Ricerca Operativa

Ottimizzazione non vincolata

Prof. Paolo Brandimarte
Dip. di Scienze Matematiche – Politecnico di Torino
e-mail: paolo.brandimarte@polito.it
URL: staff.polito.it/paolo.brandimarte

Questa versione: 18 aprile 2023

Contenuto

Le slide seguenti sono tratte dal capitolo 5 di: P. Brandimarte, *Ottimizzazione per la Ricerca Operativa*, CLUT 2022.

- Condizioni di ottimalità del primo e del secondo ordine.
- Metodi numerici per il caso differenziabile:
  - gradiente;
  - Newton;
  - trust region.
- Metodi derivative-free e black-box.
- Funzioni di penalità.
- Ottimizzazione nonsmooth.

Facciamo riferimento a un problema di minimo,

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}), \tag{1}$$

e assumiamo che la funzione f sia sufficientemente differenziabile.

Definizione: minimo globale e locale. Dato il problema (1), diciamo che $\mathbf{x}^* \in \mathbb{R}^n$ è un punto di minimo globale se $f(\mathbf{x}^*) \leq f(\mathbf{x})$ per ogni $\mathbf{x} \in \mathbb{R}^n$. Ci riferiamo al valore $f(\mathbf{x}^*)$ come al *minimo* e a volte usiamo la notazione

$$\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}),$$

dove non si dà per scontato che il *punto di minimo* sia unico. Se esiste un vicinato $N_\epsilon(\mathbf{x}^*)$ tale che $f(\mathbf{x}^*) \leq f(\mathbf{x})$ per ogni $\mathbf{x} \in N_\epsilon(\mathbf{x}^*)$, parleremo di punto di minimo locale. Se le disuguaglianze sono strette per ogni $\mathbf{x} \neq \mathbf{x}^*$, su $\mathbb{R}^n$ o nel vicinato, parleremo di punto di minimo stretto (globale o locale).

In assenza di vincoli, l'intuizione e le conoscenze di base suggeriscono la condizione $f'(x) = 0$, per il caso di una sola variabile. Dovremmo anche verificare condizioni circa la derivata del secondo ordine, a meno che la funzione non sia convessa.

Il controesempio $\min_{x\in\mathbb{R}} e^x$ illustra un caso di funzione convessa e differenziabile, per il quale tuttavia il minimo non esiste.

Il teorema di Weierstrass garantisce l'esistenza di minimi e massimi per funzioni continue su un insieme compatto, ovvero chiuso e limitato. Se abbiamo un insieme aperto, o un problema non vincolato, questo non è garantito.

Possiamo, in questi casi, chiedere una proprietà diversa della funzione, ovvero la coercività:

$$\lim_{\|\mathbf{x}\|\to\infty} f(\mathbf{x}) = +\infty.$$

Teorema: condizione necessaria del primo ordine. Se $\mathbf{x}^*$ è un punto di minimo locale per il problema non vincolato $\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$, dove $f \in \mathcal{C}^1$, allora $\nabla f\left(\mathbf{x}^*\right) = \mathbf{0}_n$.

La condizione del primo ordine è necessaria, ma non sufficiente. Essa diventa necessaria e sufficiente per l'ottimalità globale nel caso convesso differenziabile.

Teorema: condizione necessaria del secondo ordine. Se $\mathbf{x}^*$ è un punto di minimo locale per il problema non vincolato $\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$, dove $f \in \mathcal{C}^2$, allora $\nabla^2 f\left(\mathbf{x}^*\right) \in \mathbb{S}^n_+$.

Teorema: condizione sufficiente del secondo ordine. Se $\mathbf{x}^*$ è un punto di stazionarietà per la funzione $f \in \mathcal{C}^2$, e $\nabla^2 f\left(\mathbf{x}^*\right) \in \mathbb{S}^n_{++}$, allora $\mathbf{x}^*$ è un punto di minimo locale.

I due teoremi forniscono rispettivamente una condizione relativamente debole e quindi solo necessaria, e una sufficiente, e quindi anche più forte del necessario.

Un tipico esempio di ottimizzazione convessa non vincolata è il problema dei minimi quadrati,

$$\min \left\|\mathbf{A}\mathbf{x}-\mathbf{b}\right\|_2^2,$$

dove $\mathbf{A} \in \mathbb{R}^{n\times p}$, $p < n$, e si cerca la proiezione di $\mathbf{b} \in \mathbb{R}^n$ sul sottospazio lineare generato dalla combinazione lineare delle p colonne di $\mathbf{A}$.

Per verificare la convessità, scriviamo:

$$\big(\mathbf{A}\mathbf{x}-\mathbf{b}\big)^{\mathsf{T}}\big(\mathbf{A}\mathbf{x}-\mathbf{b}\big) = \mathbf{x}^{\mathsf{T}}\mathbf{A}^{\mathsf{T}}\mathbf{A}\mathbf{x} - 2\mathbf{b}^{\mathsf{T}}\mathbf{A}\mathbf{x} + \mathbf{b}^{\mathsf{T}}\mathbf{b}.$$

La funzione è convessa, poiché $\mathbf{A}^{\mathsf{T}}\mathbf{A} \in \mathbb{S}^p_+$. Per vedere ciò basta introdurre la variabile $\mathbf{y} = \mathbf{A}\mathbf{x}$ e scrivere

$$\mathbf{x}^{\mathsf{T}}\mathbf{A}^{\mathsf{T}}\mathbf{A}\mathbf{x} = \mathbf{y}^{\mathsf{T}}\mathbf{y} = \|\mathbf{y}\|_2^2.$$

Applichiamo quindi la condizione di ottimalità del primo ordine:

$$2\mathbf{A}^{\mathsf{T}}\mathbf{A}\mathbf{x} - 2\mathbf{A}^{\mathsf{T}}\mathbf{b} = \mathbf{0} \quad \Rightarrow \quad \mathbf{x}^* = \left[\mathbf{A}^{\mathsf{T}}\mathbf{A}\right]^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{b}$$

Tale scrittura è valida a patto che la matrice $\mathbf{A}^{\mathsf{T}}\mathbf{A}$ sia invertibile.

Questo però richiede la condizione più forte $\mathbf{A}^{\mathsf{T}}\mathbf{A} \in \mathbb{S}^p_{++}$, che è garantita se le colonne di $\mathbf{A}$ sono linearmente indipendenti.

Se ciò non accade, un modo per ovviare alla difficoltà è ricorrere a una forma di **regolarizzazione**:

$$\min \big\|\mathbf{Ax}-\mathbf{b}\big\|_2^2+\gamma\big\|\mathbf{x}\big\|_2^2,$$

dove $\gamma>0$. La soluzione del problema è:

$$\big(\mathbf{Ax}-\mathbf{b}\big)^{\mathsf{T}}\big(\mathbf{Ax}-\mathbf{b}\big)+\gamma\mathbf{x}^{\mathsf{T}}\mathbf{x}=\mathbf{x}^{\mathsf{T}}\big[\mathbf{A}^{\mathsf{T}}\mathbf{A}+\gamma\mathbf{I}_p\big]\mathbf{x}-2\mathbf{b}^{\mathsf{T}}\mathbf{Ax}+\mathbf{b}^{\mathsf{T}}\mathbf{b},$$

dove $\mathbf{I}_p$ è la matrice identità di ordine p, e la condizione del primo ordine diventa

$$\mathbf{x}^*=\big[\mathbf{A}^{\mathsf{T}}\mathbf{A}+\gamma\mathbf{I}_p\big]^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{b}.$$

Il coefficiente $\gamma>0$ sposta gli autovalori della matrice Gramiana, rendendola positiva definita e quindi invertibile.

Una regressione con penalità l_2 è nota come regressione **ridge**.

Nel caso si penalizzi con norma l_1, si ottiene la regressione **lasso**, che tende anche a operare una selezione delle variabili esplicative (regressione sparsa), al prezzo di una non differenziabilità.

La condizione di stazionarietà $\nabla f(\mathbf{x}) = \mathbf{0}_n$ si traduce, nel caso generale, in un sistema di n equazioni non lineari in n incognite, che di regola non vengono risolte direttamente.

I metodi iterativi per la programmazione non lineare si basano su una successione di punti che migliorano l'obiettivo.

Dato il punto corrente $\mathbf{x}^\circ$, si cerca una direzione di discesa $\mathbf{d}$, lungo la quale

$$f\big(\mathbf{x}^\circ + \alpha\mathbf{d}\big) < f\big(\mathbf{x}^\circ\big),$$

per un passo $\alpha > 0$.

Si fa comunemente ricorso al gradiente della funzione obiettivo e ai gradienti delle funzioni che descrivono i vincoli, per caratterizzare le direzioni di discesa da un lato, e quelle ammissibili dall'altro.

L'idea è caratterizzare il comportamento di una funzione non lineare mediante la sua linearizzazione, ma non è sempre garantito che tale idea funzioni.

Teorema: direzione di discesa. Supponiamo che $f : \mathbb{R}^n \to \mathbb{R}$ sia una funzione differenziabile in $\mathbf{x}^\circ$. Se esiste un vettore $\mathbf{d} \in \mathbb{R}^n$ tale che $\nabla f(\mathbf{x}^\circ)^\mathsf{T}\mathbf{d} < 0$, allora esiste un $\delta > 0$ tale che

$$f(\mathbf{x}^\circ + \lambda\mathbf{d}) < f(\mathbf{x}^\circ), \qquad \forall \lambda \in (0, \delta),$$

e diremo che $\mathbf{d}$ è una direzione di discesa di f in $\mathbf{x}^\circ$.

Dimostrazione. La differenziabilità di f implica

$$f(\mathbf{x}^\circ + \lambda\mathbf{d}) = f(\mathbf{x}^\circ) + \lambda\nabla f(\mathbf{x}^\circ)^\mathsf{T}\mathbf{d} + \lambda\|\mathbf{d}\| \cdot \alpha(\mathbf{x}^\circ; \lambda\mathbf{d}),$$

dove $\alpha(\mathbf{x}^\circ; \lambda\mathbf{d}) \to 0$ per $\lambda \to 0$, che possiamo riscrivere come

$$\frac{f(\mathbf{x}^\circ + \lambda\mathbf{d}) - f(\mathbf{x}^\circ)}{\lambda} = \nabla f(\mathbf{x}^\circ)^\mathsf{T}\mathbf{d} + \|\mathbf{d}\| \cdot \alpha(\mathbf{x}^\circ; \lambda\mathbf{d}).$$

Dato che $\nabla f(\mathbf{x}^\circ)^\mathsf{T}\mathbf{d} < 0$ e $\alpha(\mathbf{x}^\circ; \lambda\mathbf{d}) \to 0$, avremo che esiste $\delta > 0$ per cui

$$\nabla f(\mathbf{x}^\circ)^\mathsf{T}\mathbf{d} + \|\mathbf{d}\| \cdot \alpha(\mathbf{x}^\circ; \lambda\mathbf{d}) < 0, \quad \forall \lambda \in (0, \delta),$$

da cui segue il risultato. □

Teorema: direzione di massima discesa. Supponiamo che $f : \mathbb{R}^n \to \mathbb{R}$ sia una funzione differenziabile in $\mathbf{x}$ e che $\nabla f(\mathbf{x}) \neq \mathbf{0}$. Allora, la direzione di massima discesa (steepest descent) è data dal gradiente cambiato di segno. Più formalmente, la soluzione del problema

$$\begin{aligned} \min \quad & f'(\mathbf{x}; \mathbf{d}) \\ \text{s.t.} \quad & \|\mathbf{d}\|_2 = 1, \end{aligned}$$

è data da $\mathbf{d}^* = -\nabla f(\mathbf{x})/\|\nabla f(\mathbf{x})\|_2$.

Dimostrazione. La differenziabilità di f in $\mathbf{x}$ permette di scrivere

$$f'(\mathbf{x}; \mathbf{d}) = \lim_{\alpha \downarrow 0} \frac{f(\mathbf{x} + \alpha \mathbf{d}) - f(\mathbf{x})}{\alpha} = \nabla f(\mathbf{x})^{\mathsf{T}} \mathbf{d},$$

che vogliamo minimizzare con sotto il vincolo di normalizzazione $\|\mathbf{d}\|_2 = 1$. La disuguaglianza di Cauchy–Schwartz e il vincolo di normalizzazione ci garantiscono che

$$\nabla f(\mathbf{x})^{\mathsf{T}} \mathbf{d} \geq -\|\nabla f(\mathbf{x})\|_2 \|\mathbf{d}\|_2 \geq -\|\nabla f(\mathbf{x})\|_2,$$

dove abbiamo uguaglianza se e solo se

$$\mathbf{d} = \mathbf{d}^* = -\frac{\nabla f(\mathbf{x})}{\|\nabla f(\mathbf{x})\|_2}.$$

Da questo segue il risultato. Più intuitivamente, il prodotto interno tra due vettori è massimizzato qundo essi sono paralleli. In questo caso, volendolo minimizzare, dobbiamo cambiare il segno, e poi normalizzare la direzione. □

Controesempio: la superficie [sella] di Peano

Consideriamo la funzione (sella di Peano)

$$f(x_1, x_2) = \big(2x_1^2 - x_2\big)\big(x_2 - x_1^2\big).$$

L'origine è un punto di stazionarietà e tutte le direzioni lineari a partire dall'origine sono direzioni di discesa.

Esprimiamo la superficie in forma parametrica lungo rette che passano per l'origine, quindi della forma $x_2 = mx_1$,

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} t \\ mt \end{bmatrix}.$$

In funzione di t abbiamo

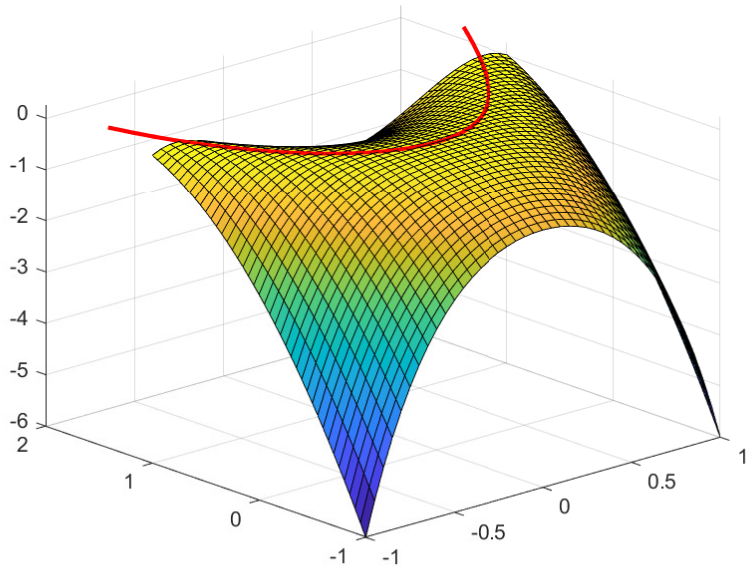
$$\begin{aligned} g(t) &= f\big(x_1(t), x_2(t)\big) = \big(2t^2 - mt\big)\big(mt - t^2\big) = -2t^4 + 3mt^3 - m^2t^2 \\ g'(t) &= -8t^3 + 9mt^2 - 2m^2t \\ g''(t) &= -24t^2 + 18mt - 2m^2 \quad \Rightarrow \quad g''(0) = -2m^2 < 0. \end{aligned}$$

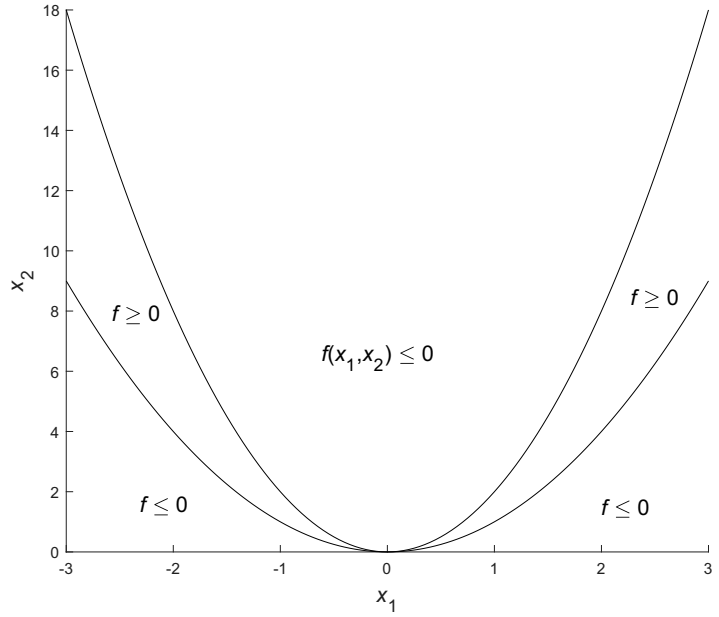
Il caso della retta verticale $x_1 = 0$, porta alla stessa conclusione:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ t \end{bmatrix} \quad \Rightarrow \quad g(t) = -t^2 \quad \Rightarrow \quad g''(0) = -2 < 0.$$

Questo sembra suggerire che l'origine sia un massimo locale, ma tale congettura è falsa. Infatti, spostandosi lungo la curva $x_2 = \sqrt{2}x_1^2$, la funzione è crescente:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} t \\ \sqrt{2}t^2 \end{bmatrix} \quad \Rightarrow \quad g(t) = \left(2t^2 - \sqrt{2}t^2\right)\left(\sqrt{2}t^2 - t^2\right) = \left(3\sqrt{2} - 4\right)t^4.$$





Generare una sequenza di punti $\mathbf{x}^{(k)}$ che convergono a un punto di minimo locale muovendosi lungo direzioni di discesa forniti dal gradiente cambiato di segno:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha^{(k)} \cdot \nabla f(\mathbf{x}^{(k)}), \tag{2}$$

dove lo scalare $\alpha^{(k)} > 0$ è detto **lunghezza del passo** (*step length*), o pù semplicemente "passo".

Una condizione di terminazione possibile è $\|\nabla f(\mathbf{x}^{(k)})\|_\infty < \epsilon$. Si ottiene in questo modo un algoritmo noto come **steepest descent**.

Nei metodi di line search esatti, si ricava il passo ottimo risolvendo il sottoproblema

$$\min_{\alpha \geq 0} h^{(k)}(\alpha) \doteq f\big(\mathbf{x}^{(k)} - \alpha \cdot \nabla f(\mathbf{x}^{(k)})\big).$$

Una strategia alternativa approssimata è il **backtracking**.

Al primo ordine, abbiamo

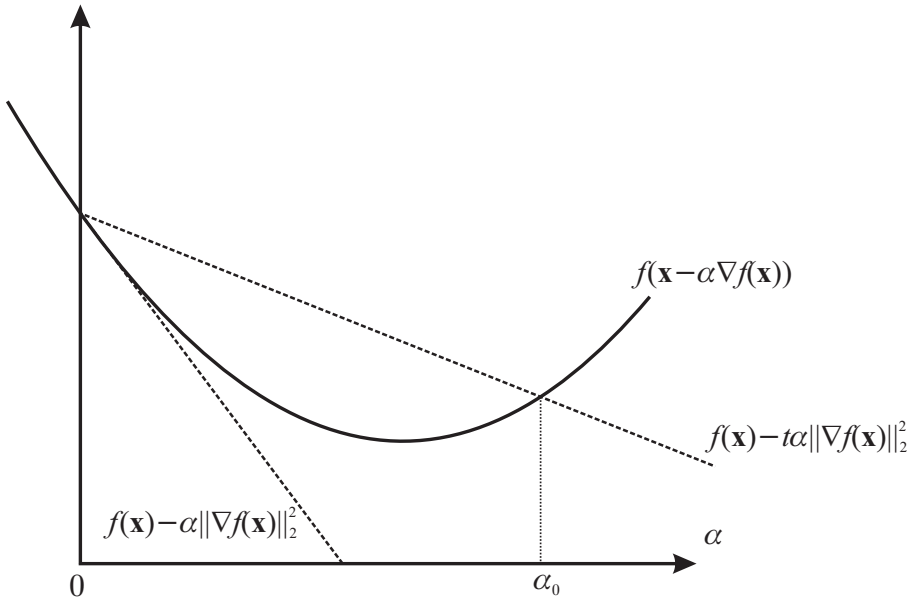
$$f(\mathbf{x} + \alpha\mathbf{d}) \approx f(\mathbf{x}) + \alpha \nabla f(\mathbf{x})^\mathsf{T}\mathbf{d} = f(\mathbf{x}) - \alpha \big\|\nabla f(\mathbf{x})\big\|_2^2.$$

Dobbiamo minimizzare la funzione $h(\alpha) = f(\mathbf{x} - \alpha\nabla f(x))$. L'approssimazione lineare fornisce la retta tangente nel punto $\alpha = 0$.

In questo caso di funzione convessa, la funzione linearizzata fornisce una stima di decrescita molto ottimistica.

Introduciamo un parametro di target $t \in (0, 0.5)$ per la decrescita:

$$f(\mathbf{x}) - t\alpha\left\|\nabla f(\mathbf{x})\right\|_2^2.$$



Esiste un passo α_0 che separa due regioni. Per $\alpha \in [0, \alpha_0]$, il valore vero della funzione è migliore dell'estrapolazione target. Se si sceglie un passo troppo grande, accade il contrario.

La procedura di backtracking parte con un passo α relativamente grande e aggressivo.

Se necessario esso viene ridotto fintanto che vale la condizione

$$f\big(\mathbf{x} - \alpha\nabla f(\mathbf{x})\big) > f\big(\mathbf{x}\big) - t\alpha\left\|\nabla f\big(\mathbf{x}\big)\right\|_2^2. \tag{3}$$

La contrazione del passo segue una formula del tipo $\alpha \leftarrow c\alpha$ ed è regolata dal parametro $c \in (0, 1)$.

È facile vedere che la procedura termina in un numero finito di iterazioni.

Le implementazioni naïve del metodo steepest descent possono soffrire di un comportamento noto come **zig-zagging**.

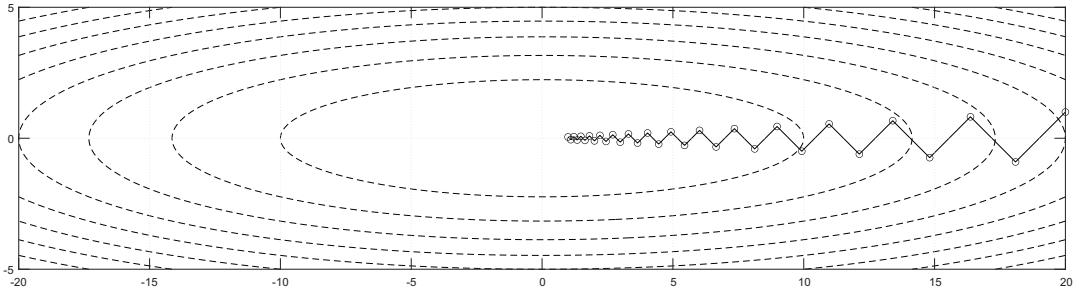
Zig-zagging e mal condizionamento

Consideriamo la funzione quadratica convessa ($\gamma > 0$) $f(x_1, x_2) = \frac{1}{2}\left(x_1^2 + \gamma x_2^2\right)$.

Se partiamo dal punto $\mathbf{x}^{(0)} = (\gamma, 1)$ e applichiamo il metodo steepest descent con line search esatte, si genera la sequenza

$$x_1^{(k)} = \gamma \left(\frac{\gamma - 1}{\gamma + 1}\right)^k, \quad x_2^{(k)} = \left(-\frac{\gamma - 1}{\gamma + 1}\right)^k, \quad f\left(\mathbf{x}^{(k)}\right) = \left(\frac{\gamma - 1}{\gamma + 1}\right)^{2k} f\left(\mathbf{x}^{(0)}\right). \qquad (4)$$

Tale sequenza, per valori grandi di γ, presenta una convergenza lenta verso l'origine (in figura abbiamo $\gamma = 20$).



Il rapporto $(\gamma - 1)/(\gamma + 1)$ è prossimo a 1 per γ grande, nel qual caso le curve di livello sono ellissi schiacciate.

Ortogonalità delle direzioni successive

Nel metodo steepest descent con line search esatta si ottengono direzioni di ricerca successive ortogonali. Se partiamo dal punto $\mathbf{x}^{(k)}$, avremo

$$\left(\mathbf{x}^{(k+2)} - \mathbf{x}^{(k+1)}\right)^{\mathsf{T}}\left(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right) = 0.$$

Infatti

$$\mathbf{x}^{(k+2)} = \mathbf{x}^{(k+1)} - \alpha^{(k+1)}\nabla f(\mathbf{x}^{(k+1)}), \qquad \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha^{(k)}\nabla f(\mathbf{x}^{(k)}),$$

e la condizione di ortogonalità che vogliamo verificare è equivalente a

$$\nabla f(\mathbf{x}^{(k+1)})^{\mathsf{T}}\nabla f(\mathbf{x}^{(k)}) = \nabla f\left(\mathbf{x}^{(k)} - \alpha^{(k)}\nabla f(\mathbf{x}^{(k)})\right)^{\mathsf{T}}\nabla f(\mathbf{x}^{(k)}) = 0. \tag{5}$$

Per ottenere il passo $\alpha^{(k)}$ ottimo, lungo la direzione $\mathbf{d}^{(k)} = -\nabla f(\mathbf{x}^{(k)})$, applichiamo la condizione di stazionarietà

$$\nabla f\left(\mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}\right)^{\mathsf{T}}\mathbf{d}^{(k)} = 0.$$

Tale condizione deriva dall'applicazione delle formule per la derivata di funzione composta e dimostra la condizione di ortogonalità (5).

Convergenza del metodo steepest descent per un problema QP

Analizziamo la convergenza del metodo del gradiente per una funzione strettamente convessa

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathsf{T}}\mathbf{Q}\mathbf{x} \quad \Rightarrow \quad \nabla f(\mathbf{x}) = \mathbf{Q}\mathbf{x},\ \nabla^2 f(\mathbf{x}) = \mathbf{Q} \in \mathbb{S}^n_{++},$$

per cui

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha^{(k)}\mathbf{Q}\mathbf{x}^{(k)} = \big(\mathbf{I}_n - \alpha^{(k)}\mathbf{Q}\big)\mathbf{x}^{(k)}.$$

Ricordando la caratterizzazione variazionale degli autovalori, si ottene

$$\left\|\mathbf{x}^{(k+1)}\right\|_2^2 = \mathbf{x}^{(k)\mathsf{T}}\big(\mathbf{I}_n - \alpha^{(k)}\mathbf{Q}\big)^2\mathbf{x}^{(k)} \leq \lambda_{\max}\big\{\mathbf{I}_n - \alpha^{(k)}\mathbf{Q}\big\}^2 \left\|\mathbf{x}^{(k)}\right\|_2^2,$$

dove $\lambda_{\max}\{\mathbf{A}\}$ è l'autovalore massimo della matrice $\mathbf{A}$.

Indichiamo con λ_j, $j \in [n]$, gli n autovalori della matrice Hessiana $\mathbf{Q}$, e con m e M quello minimo e quello massimo, rispettivamente.

Gli autovalori della matrice $\big(\mathbf{I}_n - \alpha^{(k)}\mathbf{Q}\big)^2$ sono dati da $\big(1 - \alpha^{(k)}\lambda_j\big)^2$, e dipendono da $\alpha^{(k)}$, ma possiamo scrivere che

$$\lambda_{\max}\big\{\mathbf{I}_n - \alpha^{(k)}\mathbf{Q}\big\}^2 = \max\left\{\big(1 - \alpha^{(k)}m\big)^2,\ \big(1 - \alpha^{(k)}M\big)^2\right\}.$$

Si ottiene un upper bound sul rapporto tra le norme dei due punti successivi $\mathbf{x}^{(k)}$ e $\mathbf{x}^{(k+1)}$:

$$\frac{\left\|\mathbf{x}^{(k+1)}\right\|_2}{\left\|\mathbf{x}^{(k)}\right\|_2} \leq \max\left\{|1-\alpha^{(k)}m|,\ |1-\alpha^{(k)}M|\right\}.$$

Un modo per scegliere il passo è minimizzare l'upper bound. Con calcoli elementari, troviamo

$$\alpha^* = \frac{2}{M+m} \quad \Rightarrow \quad \frac{\left\|\mathbf{x}^{(k+1)}\right\|_2}{\left\|\mathbf{x}^{(k)}\right\|_2} \leq \frac{M-m}{M+m}.$$

Questo risultato suggerisce che si avranno difficoltà quando il rapporto M/m è grande, ovvero quando la matrice $\mathbf{Q}$ è mal condizionata.

Il metodo di Newton si basa sull'approssimazione al secondo ordine della funzione obiettivo,

$$f\big(\mathbf{x}^{(k)} + \boldsymbol{\delta}\big) \approx f\big(\mathbf{x}^{(k)}\big) + \big[\nabla f\big(\mathbf{x}^{(k)}\big)\big]^{\mathsf{T}}\boldsymbol{\delta} + \frac{1}{2}\boldsymbol{\delta}^{\mathsf{T}}\mathbf{H}\big(x^{(k)}\big)\boldsymbol{\delta},$$

dove $\boldsymbol{\delta}$ è lo spostamento rispetto al punto corrente $\mathbf{x}^{(k)}$ e $\mathbf{H}\big(x^{(k)}\big)$ è la matrice Hessiana.

Dato il modello locale, se $\mathbf{H}$ è positiva definita, possiamo ricavare lo spostamento ottimale risolvendo

$$\mathbf{H}\big(\mathbf{x}^{(k)}\big)\,\boldsymbol{\delta} = -\nabla f\big(\mathbf{x}^{(k)}\big),$$

e ponendo $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}$.

Il metodo di Newton può essere interpretato come l'applicazione del corrispondente metodo per la soluzione di sistemi di equazioni non lineari alle condizioni di stazionarietà del primo ordine:

$$\mathbf{g}(\mathbf{x}) \doteq \nabla\big(f(\mathbf{x})\big) = \mathbf{0}_n,$$

il metodo di Newton per la sua soluzione è

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \big[D\mathbf{g}\big(\mathbf{x}^{(k)}\big)\big]^{-1}\mathbf{g}\big(\mathbf{x}^{(k)}\big),$$

dove la matrice Jacobiana $D\mathbf{g}(\mathbf{x}^{(k)})$ di $\mathbf{g}$ è la matrice Hessiana $\mathbf{H}(x^{(k)})$ di f.

Il metodo di Newton gode di buone proprietà di convergenza locale, se inizializzato vicino alla soluzione, ma non è globalmente convergente.

Dal punto di vista dell'ottimizzazione, il metodo di Newton soffre di alcune difficoltà:

- il calcolo e l'inversione della matrice Hessiana possono essere problematici;
- non è detto che l'Hessiana sia positiva definita o comunque invertibile;
- non è garantito che il nuovo punto sia migliore di quello precedente.

Per ovviare al primo problema, sono stati proposti metodi detti quasi-Newton, in cui si applicano opportune approssimazioni della matrice Hessiana:

- il metodo DFP (Davidon–Fletcher–Powell);
- il metodo BFGS (Broyden–Fletcher–Goldfarb–Shanno);
- il metodo Levenberg–Marquardt, usato anche per il training di reti neurali, e adatto a problemi di minimi quadrati non lineari della forma

$$F(\mathbf{x}) = \sum_{k=1}^{m} f_k^2(\mathbf{x}).$$

La relazione tra il metodo steepest descent e il metodo di Newton

Il metodo steepest descent può essere interpretato come una approssimazione semplicistica del metodo di Newton.

Sostituiamo alla matrice Hessiana la matrice identità, $\mathbf{I}_n/\alpha$, scalata per un fattore $\alpha > 0$:

$$f(\mathbf{x} + \boldsymbol{\delta}) \approx f(\mathbf{x}) + \nabla f(\mathbf{x})^{\mathsf{T}}\boldsymbol{\delta} + \frac{1}{2\alpha}\|\boldsymbol{\delta}\|_2^2, \tag{6}$$

dove l'ultimo termine può essere interpretato come come un termine di prossimità, che penalizza le deviazioni rispetto al punto corrente.

Infatti, non avrebbe senso minimizzare l'approssimazione lineare, e il termine correttivo evita di trovare un minimo all'infinito.

Se fissiamo il fattore α e applichiamo la condizione del primo ordine alla funzione (6) troviamo

$$\nabla f(\mathbf{x}) + \frac{1}{\alpha}\boldsymbol{\delta} = \mathbf{0}_n \quad \Rightarrow \quad \boldsymbol{\delta} = -\alpha\nabla f(\mathbf{x}).$$

La lunghezza del passo ha una relazione inversa con la penalizzazione di prossimità.

Definiamo l'approssimazione al secondo ordine

$$f_2(\mathbf{x}) = f(\mathbf{x}^{(k)}) + \mathbf{g}_k^\top(\mathbf{x} - \mathbf{x}^{(k)}) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^{(k)})^\top \mathbf{H}_k(\mathbf{x} - \mathbf{x}^{(k)}),$$

dove per comodità poniamo $\mathbf{g}_k \doteq \nabla f(\mathbf{x}^{(k)})$ e $\mathbf{H}_k \doteq \nabla^2 f(\mathbf{x}^{(k)})$.

Essa è affidabile solo in un intorno non troppo ampio della soluzione corrente. Pertanto, a ogni iterazione si risolve il problema vincolato

$$\begin{aligned} \min \quad & \frac{1}{2}\boldsymbol{\delta}_k^\top \mathbf{H}_k \boldsymbol{\delta}_k + \mathbf{g}_k^\top \boldsymbol{\delta}_k \\ \text{s.t.} \quad & \|\boldsymbol{\delta}_k\|_2^2 \leq \Delta_k, \end{aligned}$$

dove il parametro Δ_k limita il passo.

Questo può essere valutato confrontando la riduzione di obiettivo predetta dal modello per uno spostamento $\boldsymbol{\delta}_k$ rispetto a quello davvero ottenuto. A questo scopo, risolto il problema con il bound Δ_k, si calcola il rapporto di riduzione

$$R_k = \frac{f(\mathbf{x}^{(k)}) - f(\mathbf{x}^{(k+1)})}{f_2(\mathbf{x}^{(k)}) - f_2(\mathbf{x}^{(k+1)})}.$$

Un rapporto minore di 1 indica che la previsione è ottimistica, e quindi il modello locale quadratico risulta inadeguato per un passo così grande; pertanto, occorre ridurre il bound Δ_k.

Scegliamo i parametri β_1, β_2, γ_1, γ_2, soggetti alle condizioni $0 < \beta_1 < \beta_2 < 1$, $0 < \gamma_1 < 1 < \gamma_2$. Settiamo la soluzione iniziale $\mathbf{x}^{(0)}$, il raggio iniziale della trust region Δ_0 ed il contatore di iterazione $k \leftarrow 0$.

1: **if** $\left\|\nabla f\left(\mathbf{x}^{(k)}\right)\right\|_\infty < \epsilon$ **then**
2: stop con soluzione ottima
3: **end if**
4: Trova $\widehat{\mathbf{x}}^{(k)}$ risolvendo il problema min $f_2(\mathbf{x})$ sotto il vincolo

$$\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 \leq \Delta_k$$

5: **if** $f\left(\widehat{\mathbf{x}}^{(k)}\right) < f\left(\mathbf{x}^{(k)}\right)$ **then**
6: Poni $\mathbf{x}^{(k+1)} \leftarrow \widehat{\mathbf{x}}^{(k)}$, calcola il rapporto R_k e vai a step 10.
7: **else**
8: Poni $\Delta_{k+1} \leftarrow \gamma_1 \Delta_k$ e vai a step 1
9: **end if**
10: **if** $R_k < \beta_1$ **then**
11: Poni $\Delta_{k+1} \leftarrow \gamma_1 \Delta_k$
12: **else if** $R_k > \beta_2$ e $\left\|\widehat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k)}\right\|_2 = \Delta_k$ **then**
13: Poni $\Delta_{k+1} \leftarrow \gamma_2 \Delta_k$
14: **else**
15: Poni $\Delta_{k+1} \leftarrow \Delta_k$
16: **end if**
17: Poni $k \leftarrow k + 1$ e vai a step 1.

Si possono dimostrare i seguenti fatti:

1. Esiste $\mu_k \in \mathbb{R}_+$ tale che
$$\big(\mathbf{H}_k + \mu_k \mathbf{I}_n\big)\boldsymbol{\delta}_k + \mathbf{g}_k = \mathbf{0}_n. \tag{7}$$
2. La matrice $\mathbf{H}_k + \mu_k \mathbf{I}_n$ è positiva semidefinita.

Riscriviamo l'equazione (7) come
$$-\mathbf{g}_k = \big(\mathbf{H}_k + \mu_k \mathbf{I}_n\big)\boldsymbol{\delta}_k,$$
premoltiplichiamo per $\boldsymbol{\delta}_k^\mathsf{T}$, e sfruttiamo la seconda proprietà, concludendo che
$$-\boldsymbol{\delta}_k^\mathsf{T}\mathbf{g}_k = \boldsymbol{\delta}_k^\mathsf{T}\big(\mathbf{H}_k + \mu_k \mathbf{I}_n\big)\boldsymbol{\delta}_k \geq 0.$$
Nel caso in cui la matrice $\mathbf{H}_k + \mu_k \mathbf{I}_n$ sia positiva definita, questo implica che il metodo trust region individua una direzione di discesa.

Metodi derivative-free e black-box

Il metodo del gradiente assume che esista il gradiente e ne sfrutta la conoscenza. Esistono metodi alternativi di ricerca diretta (direct search) che richiedono solo la capacità di valutare la funzione.

I metodi **derivative-free** non usano derivate, ma la loro esistenza viene sfruttata per dimostrarne di convergenza, a differenza dei metodi **black-box**.

Il metodo di direct search più banale è noto come coordinate search ed esplora l'intorno della soluzione corrente perturbando una variabile decisionale per volta. Esso si arresta quando non si riesce più a migliorare l'obiettivo.

- L'esplorazione lungo direzioni parallele agli assi coordinati non è necessariamente efficiente o efficace. Metodi alternativi, come simplex search e pattern search usano meccanismi di polling diversi.

- Si tratta di un metodo di ricerca locale che mira a un minimo locale, ma non globale. Metodi alternativi, come gli algoritmi genetici o le varianti di particle swarm optimization si basano su una popolazione di soluzioni e su meccanismi di ricerca stocastica per tentare di non cadere in minimi locali.

L'algoritmo è anche noto come *simplex search*, ma non va confuso con il metodo del simplesso per la programmazione lineare.

Esso non esplora lo spazio delle soluzioni sulla base di un singolo punto corrente, ma di una popolazione di soluzioni che forma un simplesso.

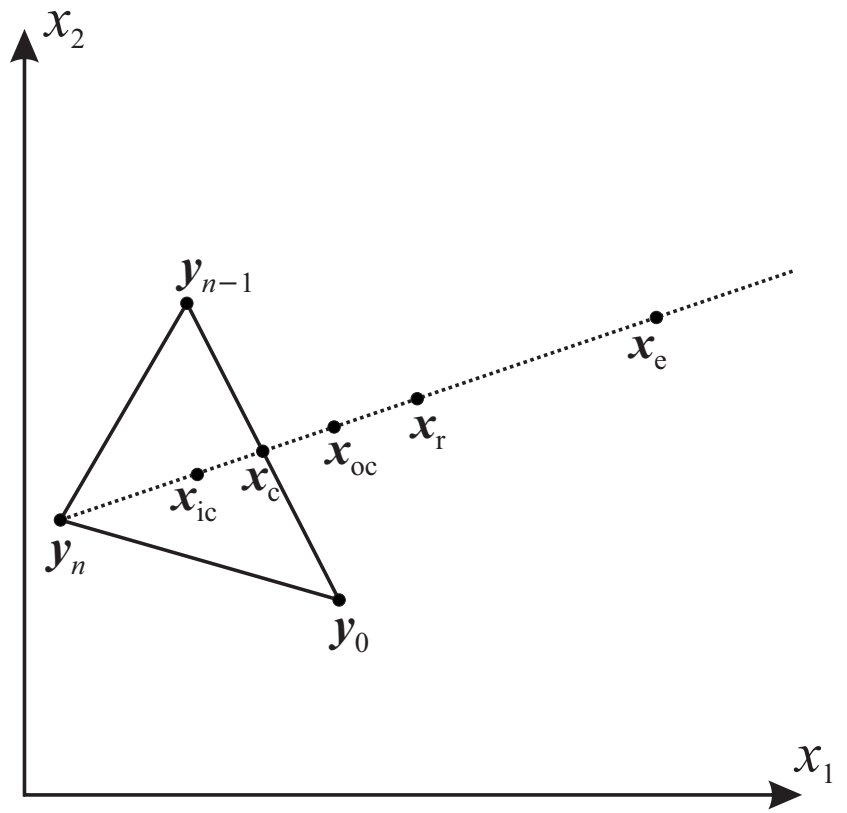
A ogni iterazione dell'algoritmo avremo un simplesso formato dagli $n+1$ punti $\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_n$ in $\mathbb{R}^n$, numerati in modo tale che

$$f(\mathbf{y}_0) \leq f(\mathbf{y}_1) \leq \cdots \leq f(\mathbf{y}_{n-1}) \leq f(\mathbf{y}_n),$$

per cui $\mathbf{y}_0$ è il punto migliore e $\mathbf{y}_n$ il punto peggiore.

Si trasforma il simplesso eliminando il punto peggiore e sostituendolo con un altro vertice, che sta su una direzione di ricerca che va dal punto peggiore al centroide degli altri n punti,

$$\mathbf{x}_c = \frac{1}{n} \sum_{k=0}^{n-1} \mathbf{y}_k.$$



Possiamo considerare punti del tipo

$$\mathbf{x}_c + \alpha(\mathbf{x}_c - \mathbf{y}_n), \tag{8}$$

con $\alpha > 0$ da determinare. Inoltre, per fare convergere il metodo a qualche punto, dovremo contrarre il simplesso, in modo che al limite esso collassi in un solo punto.

Un nuovo punto candidato è ottenuto per riflessione del punto $\mathbf{y}_n$ attraverso $\mathbf{x}_c$, con $\alpha = 1$,

$$\mathbf{x}_r = \mathbf{y}_n + (\mathbf{x}_c - \mathbf{y}_n).$$

Se il nuovo punto è il nuovo ottimo corrente, questo suggerisce di espandere il simplesso nella direzione di ricerca individuata, aumentando α. Al contrario, se esso è il peggiore, occorre effettuare un'operazione di contrazione, riducendo α.

La scelta dipende da dove si posiziona il valore della funzione nel punto riflesso rispetto al valore nel punto migliore $\mathbf{y}_0$ del simplesso, nel punto peggiore $\mathbf{y}_n$, e nel punto $\mathbf{y}_{n-1}$.

Caso $f(\mathbf{x}_r) < f(\mathbf{y}_0)$. La direzione può essere esplorata in modo più aggressivo, confrontando i valori del punto riflesso $\mathbf{x}_r$ e del punto di espansione $\mathbf{x}_e$, per α pari al parametro di espansione $\delta_e \in (1, \infty)$. Sia $\mathbf{x}_{\text{new}}$ il punto migliore tra $\mathbf{x}_r$ e $\mathbf{x}_e$. Il nuovo simplesso è formato dai vertici $\{\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}, \mathbf{x}_{\text{new}}\}$.

Caso $f(\mathbf{y}_0) \leq f(\mathbf{x}_r) < f(\mathbf{y}_{n-1})$**.** In questa situazione, la direzione di ricerca è utile, ma non tanto da suggerire una possibile espansione del simplesso. Il nuovo simplesso è formato dai vertici $\{\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}, \mathbf{x}_r\}$.

Caso $f(\mathbf{y}_{n-1}) \leq f(\mathbf{x}_r) < f(\mathbf{y}_n)$**.** Sebbene il nuovo punto migliori quello peggiore nel simplesso attuale, la direzione non è così promettente e si considera una contrazione esterna. Dato un parametro di contrazione esterna $\delta_{oc} \in (0, 1)$, si confronta il punto riflesso $\mathbf{x}_r$, a cui corrisponde $\alpha = 1$, con il punto $\mathbf{x}_{oc}$ ottenuto ponendo $\alpha = \delta_{oc}$. Sia $\mathbf{x}_{\text{new}}$ il punto migliore tra $\mathbf{x}_r$ e $\mathbf{x}_{oc}$. Il nuovo simplesso è formato dai vertici $\{\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}, \mathbf{x}_{\text{new}}\}$.

Caso $f(\mathbf{y}_n) \leq f(\mathbf{x}_r)$**.** In questa situazione, il punto riflesso è peggiore di tutti quelli del simplesso corrente, che viene quindi contratto internamente. Dato un parametro di contrazione interna $\delta_{ic} \in (-1, 0)$, si confronta il punto peggiore $\mathbf{y}_n$, con il punto $\mathbf{x}_{ic}$ ottenuto ponendo $\alpha = \delta_{ic}$ (tale punto è interno al simplesso corrente).

1. Se $f(\mathbf{x}_{ic}) < f(\mathbf{y}_n)$, si forma un nuovo simplesso contratto e formato dai punti $\{\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}, \mathbf{x}_{ic}\}$.
2. Se $f(\mathbf{x}_{ic}) \geq f(\mathbf{y}_n)$, si contrae il simplesso nella direzione del punto migliore. dato un parametro di contrazione $\gamma \in (0, 1)$, si determinano i nuovi vertici

$$\mathbf{x}_k = \mathbf{y}_0 + \gamma(\mathbf{y}_k - \mathbf{y}_0), \quad k = 1, 2, \ldots, n,$$

e si costruisce il nuovo simplesso dato dai vertici $\{\mathbf{y}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n\}$.

Si cerca di ovviare ai limiti del polling tipico del metodo *coordinate search*, in cui si considerano punti del tipo $\mathbf{y} \in \{\mathbf{x}^{(k)} \pm \delta^{(k)}\mathbf{e}_i : i \in [n]\}$ e si procede a contrazioni successive di $\delta^{(k)}$,

Teorema: spanning set positivi e direzioni di discesa. Sia $\mathcal{D}$ uno spanning set positivo per $\mathbb{R}^n$ e $\mathbf{y} \in \mathbb{R}^n$ un vettore non nullo. Allora esiste un vettore $\mathbf{d} \in \mathcal{D}$ tale che $\mathbf{y}^\mathsf{T}\mathbf{d} < 0$. Inoltre, se $f \in \mathcal{C}^1$ e $\nabla f(\mathbf{x}) \neq \mathbf{0}_n$ per un punto $\mathbf{x} \in \mathbb{R}^n$, allora esiste $\mathbf{d} \in \mathcal{D}$ che è una direzione di discesa per f in $\mathbf{x}$.

Dimostrazione. Poiché $\mathcal{D}$ è uno spanning set positivo, che supponiamo costituito da vettori $\mathbf{d}_i$, $i \in [m]$, possiamo scrivere

$$-\mathbf{y} = \sum_{i \in [m]} \lambda_i \mathbf{d}_i,$$

per un vettore di coefficienti $\boldsymbol{\lambda} \in \mathbb{R}^n_+$. Quindi

$$0 > -\|\mathbf{y}\|_2^2 = -\mathbf{y}^\mathsf{T}\mathbf{y} = \sum_{i \in [m]} \lambda_i \mathbf{y}^\mathsf{T}\mathbf{d}_i.$$

Dato che $\lambda_i \geq 0$, per almeno un indice i avremo $\mathbf{y}^\mathsf{T}\mathbf{d}_i < 0$. Nel caso di una funzione differenziabile, per cui la derivata direzionale è data da $f'(\mathbf{x}; \mathbf{d}) = \nabla f(\mathbf{x})^\mathsf{T}\mathbf{d}$, basta scegliere $\mathbf{y} = \nabla(\mathbf{x}) \neq \mathbf{0}_n$ e applicare la prima parte del teorema. □

Gli algoritmi GPS usano basi positive per esplorare localmente lo spazio delle soluzioni vicino alla soluzione corrente, attraverso un meccanismo di polling. Inoltre si applicano anche fasi di search per esplorare lo spazio delle soluzioni in maniera più globale.

Oltre al concetto di spanning set positivo, risulta fondamentale quello di mesh.

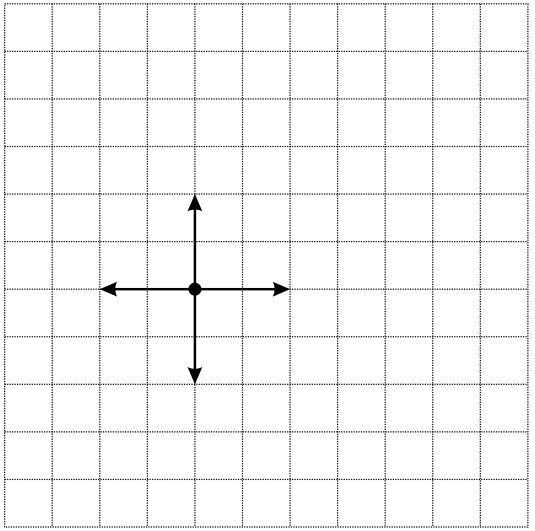
Definizione: mesh. Consideriamo una matrice invertibile $\mathbf{G} \in \mathbb{R}^{n\times n}$ e una matrice $\mathbf{Z} \in \mathbb{Z}^{n\times p}$, le cui colonne costituiscono un spanning set positivo per $\mathbb{R}^n$. La mesh generata dalla matrice $\mathbf{D} = \mathbf{GZ}$, centrata nel punto $\mathbf{x}^{(k)}$ con parametro di granularità (coarseness) $\delta^{(k)} > 0$ è definita da $\mathcal{M}^{(k)} = \left\{\mathbf{x}^{(k)} + \delta^{(k)}\mathbf{Dy} : \mathbf{y} \in \mathbb{N}^p\right\}$, dove $\mathbb{N}$ è l'insieme dei numeri naturali.

La figura illustra la mesh generata in $\mathbb{R}^2$ se poniamo

$$\mathbf{G} = \mathbf{I}_2, \qquad \mathbf{Z} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}.$$

Si tratta di una semplice mesh con maglie quadrate, ma possiamo generare mesh con maglie triangolari e così via.

La mesh non viene esplicitamente costruita, ma definisce semplicemente un modo di discretizzare lo spazio.



Indichiamo con $\mathcal{D}$ il set di direzioni formato dalle colonne della matrice $\mathbf{D}$.

Step 0: inizializzazione. Scegli $\mathbf{x}^{(0)}$ il punto iniziale, un parametro di contrazione $\tau \in (0,1)$, una tolleranza ϵ. Inizializza il passo (mesh size) $\delta^{(0)} \in (0,\infty)$ e poni $k \leftarrow 0$.

Step 1: ricerca. Se $f\big(\mathbf{y}\big) < f\big(\mathbf{x}^{(k)}\big)$ per qualche $\mathbf{y}$ in un sottoinsieme finito della mesh $\mathcal{M}^{(k)}$, allora poni $\mathbf{x}^{(k+1)} \leftarrow \mathbf{y}$, $\delta^{(k+1)} \leftarrow \tau^{-1}\delta^{(k)}$ e vai a step 3; altrimenti vai a step 2.

Step 2: polling. Seleziona uno spanning set positivo $\mathcal{D}^{(k)} \subseteq \mathcal{D}$. Se $f\big(\mathbf{y}\big) < f\big(\mathbf{x}^{(k)}\big)$ per qualche $\mathbf{y} \in \big\{\mathbf{x}^{(k)} + \delta^{(k)}\mathbf{d} : \mathbf{d} \in \mathcal{D}^{(k)}\big\}$, allora poni $\mathbf{x}^{(k+1)} \leftarrow \mathbf{y}$, $\delta^{(k+1)} \leftarrow \tau^{-1}\delta^{(k)}$. Altrimenti $\mathbf{x}^{(k+1)}$ è un ottimizzatore locale per la mesh; poni $\mathbf{x}^{(k+1)} \leftarrow \mathbf{x}^{(k)}$, $\delta^{(k+1)} \leftarrow \tau\delta^{(k)}$.

Step 3: terminazione. Se $\delta^{(k+1)} > \epsilon$, incrementa $k \leftarrow k+1$ e vai a step 1. Altrimenti stop.

Gli algoritmi genetici sono una classe di metodi basati su una popolazione di soluzioni, che evolve secondo un algoritmo stocastico che imita il meccanismo della selezione evolutiva.

Occorre scegliere uno schema di **codifica** delle soluzioni. A ogni soluzione corrisponde un *cromosoma*, che è una collezione di attributi, a loro volta corrispondenti ai *geni* del cromosoma.

Nella versione originale del metodo, le soluzioni erano codificate su geni binari. In un problema di ottimizzazione nel continuo, sembra più opportuno rappresentare una soluzione secondo una codifica naturale.

Nel caso di problemi di ottimizzazione combinatoria, si pongono ulteriori problemi, come quello della rappresentazione di una permutazione di oggetti.

Un'altra scelta fondamentale è quella della **funzione di fitness** g, che non coincide necessariamente con la funzione obiettivo f. Spesso si opera una trasformazione in modo che la fitness sia positiva, e definisce la **probabilità di selezione** dell'individuo j come

$$\frac{g(\mathbf{x}_j)}{\sum_{i=1}^{p} g(\mathbf{x}_i)}. \tag{9}$$

Step 0: Inizializzazione. Fissa la dimensione p della popolazione e crea quella iniziale: $\mathcal{P}^{(0)} \leftarrow \{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$. Fissa la probabilità di mutazione $\gamma \in (0, 1)$ e il contatore $k \leftarrow 0$.

Step 1: Valutazione. Valuta una funzione di fitness g per ogni elemento $\mathbf{x}$ della popolazione corrente $\mathcal{P}^{(k)}$.

Step 2: Riproduzione/Sopravvivenza. Vai, in maniera casuale, a step 3 oppure step 4.

Step 3: Sopravvivenza. Seleziona in modo casuale, sulla base della fitness g, un individuo e aggiungilo alla nuova popolazione $\mathcal{P}^{(k+1)}$. Vai a step 5.

Step 4: Riproduzione. Esegui le due operazioni seguenti.

Crossover: scegli, sulla base della fitness, due individui che vengono impiegati come genitori per la creazione di un nuovo individuo.

Mutazione: con probabilità γ applica una mutazione casuale al nuovo elemento; aggiungi il nuovo elemento alla nuova popolazione $\mathcal{P}^{(k+1)}$.

Step 5: Aggiornamento. Se necessario, aggiorna il miglior elemento corrente, $\mathbf{x}_{\text{best}}$, in termini della funzione obiettivo f. Se la nuova popolazione $\mathcal{P}^{(k+1)}$ ha dimensione inferiore a p, torna a step 2. Se è soddisfatto un criterio di terminazione restituisci $\mathbf{x}_{\text{best}}$. Altrimenti poni $k \leftarrow k + 1$ e torna al passo 1.

Un aspetto delicato è fissare il grado di **elitarismo**, che mira a mantenere alcuni elementi con caratteristiche di eccellenza nella popolazione, in modo da intensificare la ricerca nelle zone promettenti dello spazio di ricerca.

Un algoritmo troppo elitario finisce per convergere prematuramente a una soluzione scarsa. La riproduzione, al contrario, cerca di diversificare la popolazione, in modo da esplorare regioni alternative dello spazio di ricerca.

Poiché è improbabile che un individuo con un valore scarso di fitness venga selezionato sulla base di probabilità date dalla equazione (9), un'alternativa si basa su **mini-tornei**, in cui si campiona in modo casuale un sottoinsieme della popolazione, e si applicano le probabilità di selezione al sottoinsieme, aumentando quindi la probabilità di sopravvivenza o riproduzione di un elemento scarso.

Inoltre, occorre definire come vengono realizzate le operazioni di mutazione e crossover, che dipendono dal tipo di codifica selezionato.

Per esempio, dati due individui nella popolazione corrente, possiamo scegliere casualmente una posizione di breakpoint $k \in \{1, 2, \ldots, n\}$ e generare due successori (offspring) come segue:

$$\left\{ \begin{array}{l} x_1, x_2, \ldots, x_k, x_{k+1}, \ldots, x_n \\ y_1, y_2, \ldots, y_k, y_{k+1}, \ldots, y_n \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} x_1, x_2, \ldots, x_k, y_{k+1}, \ldots, y_n \\ y_1, y_2, \ldots, y_k, x_{k+1}, \ldots, x_n \end{array} \right\}.$$

In alternativa, si può applicare un crossover con doppio breakpoint, oppure creare un nuovo individuo come combinazione convessa di due o più genitori.

Gli algoritmi particle swarm optimization (**PSO**), come gli algoritmi genetici, sono algoritmi di ricerca stocastica basati su una popolazione di individui che corrispondono a soluzioni del problema di ottimizzazione.

Consideriamo una popolazione (sciame) di m particelle, $j \in [m]$, la cui *posizione* al tempo discreto $t = 1, 2, 3, \ldots$, è un vettore $\mathbf{x}_j(t) = \Big[x_{1j}(t),\, x_{2j}(t),\, \ldots,\, x_{nj}(t)\Big]^{\mathsf{T}} \in \mathbb{R}^n$. Il moto di ogni particella è funzione di tre fattori:

1. **Fattore di inerzia**. Il vettore di velocità $\mathbf{v}_j(t)$ tende ad essere mantenuto.

2. **Fattore cognitivo**. Ogni particella tende a muoversi verso il proprio punto di ottimo corrente $\mathbf{p}_j^*(t)$.

3. **Fattore sociale**. Ogni particella tende a muoversi verso l'ottimo corrente dello sciame $\mathbf{g}^*(t)$.

Tipica versione dell'algoritmo PSO:

$$\mathbf{v}_j(t+1) = \mathbf{v}_j(t) + c_1 r_{1j}(t) \cdot \big[\mathbf{p}_j^*(t) - \mathbf{x}_j(t)\big] + c_2 r_{2j}(t) \cdot [\mathbf{g}^*(t) - \mathbf{x}_j(t)]\,, \tag{10}$$

$$\mathbf{x}_j(t+1) = \mathbf{x}_j(t) + \mathbf{v}_j(t+1). \tag{11}$$

I singoli contributi cono scalati da due coefficienti c_1 e c_2 e moltiplicati per variabili casuali $r_{1j}(t)$ e $r_{2j}(t)$ (una possibilità è usare variabili con distribuzione uniforme).

Uso di metamodelli e ottimizzazione surrogata

I metodi stocastici basati su una popolazione, come gli algoritmi genetici e PSO, hanno forti limitazioni intrinseche:

- Essi sono molto costosi quando ogni singola valutazione della funzione obiettivo è computazionalmente onerosa.
- Quando la valutazione è soggetta a rumore, come nel caso di una simulazione Monte Carlo, il confronto di due alternative è soggetto a errori.
- Non si sfrutta il fatto che, se la funzione è continua, valutare la funzione in un punto può fornire informazioni utili circa i valori in un intorno del punto testato.

Per ovviare a tali difficoltà, è possibile ricorrere ad altri approcci, basati su un **metamodello** o **funzione surrogata** $\widehat{f}(\mathbf{x})$, che approssima la funzione obiettivo $f(\mathbf{x})$ e richiede uno sforzo computazionale sensibilmente ridotto per la sua valutazione.

Abbiamo un campione di punti $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\}$, ai quali corrisponde il valore $y^{(i)}$, $i \in [n]$, ottenuto valutando (o stimando) il valore della funzione.

In assenza di rumore possiamo chiedere la condizione di interpolazione $\widehat{f}(\mathbf{x}^{(i)}) = y^{(i)}$, $i \in [n]$. In altri casi, dobbiamo ipotizzare un meccanismo di generazione dei dati come

$$y^{(i)} = f(\mathbf{x}^{(i)}) + \epsilon^{(i)},$$

dove l'errore $\epsilon^{(i)}$ non è osservabile, e minimizzare una misura di distanza, ad esempio

$$\sum_{i \in [n]} \left(\widehat{f}(\mathbf{x}^{(i)}) - y^{(i)}\right)^2.$$

A differenza del classico approccio della regressione lineare, questo **non** porta necessariamente a un sottoproblema di fitting di facile soluzione, in quanto l'architettura di approssimazione può essere non lineare nei parametri o non parametrica.

Approcci possibili:

- metodi classici delle superfici di risposta (**RSM** – response surface method), di tipo lineare o quadratico;
- metodi basati su funzioni di base radiali (**RBF** – radial basis function);
- metodi storicamente noti come kriging (dal nome di Danie Krige, ingegnere minerario sudafricano che li introdusse per applicazioni geostatistiche), ora etichettati come Gaussian Process Regression (**GPR**).

Funzioni di penalità: il caso di vincoli di uguaglianza

Il problema con soli vincoli di uguaglianza

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & h_i(\mathbf{x}) = 0, \qquad i \in E \end{aligned}$$

può essere approssimato mediante il problema non vincolato

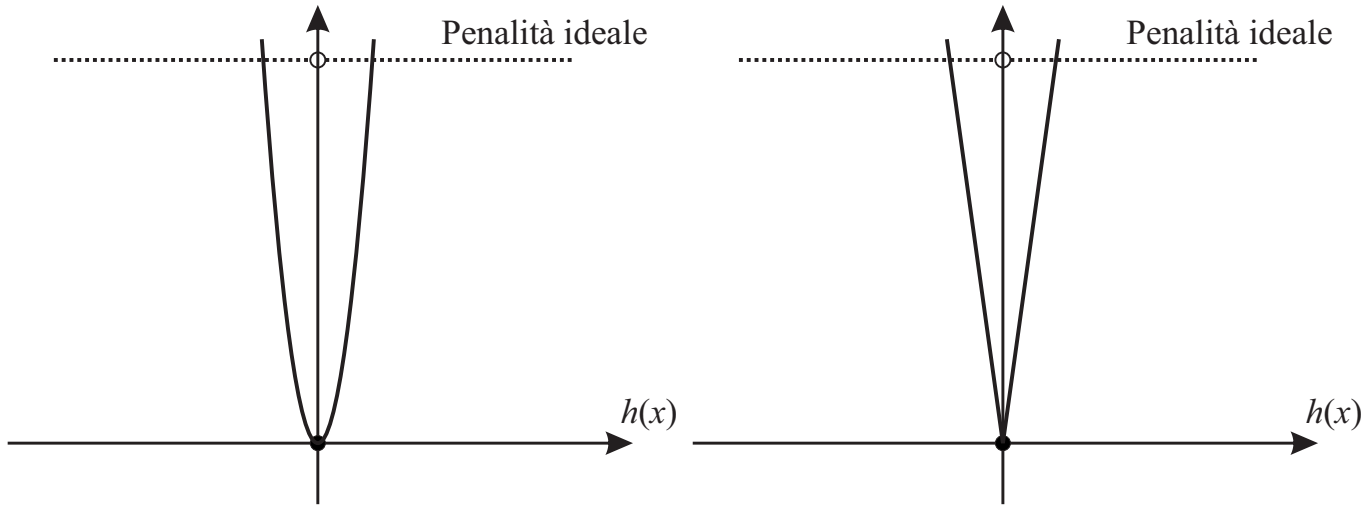
$$\min \Phi(\mathbf{x}, \sigma) = f(\mathbf{x}) + \sigma \sum_{i \in E} h_i^2(\mathbf{x}).$$

Se il coefficiente di penalità σ è sufficientemente grande, il metodo funziona come un'ottimizzazione lessicografica approssimata.

Possiamo sempre ricondurre il problema vincolato $\min_{\mathbf{x} \in S} f(\mathbf{x})$ al problema non vincolato $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) + \Phi(\mathbf{x})$, dove $\Phi(\mathbf{x})$ è una funzione a valori estesi

$$\Phi(\mathbf{x}) = \begin{cases} +\infty & \text{se } \mathbf{x} \notin S \\ 0 & \text{se } \mathbf{x} \in S. \end{cases}$$

Chiaramente, tale funzione di penalità ideale non è computazionalmente trattabile e va approssimata.



La figura illustra anche una penalità alternativa a quella quadratica:

$$\min \Phi(\mathbf{x}, \sigma) = f(\mathbf{x}) + \sigma \sum_{i \in E} |h_i(\mathbf{x})|.$$

La penalità l_1 non è differenziabile, cosa che può essere rilevante o meno (non lo è se si applica un algoritmo genetico, per esempio). D'altro canto, essa penalizza maggiormente piccole deviazioni, e quindi può richiedere valori più piccoli del coefficiente σ.

Un valore molto grande di σ rende il problema mal condizionato. Si fissa quindi una sequenza crescente di coefficienti $\sigma^{(k)}$, a cui corrisponde una sequenza di approssimazioni

$$\min \Phi\left(\mathbf{x}, \sigma^{(k)}\right) \quad \Rightarrow \quad \widehat{\mathbf{x}}^{(k)}.$$

La stima $\widehat{\mathbf{x}}^{(k)}$ della soluzione ottima al passo k sarà soluzione iniziale per la prossima iterazione.

Nel caso di vincoli di disuguaglianza

$$\begin{array}{ll} \min & f(\mathbf{x}) \\ \text{s.t.} & g_i(\mathbf{x}) \leq 0 \qquad i \in I, \end{array}$$

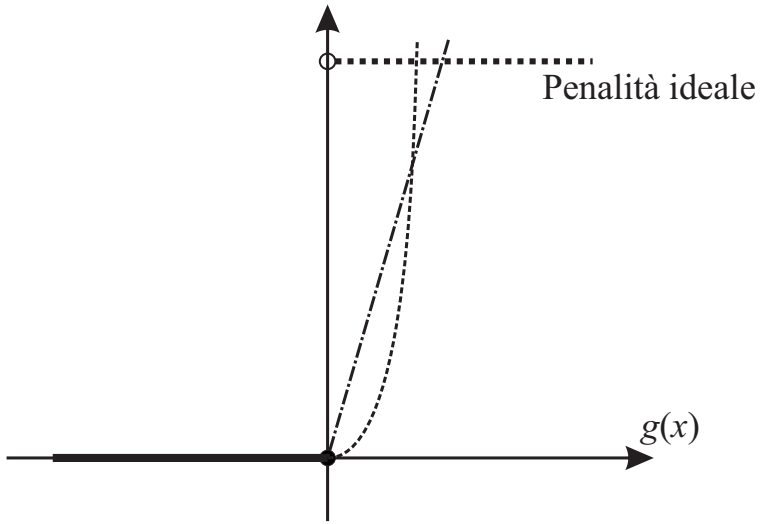
occorre penalizzare solo valori positivi $g_i^+(\mathbf{x})$ dei vincoli.

Possiamo utilizzare penalità differenziabili (smooth)

$$f(\mathbf{x}) + \sigma \sum_{i \in I} \left[g_i^+(\mathbf{x}) \right]^2$$

o non smooth

$$f(\mathbf{x}) + \sigma \sum_{i \in I} g_i^+(\mathbf{x}).$$



Esiste un tradeoff tra avere punti di non differenziabilità o dovere utilizzare coefficienti di penalità grandi. In effetti le funzioni di penalità esatte sono state tra i fattori che hanno portato allo sviluppo di metodi di ottimizzazione non smooth.

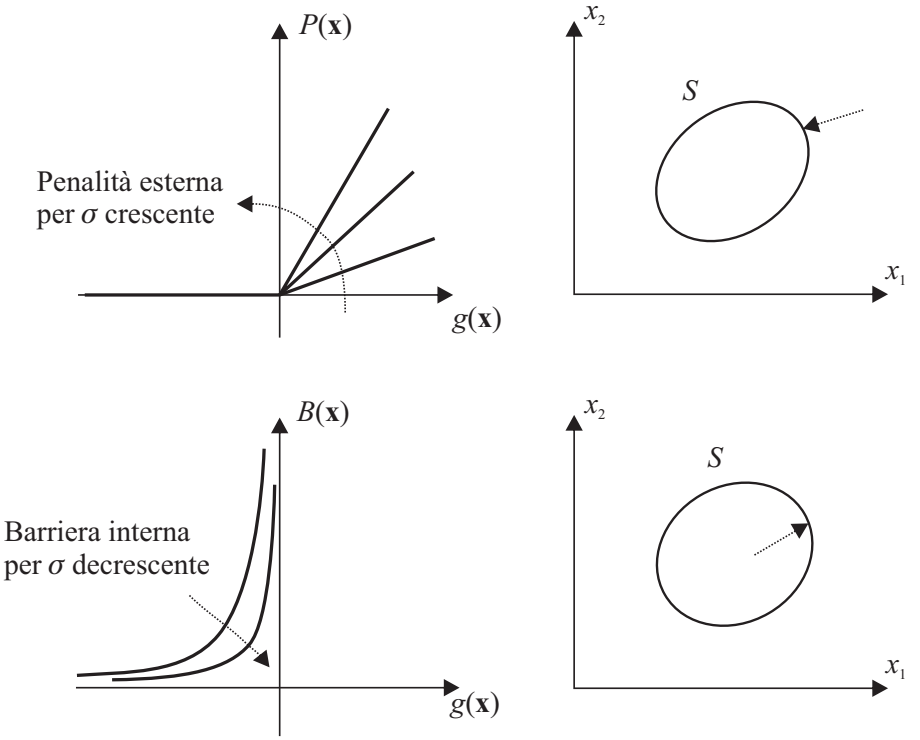
Abbiamo considerato penalità *esterne*, che portano a convergere verso la regione di ammissibilità dal di fuori.

Nel caso di vincoli di disuguaglianza hard, un approccio alternativo si basa su una penalità interna detta *barriera*, per cui si converge alla soluzione ottima dall'interno.

Nel caso di una barriera interna, si risolve comunque un problema non vincolato

$$\min f(\mathbf{x}) + \sigma B(\mathbf{x}),$$

ma per valori *decrescenti* del coefficente σ.



Possibile funzioni di penalità interna sono la barriera inversa

$$B(\mathbf{x}) = -\sum_{i \in I} \frac{1}{g_i(\mathbf{x})}.$$

e la **barriera logaritmica**:

$$B(\mathbf{x}) = -\sum_{i \in I} \log(-g_i(\mathbf{x})).$$

I metodi di ottimizzazione conica per punti interni usano barriere logaritmiche. Per esempio, a un vincolo di non negatività $x_j \geq 0$ in un problema di massimizzazione si associa la barriera $\log(x_j)$.

Il metodo del subgradiente si applica a problemi convessi ma non differenziabili.

ppuò non essere semplice ricavare un subgradiente, ma esistono casi particolari in cui il compito risulta agevole, come nelle applicazioni della teoria della dualità.

In questo caso si tratta di massimizzare una funzione concava $w(\boldsymbol{\mu})$ ma non necessariamente differenziabile, e spesso con vincoli di non negatività sulle variabili $\boldsymbol{\mu}$.

Supponiamo quindi di dover risolvere il problema convesso di massimizzazione

$$\max_{\boldsymbol{\mu} \geq \mathbf{0}_m} w(\boldsymbol{\mu}),$$

e di conoscere un subgradiente $\boldsymbol{\gamma}^{(k)}$ nel punto $\boldsymbol{\mu}^{(k)}$.

Ci chiediamo se sia possibile applicare un metodo di ricerca del tipo

$$\boldsymbol{\mu}^{(k+1)} = \max\left\{\mathbf{0}_m,\ \boldsymbol{\mu}^{(k)} + \alpha^{(k)}\boldsymbol{\gamma}^{(k)}\right\},$$

che è il chiaro analogo di un metodo steepest ascent, con una proiezione su $\mathbb{R}^m_+$.

La risposta è positiva, ma c'è una complicazione. La difficoltà sta nel fatto che un subgradiente non garantisce, a differenza del gradiente, di trovare direzioni di ascesa o discesa.

Consideriamo la massimizzazione della funzione concava non differenziabile

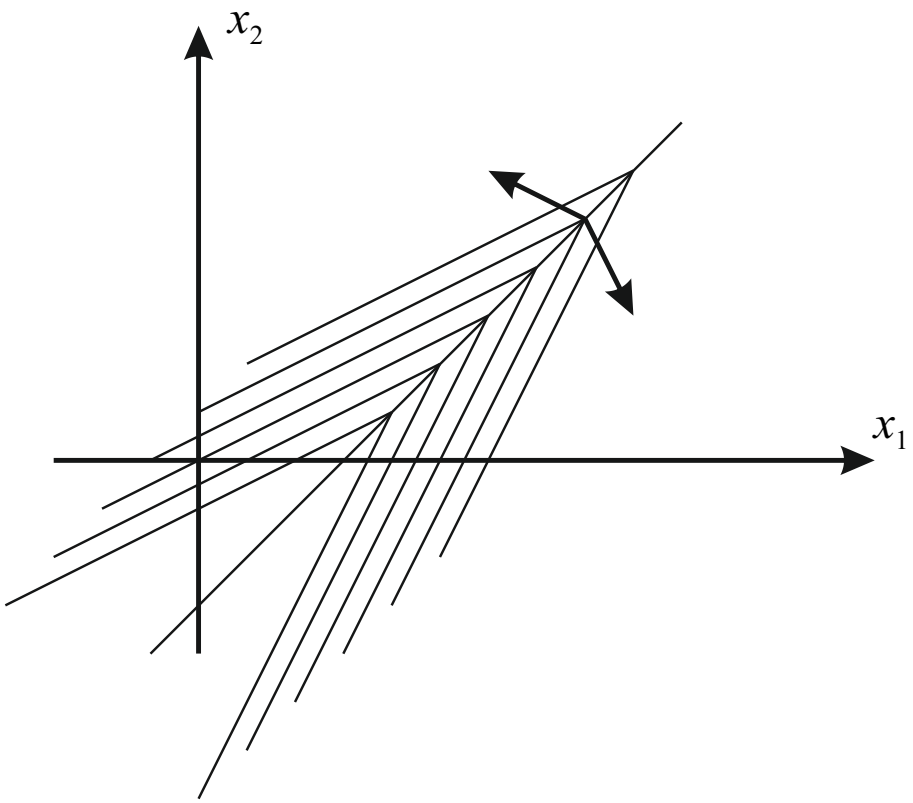
$$\min\{3 - 2x_1 + x_2, 2 + x_1 - 2x_2\}.$$

I due piani si intersecano sulla retta individuata da

$$3 - 2x_1 + x_2 = 2 + x_1 - 2x_2$$

ovvero $x_2 = x_1 - 1/3$. Tale linea è il luogo dei punti di non differenziabilità della funzione.

I subgradienti sono legati alle derivate direzionali, cioè $[-2, 1]^{\mathsf{T}}$ per la funzione $3 - 2x_1 + x_2$ e $[1, -2]^{\mathsf{T}}$ per la funzione $2 + x_1 - 2x_2$. I gradienti sono ortogonali alle curve di livello di ogni funzione lineare,



La figura suggerisce che nel subdifferenziale si possono trovare direzioni di ascesa, ma i subgradienti (supergradienti in questo caso) estremi non sono direzioni di ascesa.

Ne segue che non è possibile determinare il passo α mediante una line search; la successione di valori della funzione obiettivo non è necessariamente monotona.

Una condizione che garantisce la convergenza

$$\sum_{k=1}^{\infty} \alpha^{(k)} = \infty, \qquad \sum_{k=1}^{\infty} \left[\alpha^{(k)}\right]^2 < \infty.$$

Una scelta compatibile con queste condizioni di convergenza è la serie armonica $\alpha^{(k)} = 1/k$.

Teorema: riduzione della distanza dal punto di ottimo.

Consideriamo una funzione convessa $f(\mathbf{x})$ e i suoi sublevel set $L_f(\theta) = \{\mathbf{x} \mid f(\mathbf{x}) \leq \theta\}$. Per ogni α e $\widehat{\mathbf{x}} \notin L(\theta)$, dato un subgradiente $\boldsymbol{\gamma} \in \partial f(\widehat{\mathbf{x}})$, esiste un δ tale che

$$\|\mathbf{x} - (\widehat{\mathbf{x}} - \alpha\boldsymbol{\gamma})\| < \|\mathbf{x} - \widehat{\mathbf{x}}\|,$$

per ogni $\mathbf{x} \in L_f(\theta)$ e $\alpha \in (0, \delta)$.

Dimostrazione. Scegliamo θ tale che il sublevel set corrispondente non è vuoto e un punto $\mathbf{x} \in L_f(\theta)$, che migliora l'obiettivo rispetto a $\widehat{\mathbf{x}}$. Abbiamo

$$\|\mathbf{x} - (\widehat{\mathbf{x}} - \alpha\boldsymbol{\gamma})\|^2 = \|\mathbf{x} - \widehat{\mathbf{x}}\|^2 + 2\alpha\boldsymbol{\gamma}^{\mathsf{T}}(\mathbf{x} - \widehat{\mathbf{x}}) + \alpha^2\|\boldsymbol{\gamma}\|^2,$$

che possiamo riscrivere come

$$\|\mathbf{x} - (\widehat{\mathbf{x}} - \alpha\boldsymbol{\gamma})\|^2 - \|\mathbf{x} - \widehat{\mathbf{x}}\|^2 = 2\alpha\boldsymbol{\gamma}^{\mathsf{T}}(\mathbf{x} - \widehat{\mathbf{x}}) + \alpha^2\|\boldsymbol{\gamma}\|^2. \tag{12}$$

Ma, dato che $\widehat{\mathbf{x}} \notin L(\theta)$ e $\boldsymbol{\gamma}$ è un subgradiente in $\widehat{\mathbf{x}}$, abbiamo

$$f(\mathbf{x}) < f(\widehat{\mathbf{x}}), \quad f(\mathbf{x}) \geq f(\widehat{\mathbf{x}}) + \boldsymbol{\gamma}^{\mathsf{T}}(\mathbf{x} - \widehat{\mathbf{x}}),$$

il che implica $-\boldsymbol{\gamma}^{\mathsf{T}}(\mathbf{x} - \widehat{\mathbf{x}}) > 0$.

Ora si tratta di studiare il segno del termine a destra nella Eq. (12) che, in funzione di α, è una parabola convessa che passa per l'origine e assume valori strettamente negativi per $\alpha \in (0, \delta)$, dove

$$\delta = -\frac{2\boldsymbol{\gamma}^{\mathsf{T}}(\mathbf{x} - \widehat{\mathbf{x}})}{\|\boldsymbol{\gamma}\|^2}$$

è l'altra radice dell'equazione quadratica in α. Da questo segue il risultato. □