive $\forall j = 1 \dots p$, allora esiste un minimo di P

Optimality Conditions

1) $x^* \in \Omega$ è un minimo di P $\Leftrightarrow$ il problema di ottimizzazione ausiliario seguente

$$\begin{cases} \max \sum_{i=1}^{p} \varepsilon_i \\ f_i(x) + \varepsilon_i \leq f_i(x^*) \qquad \forall \, i = 1, \dots, p \\ x \in \Omega \\ \varepsilon \geq 0 \end{cases}$$

ha un valore ottimo uguale a 0

2) $x^* \in \Omega$ è un weak minimum di (P) $\Leftrightarrow$ il problema di ottimizzazione ausiliario seguente

$$\begin{cases} \max \ v \\ v \le \varepsilon_i & \forall \ i = 1, \dots, p \\ f_i(x) + \varepsilon_i \le f_i(x^*) & \forall \ i = 1, \dots, p \\ x \in \Omega \\ \varepsilon \ge 0 \end{cases}$$

ha un valore ottimo uguale a 0

Unconstrained Problem Opt. conditions (First Order)

Sia il problema un unconstrained multiobjective problem

$$\begin{cases} \min \ f(x) = (f_1(x), f_2(x), \dots, f_p(x)) \\ x \in \mathbb{R}^n \end{cases}$$ dove f_i è continua diff. $\forall i$

Condizione Necessaria

Se x^* è un weak minimum di (P) $\Rightarrow$ $\exists \ \xi^* \in \mathbb{R}^p$ t.c

$$\begin{cases} \sum_{i=1}^{p} \xi_i^* \nabla f_i(x^*) = 0 \\ \xi^* \ge 0, \quad \sum_{i=1}^{p} \xi_i^* = 1 \end{cases} \qquad (S)$$

Condizione Sufficente

Se (P) è convesso, cioè f_i è convessa $\forall i = 1 \dots p$, e (x^*, ξ^*) è una soluzione di (S) allora x^* è un weak minimum di (P)

Constrained Problem Opt. conditions (First Order)

Sia il problema un Constrained multiobjective problem

$$\begin{cases} \min \ f(x) = (f_1(x), f_2(x), \dots, f_p(x)) \\ g_j(x) \le 0 \quad \forall \ j = 1, \dots, m \\ h_k(x) = 0 \quad \forall \ k = 1, \dots, q \end{cases}$$ dove f_i, g_j, h_k sono continue diff. $\forall i, j, k$

Condizione Necessaria

Se x^* è un weak minimum di (P) e ACQ holds in x^*,

allora $\exists\, \xi^* \in \mathbb{R}^p$, $\lambda^* \in \mathbb{R}^m$ e $\mu^* \in \mathbb{R}^q$ t.c $(x^*, \xi^*, \lambda^*, \mu^*)$ risolve il KTT system

$$\begin{cases} \sum_{i=1}^{p} \xi_i^* \nabla f_i(x^*) + \sum_{j=1}^{m} \lambda_j^* \nabla g_j(x^*) + \sum_{k=1}^{q} \mu_k^* \nabla h_k(x^*) = 0 \\ \xi^* \geq 0, \quad \sum_{i=1}^{p} \xi_i^* = 1 \\ \lambda^* \geq 0 \\ \lambda_j^* g_j(x^*) = 0 \qquad \forall\, j = 1, \dots, m \end{cases}$$

Condizione Sufficente

Se (P) è convesso, cioè f_i convex, g_j convex e h_k affine, e $(x^*, \xi^*, \lambda^*, \mu^*)$

risolve il KTT system,

allora x^* è un weak minimum di (P)

Scalarization Method (per risolvere i multiobj problems)

Definisco un vettore di pesi associati alle obj functions

$$\alpha = (\alpha_1, \dots, \alpha_p) \geq 0 \quad \text{such that} \quad \sum_{i=1}^{p} \alpha_i = 1$$

e considero il seguente scalar optimization problem

$$\begin{cases} \min \sum_{i=1}^{p} \alpha_i f_i(x) \\ x \in \Omega \end{cases} \qquad (P_\alpha)$$

Sia S_α l'insieme delle soluzione ottime per P_α

Teorema

1) $\bigcup_{\alpha \geq 0} S_\alpha \subseteq \{\text{weak minima of (P)}\}$

2) $\bigcup_{\alpha > 0} S_\alpha \subseteq \{\text{minima of (P)}\}$

l'unione degli S_α sono un sottoinsieme dei weak minimum e dei minimum di P

Quindi risolvere P_α $\forall \alpha$ possibile non permette di trovare tutti i minimi o i weak minimum.

Theorem

1) Se P è lineare $\Rightarrow$ a) $WMIN(P) = \bigcup_{\alpha \geq 0} S_\alpha$

b) $MIN(P) = \bigcup_{\alpha > 0} S_\alpha$

2) Se P è convesso $\Rightarrow$ $WMIN(P) = \bigcup_{\alpha \geq 0} S_\alpha$

3) Se P è convesso e f_i strongly convex $\forall i$ $\Rightarrow$ $MIN(P) = WMIN(P) = \bigcup_{\alpha \geq 0} S_\alpha$

Goal Method

Nello objective space $\mathbb{R}^p$ definire un punto ideale z come

$$z_i = \min_{x \in \Omega} f_i(x), \qquad \forall\, i = 1, \dots, p.$$

Essendo che spesso (P) non ha un minimo ideale, cioè $z \notin f(\Omega)$, vogliamo trovare un punto di $f(\Omega)$ il quale è il più vicino possibile a z

$$\begin{cases} \min \ \|f(x) - z\|_s \\ x \in \Omega \end{cases} \qquad \text{with} \quad s \in [1, +\infty]. \qquad (G)$$

Teorema

1) Se $s \in [1, +\infty)$, allora ogni optimal solution di G è un minimo di P

2) Se $s = +\infty$, allora ogni optimal solution di G è un weak minimum di P

Sia P un linear multiobj opt. problem, cioè

$$\begin{cases} \min \ Cx \\ Ax \leq b \end{cases} \qquad (P) \qquad \text{dove } C \text{ è una matrice } p \times n$$

1) Se $s=2$, G è equivalente a un quadratic Programming problem

$$\begin{cases} \min \frac{1}{2}\|Cx-z\|_2^2 = \frac{1}{2}x^T C^T C x - x^T C^T z + \frac{1}{2}z^T z \\ Ax \le b \end{cases}$$

2) Se $s=1$, G è equivalente a un linear programming problem

$$\begin{cases} \min\limits_{x,y} \sum\limits_{i=1}^{p} y_i \\ y_i \ge C_i x - z_i & \forall\, i=1,\dots,p \\ y_i \ge z_i - C_i x & \forall\, i=1,\dots,p \\ Ax \le b \end{cases}$$

3) Se $s=+\infty$, allora G è equivalente a un linear programming problem

$$\begin{cases} \min\limits_{x,y} \; y \\ y \ge C_i x - z_i & \forall\, i=1,\dots,p \\ y \ge z_i - C_i x & \forall\, i=1,\dots,p \\ Ax \le b \end{cases}$$

Non Cooperative game theory

La game theory è l'analisi di situazioni conflittuali tra differenti decisioni fatte dai players che hanno interessi differenti.

Una decisione (strategia) di ciascun player può produrre risultati differenti dipendentemente dalle strategie scelte dagli altri players.

Un noncooperative game in forma normale è dato da un insieme di N players, ogni player i ha un insieme Ω_i di strategie e una cost function $f_i : \Omega_1 \times \dots \times \Omega_N \to \mathbb{R}$

L'obiettivo di ogni player i consiste nel risolvere l'opt. problem

$$\begin{cases} \min \; f_i(x_1, x_2, \dots, x_{i-1}, x_i, x_{i+1}, \dots, x_N) \\ x_i \in \Omega_i \end{cases}$$

Da ora in poi consideremo solo noncooperative games tra due players:

Player 1: $\begin{cases} \min \; f_1(x,y) \\ x \in X \end{cases}$ Player 2: $\begin{cases} \min \; f_2(x,y) \\ y \in Y \end{cases}$

Si ha una soluzione di equilibrio quando la coppia $(\bar{x}, \bar{y})$ di strategie é t.c nessuno dei due player può diminuire il suo costo di una unilateral Deviation.

$$f_1(\bar{x}, \bar{y}) = \min_{x \in X} f_1(x, \bar{y}), \qquad f_2(\bar{x}, \bar{y}) = \min_{y \in Y} f_2(\bar{x}, y).$$

In altre parole, $(\bar{x}, \bar{y})$ è un Nash equilibrium sse

1) $\bar{x}$ è la miglior risposta del player 1 alla strategia $\bar{y}$ del player 2
2) $\bar{y}$ è la miglior risposta del player 2 alla strategia $\bar{x}$ del player 1

Matrix Game

Un matrix game è un 2-person noncooperative game dove:

1) X e Y sono due set finiti: $X = \{1 \dots m\}$, $Y = \{1 \dots n\}$
2) $f_2 = -f_1$ (zero sum game) → le due cost function sono opposte

Può essere rappresentato da una matrice $m \times n$, dove c_{ij} è l'ammontare di soldi che il player 1 paga al player 2 se P1 sceglie la strategia i e P2 sceglie la strategia j

		Player 2		
		1	2	3
Player 1	1	1	−1	0
	2	3	−2	−1
	3	2	3	−2

Qual è il Nash equilibrium del matrix game sopra descritto?

Passo 1) For player 2, strategy 3 is worse than strategy 1 because his/her profit is less than the one obtained playing strategy 1 for any strategy of player 1. Hence, player 2 will never choose strategy 3, which can be deleted from the game. The game is equivalent to

		Player 2	
		1	2
Player 1	1	1	−1
	2	3	−2
	3	2	3

Passo 2) Now, for player 1 strategy 3 is worse than strategy 1.

The reduced game is

		Player 2	
		1	2
Player 1	1	1	−1
	2	3	−2

Passo 3) For player 2, strategy 2 is worse than strategy 1. Thus, player 2 will always choose strategy 1. The reduced game is

		Player 2
		1
Player 1	1	1
	2	3

Passo 4) Finally, for player 1, strategy 2 is worse than strategy 1. Therefore, player 1 will always choose strategy 1.
Hence $(1, 1)$ is a Nash equilibrium.

		Player 2
		1
Player 1	1	1

Il procedimento svolto sopra, nella teoria stiamo eliminando le "strictly dominated strategies", formalmente:

1) $x \in X$ è strictly dominated da $\tilde{x} \in X$ se $f_1(x,y) > f_1(\tilde{x},y) \quad \forall y \in Y$

2) $y \in Y$ è strictly dominated da $\tilde{y} \in Y$ se $f_2(x,y) > f_2(x,\tilde{y}) \quad \forall x \in X$

Esempi

		Player 2				
		1	2	3	4	5
Player 1	1	1	−1	1	−2	−3
	2	2	−2	3	4	0
	3	1	0	1	−3	−4
	4	4	−3	2	−1	−1
	5	5	−2	4	−3	2

$\longrightarrow$

		Player 2				
		1	2	3	4	5
Player 1	1	1		1		
	2					
	3	1		1		
	4					
	5					

NASH equilibrium

		Player 2	
		1 (odd)	2 (even)
Player 1	1 (odd)	1	−1
	2 (even)	−1	1

$\longrightarrow$ Non ha nè strictly dominated strategies nè Nash Equilibria

Mixed Strategy

Una mixed strategy è un vettore di probabilità.

Per il player-1 è un m-vector e $X = \{ x \in \mathbb{R}^m : x \geq 0, \sum_{i=1}^{m} x_i = 1 \}$ è l'insieme delle mixed strategies

Per il player-2 è un n-vector e $Y = \{ y \in \mathbb{R}^n : y \geq 0, \sum_{j=1}^{n} y_j = 1 \}$ è l'insieme delle mixed strategies gli vertici di X e Y, cioè $e_i = (0, \dots, 0, 1, 0 \dots, 0)$ e $e_j = (0, \dots, 0, 1, 0, \dots, 0)$, sono le pure strategies.

gli expected costs sono $f_1(x,y) = x^T C y$ (P_1) e $f_2(x,y) = -x^T C y$ (P_2)

Mixed Strategy Nash Equilibria

$(\bar{x}, \bar{y}) \in X \times Y$ è una mixed strategies Nash equilibrium se

$$\max_{y \in Y} \bar{x}^T C y = \bar{x}^T C \bar{y} = \min_{x \in X} x^T C \bar{y},$$

cioè $(\bar{x}, \bar{y})$ è un saddle point della funzione $x^T C y$

$(\bar{x}, \bar{y})$ è un mixed strategies Nash equilibrium sse:

$$\begin{cases} \bar{x} \text{ is an optimal solution of } \min\limits_{x \in X} \max\limits_{y \in Y} x^T C y \\ \bar{y} \text{ is an optimal solution of } \max\limits_{y \in Y} \min\limits_{x \in X} x^T C y \end{cases}$$

TEO:

1) Il problema $\min\limits_{x \in X} \max\limits_{y \in Y} x^T C y$ è equivalente al linear programming problem

$$\begin{cases} \min \ v \\ v \geq \sum\limits_{i=1}^{m} c_{ij} x_i \quad \forall \, j = 1, \dots, n \\ x \geq 0, \quad \sum\limits_{i=1}^{m} x_i = 1 \end{cases} \qquad (P_1)$$

2) Il problema $\max\limits_{y \in Y} \min\limits_{x \in X} x^T C y$ è equivalente al linear programming problem

$$\begin{cases} \max \ w \\ w \leq \sum\limits_{j=1}^{n} c_{ij} y_j \quad \forall \, i = 1, \dots, m \\ y \geq 0, \quad \sum\limits_{j=1}^{n} y_j = 1 \end{cases} \qquad (P_2)$$

3) P_2 è il duale di P_1

Corollario: Ogni matrix game ha almeno un mixed strategies Nash equilibrium

esempio

Example. (Odds and evens)

		Player 2: 1 (odd)	Player 2: 2 (even)
Player 1	1 (odd)	1	−1
	2 (even)	−1	1

$$(P_1)\begin{cases}\min\ v\\ v \geq x_1 - x_2\\ v \geq -x_1 + x_2\\ x \geq 0\\ x_1 + x_2 = 1\end{cases} \text{ is equivalent to } \begin{cases}\min\ v\\ v \geq 2x_1 - 1\\ v \geq 1 - 2x_1\\ 0 \leq x_1 \leq 1\end{cases} \Rightarrow \bar{x} = (1/2, 1/2)$$

$$(P_2)\begin{cases}\max\ w\\ w \leq y_1 - y_2\\ w \leq -y_1 + y_2\\ y \geq 0\\ y_1 + y_2 = 1\end{cases} \text{ is equivalent to } \begin{cases}\max\ w\\ w \leq 2y_1 - 1\\ w \leq 1 - 2y_1\\ 0 \leq y_1 \leq 1\end{cases} \Rightarrow \bar{y} = (1/2, 1/2)$$

Bimatrix Games

Un Bimatrix game è two-person noncooperative game dove:

1) l'insieme delle pure strategies sono finite, quindi l'insieme dei sets di mixed strategies sono

$$X = \{x \in \mathbb{R}^m : \ x \geq 0, \quad \textstyle\sum_{i=1}^m x_i = 1\}$$
$$Y = \{y \in \mathbb{R}^n : \ y \geq 0, \quad \textstyle\sum_{j=1}^n y_j = 1\}$$

2) $f_2 \neq f_1$ (non-zero sum game), le cost function sono:
- $f_1(x,y) = x^T C_1 y$
- $f_2(x,y) = x^T C_2 y$

TEO: Un bimatrix game ha almeno un mixed strategies Nash Equilibrium

TEO: Se definiamo i best response mappings $B_1 : Y \to X$ e $B_2 : X \to Y$ come

$$B_1(y) = \left\{\text{optimal solutions of } \min_{x \in X}\ x^T C_1 y\right\},$$
$$B_2(x) = \left\{\text{optimal solutions of } \min_{y \in Y}\ x^T C_2 y\right\},$$

allora $(\bar{x}, \bar{y})$ è un Nash eq. $\Leftrightarrow$ $\bar{x} \in B_1(\bar{y})$ e $\bar{y} \in B_2(\bar{x})$

i nash equilibria sono dati dall'intersezione dei grafi delle Best response mappings B_1 e B_2.

KKT conditions per Nash Equilibria

$(\bar{x},\bar{y})$ è un Nash equilibrium $\Leftrightarrow$ esistono $\mu_1, \mu_2 \in \mathbb{R}$ t.c

$$\begin{cases} C_1\bar{y} + \mu_1 e \geq 0 \\ \bar{x} \geq 0, \quad \sum_{i=1}^{m} \bar{x}_i = 1 \\ \bar{x}_i(C_1\bar{y} + \mu_1 e)_i = 0 \quad \forall\, i = 1,\dots,m \\ C_2^{\mathsf{T}}\bar{x} + \mu_2 e \geq 0 \\ \bar{y} \geq 0, \quad \sum_{j=1}^{n} \bar{y}_j = 1 \\ \bar{y}_j(C_2^{\mathsf{T}}\bar{x} + \mu_2 e)_j = 0 \quad \forall\, j = 1,\dots,n \end{cases}$$

dove $e = (1,\dots,1)^{\mathsf{T}}$.

Assumiamo che $C_1 < 0$ e $C_2 < 0$:

1) Se $(\bar{x},\bar{y})$ è un nash eq. $\exists\, u>0, v>0$ t.c $\tilde{x} = \bar{x}/u$ e $\tilde{y} = \bar{y}/v$ risolvono:

$$\begin{cases} \tilde{x} \geq 0, \quad C_1\tilde{y} + e \geq 0, \quad \tilde{x}_i(C_1\tilde{y} + e)_i = 0 \quad \forall\, i = 1,\dots,m \\ \tilde{y} \geq 0, \quad C_2^{\mathsf{T}}\tilde{x} + e \geq 0, \quad \tilde{y}_j(C_2^{\mathsf{T}}\tilde{x} + e)_j = 0 \quad \forall\, j = 1,\dots,n \end{cases} \qquad (S)$$

2) Se $(\tilde{x},\tilde{y})$ risolve (S), con $\tilde{x} \neq 0$ e $\tilde{y} \neq 0$ $\rightarrow$ $\left(\dfrac{\tilde{x}}{\sum_{i=1}^{m} \tilde{x}_i}, \dfrac{\tilde{y}}{\sum_{j=1}^{n} \tilde{y}_j} \right)$ è un nash eq.

Define the polyhedra

$$P = \left\{ (x_1,\dots,x_m) : \begin{array}{ll} x_i \geq 0 & \forall\, i = 1,\dots,m \\ (C_2^{\mathsf{T}}x + e)_j \geq 0 & \forall\, j = m+1,\dots,m+n \end{array} \right\}$$

$$Q = \left\{ (y_{m+1},\dots,y_{m+n}) : \begin{array}{ll} (C_1 y + e)_i \geq 0 & \forall\, i = 1,\dots,m \\ y_j \geq 0 & \forall\, j = m+1,\dots,m+n \end{array} \right\}$$

Theorem

- $(\tilde{x},\tilde{y})$ solves system (S) if and only if $\tilde{x} \in P$, $\tilde{y} \in Q$ and for any $k = 1,\dots,m+n$ either the k-th constraint of P is active in $\tilde{x}$ or the k-th constraint of Q is active in $\tilde{y}$.
- If the vertices of P and Q are non-degenerate and $(\tilde{x},\tilde{y})$ solves system (S), then $\tilde{x}$ is a vertex of P and $\tilde{y}$ is a vertex of Q.

Therefore, if $C_1 < 0$, $C_2 < 0$ and vertices of P and Q are non-degenerate, then we can find all the Nash equilibria analyzing all the pairs (x,y) of vertices of P and Q, checking if for any $k = 1,\dots,m+n$ either the k-th constraint of P is active in x or the k-th constraint of Q is active in y.

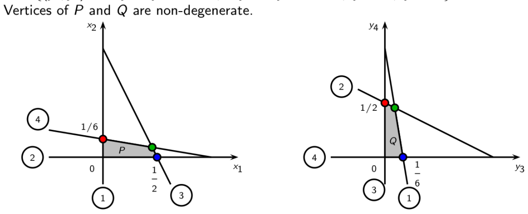
Example. (Battle of the buddies)

$$C_1 = \begin{pmatrix} -5 & 0 \\ 0 & -1 \end{pmatrix} \qquad C_2 = \begin{pmatrix} -1 & 0 \\ 0 & -5 \end{pmatrix}$$

Since the elements of C_1 and C_2 are not all negative, we can reformulate the game setting $(C_1)_{ij} = (C_1)_{ij} - 1$ and $(C_2)_{ij} = (C_2)_{ij} - 1$:

$$C_1 = \begin{pmatrix} -6 & -1 \\ -1 & -2 \end{pmatrix} \qquad C_2 = \begin{pmatrix} -2 & -1 \\ -1 & -6 \end{pmatrix}$$

$P = \{(x_1,x_2) :\ x_1 \geq 0,\ x_2 \geq 0,\ -2x_1 - x_2 + 1 \geq 0,\ -x_1 - 6x_2 + 1 \geq 0\}$
$Q = \{(y_3,y_4) :\ -6y_3 - y_4 + 1 \geq 0,\ -y_3 - 2y_4 + 1 \geq 0,\ y_3 \geq 0,\ y_4 \geq 0\}$
Vertices of P and Q are non-degenerate.



Solutions of system (S)	Nash equilibria
$x = (0, 1/6)$ (active const. 1, 4) $y = (0, 1/2)$ (active const. 2, 3)	$x = (0,1)$ $y = (0,1)$
$x = (5/11, 1/11)$ (active const. 3, 4) $y = (1/11, 5/11)$ (active const. 1, 2)	$x = (5/6, 1/6)$ $y = (1/6, 5/6)$
$x = (1/2, 0)$ (active const. 2, 3) $y = (1/6, 0)$ (active const. 1, 4)	$x = (1,0)$ $y = (1,0)$

Lemke - Howson Algorithm

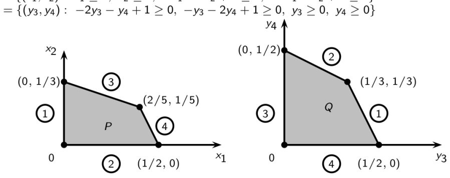
When $C_1 < 0$, $C_2 < 0$ and vertices of P and Q are non-degenerate, a Nash equilibrium can be found by using the Lemke-Howson algorithm:

1. Set $x = 0$, $y = 0$. Define $I_x = \{1, \dots, m\}$, $I_y = \{m+1, \dots, m+n\}$. Choose an exiting index $h \in \{1, \dots, m\}$.
2. In the polyhedron P, move from x along the edge given by constraints in $I_x \setminus \{h\}$ to the adjacent vertex x'.
3. Find the entering index k corresponding to the new active constraint at x'. Set $x = x'$ and $I_x = (I_x \setminus \{h\}) \cup \{k\}$.
4. If $k \notin I_y$ then STOP: (x, y) solves (S) $\rightarrow$ find a Nash equilibrium. else set $h = k$ and in the polyhedron Q move from y along the edge given by constraints in $I_y \setminus \{h\}$ to the adjacent vertex y'.
5. Find the entering index k corresponding to the new active constraint at y'. Set $y = y'$ and $I_y = (I_y \setminus \{h\}) \cup \{k\}$.
6. If $k \notin I_x$ then STOP: (x, y) solves (S) $\rightarrow$ find a Nash equilibrium. else set $h = k$ and go to step 2.

Example. Consider the following bimatrix game: $C_1 = \begin{pmatrix} -2 & -1 \\ -1 & -2 \end{pmatrix}$ $C_2 = \begin{pmatrix} -1 & -2 \\ -3 & -1 \end{pmatrix}$

$P = \{(x_1, x_2): \ x_1 \geq 0,\ x_2 \geq 0,\ -x_1 - 3x_2 + 1 \geq 0,\ -2x_1 - x_2 + 1 \geq 0\}$

$Q = \{(y_3, y_4): \ -2y_3 - y_4 + 1 \geq 0,\ -y_3 - 2y_4 + 1 \geq 0,\ y_3 \geq 0,\ y_4 \geq 0\}$



Iter	h	k	x	I_x	y	I_y
1			$(0,0)$	$\{1,2\}$	$(0,0)$	$\{3,4\}$
2	1	4	$(1/2,0)$	$\{2,4\}$	$(0,0)$	$\{3,4\}$
3	4	2	$(1/2,0)$	$\{2,4\}$	$(0,1/2)$	$\{2,3\}$
4	2	3	$(2/5,1/5)$	$\{3,4\}$	$(0,1/2)$	$\{2,3\}$
5	3	1	$(2/5,1/5)$	$\{3,4\}$	$(1/3,1/3)$	$\{1,2\}$
	Nash	equil.	$(2/3,1/3)$		$(1/2,1/2)$	

Convex Games