Appunti di
Ricerca Operativa e Pianificazione delle Risorse

A cura di:
Francesco Refolli
Matricola 865955

Anno Accademico 2022-2023

Chapter 1

Note sul Corso

todo: segnare delle note

Part I

Teoria

Chapter 2

Modelli nella Ricerca Operativa

2.1 Introduzione

Problemi di Ottimizzazione Dato un problema di ottizzazione:

$$
\begin{aligned}
opt f(x) \\
s.a. x \in X
\end{aligned}
$$

si danno le seguenti definizioni:

$f(x)$ La **funzione obiettivo** e' la funzione della quale cerchiamo un argmento $xinX$ ottimale. L'operatore di ottimizzazione *opt* trova argomenti ottimi o ottimali. Non potendo sempre disporre di ottimi assoluti, si indagano gli ottimali, i quali, come dice il nome, possono essere multipli.

opt Questo operatore e' un placeholder che sta per:

$$
opt = \begin{cases} max \\ min \end{cases}
$$

In particolare si nota che $max f(x) = -min(-f(x))$.
E' possibile ridursi a entrambe le situazioni per sfruttare i vantaggi di eventuali semplificazioni o riduzioni del problema iniziale!

X La **regione d'ammissibilita'** e' un sottoinsieme $X \subseteq \mathbb{R}^n$ che delimita' le possibili soluzioni al problema di ottimizzazione. Le soluzioni che non soddisfano questo requisito sono dette **soluzioni inammissibili**.

x La generica soluzione $x \in X$. In caso di funzione multivariabile, questa assume la forma di una n-upla, come si nota dal dominio della **funzione obiettivo**.

Ottimizzazione Vincolata o non Vincolata

Vincoli Se l'ottimizzazione cerca soluzioni in una regione di ammissibilita' X che sia un sottoinsieme proprio di $\mathbb{R}^n$, ovvero nel caso in cui non coincida, si parla di **Ottimizzazione Vincolata**. Nel caso in cui $X = \mathbb{R}^n$ questa e' definita **Ottimizzazione non Vincolata**.

Esempi Alcuni esempi di ottimizzazione vincolata sono:

- Se $X = \mathbb{Z}^n$ si definisce **Ottimizzazione Intera**.
- Se $X = (0, 1)^n$ si definisce **Ottimizzazine Binaria**.
- Se le variabili appartengono a spazi differenti (es: $\mathbb{N} \times \mathbb{Z} \times \mathbb{R}$), si parla di **Ottimizzazione Mista**.

Programmazione Matematica Se la regione di ammissibilita' X e' espressa tramite vincoli aritmetici (equazioni e disequazioni), si tratta di **Programmazione Matematica**.

Ogni vincolo e' definito come segue:

$$v(x) = \begin{cases} g(x) \geq 0 \\ g(x) = 0 \\ g(x) \leq 0 \end{cases}$$

Di consequenza la regione X e' esprimibile con:
$X = \{x \mid x \in \mathbb{R}^n \wedge g_i(x) \begin{cases} \geq \\ = \\ \leq \end{cases} 0\}$

Esiti di Problemi

- Si dice **Problema Inammissibile** se $X = \emptyset$. Si dice **Problema Illimitato** se non esise un ottimale, in particolare:
- Se $opt = min \wedge foreachc \in R, existsxin \in Xsothatf(x) \leq c$, e' illimitato inferiormente
- Se $opt = max \wedge foreachc \in R, existsxin \in Xsothatf(x) \geq c$, e' illimitato superiormente
- Si dice **Problema con Soluzioni Ottime Multiple** (o Infinite) se tutte le soluzioni ottimali hanno lo stesso grado di ottimizzazione, ovvero se non esiste una soluzione migliore delle altre.
- Si dice **Problema con Soluzione Ottima Unica** nel caso semplice in cui esiste una e una sola soluzione ottimale (= ottimo).

Localizzazione di Ottimi Un ottimo locale $y \in X$ si dice globale se:

$$\begin{cases} f(y) \geq f(x) & opt = max \\ f(y) \leq f(x) & opt = min \end{cases}$$

E' importante notare che un problema di ottimizzazione puo' avere sia piu' di un ottimo locale, che piu' di un ottimo globale. Un ottimo globale e' anche locale. Si mette in evidenza che i vincoli possono assumere caratteristiche non lineari se scomponibili in fattori lineari.

Linearita' Si ricorda che una funzione e' lineare se, per esempio e' nella forma $\Sigma a_i x_i + b$.

Programmazione Lineare I vincoli sono espressi tramite equazioni e disequzioni lineari, la funzione obiettivo e' lineare.

Programmazione Lineare Intera E' un problema di programmazione lineare con regione d'ammissione ristretta a $X = \mathbb{Z}^n$.

Programmazione non Lineare I vincoli o la funzione obiettivo hanno caratteristiche non lineari.

Chapter 3

Introduzione alla Programmazione Lineare

Esempio La Wyndor Glass CO produce vetri di elevata qualita', incluso finestre e porte.

- Impianto 1: produce cornici in alluminio e le altre componenti
- Impiano 2: produce cornici di legno
- Impianto 3: produce i vetri e assembla i prodotti

Si vuole produrre:

- Prodotto 1 - una porta di vetro (impianti 1, 3)
- Prodotto 2 - finestra con doppia apertura (impianti 2, 3)

La domanda e' virtualmente infinita. I prodotti sono raggruppati in lotti da 20 unita'. I tassi sono *Lotti/Settimana*. Determinare i tassi di produzione per massimizzare la produzione e il profitto finale.

Dati

Impianto	Prodotto 1	Prodotto 2	Tempo Produttivo
Impianto 1	1	0	4
Impianto 2	0	2	12
Impianto 3	3	2	18
Profitto	3000	5000	

$$maxZ = 3 \cdot x_1 + 5 \cdot x_2 \tag{3.1}$$
$$x_1 \leq 4 \tag{3.2}$$
$$2 \cdot x_2 \leq 12 \tag{3.3}$$
$$3 \cdot x_1 + 2 \cdot x_2 \leq 18 \tag{3.4}$$
$$x_1 \geq 0 \tag{3.5}$$
$$x_2 \geq 0 \tag{3.6}$$

Considerazioni

- Z = valore di misura di prestazione.
- x_j = livello di attivita' j.
- c_j = incremento del valore della misura di prestazione Z corrispondente all'incremento di un'unita' del valore dell'attivita' x_j.
- b_i = quantita' di risorsa i allocabile alle attivita' x_j.
- a_{ij} = quantita' di risorsa i consumata da ogni attivita' x_j.

In Programmazione Lineare la regione ammissibile e' un Poliedro Convesso in R^n Se la regione e' limitata si dice Politopo.

$$optZ = \Sigma_{j=1}^{n} c_j \cdot x_j \tag{3.7}$$

$$vincoli \Rightarrow \Sigma_{j=1}^{n} a_{ij} \cdot x_j \leq b_i \tag{3.8}$$

$$c_j \rightarrow coefficiente\ di\ costo \tag{3.9}$$

$$a_{ij} \rightarrow termine\ noto\ sinistro \tag{3.10}$$

$$b_i \rightarrow termine\ noto\ destro \tag{3.11}$$

Assunzioni Un problema PL si poggia su quattro assunzioni implicite.

Proporzionalita' Il contributo di ogni attivita' al valore della funzione obiettivo e del vincolo e' proporzionale al livello di attivita'.

Additivita' Il valore della funzione obiettivo e dei vincoli e' dato dalla somma dei contributi individuali delle rispettive attivita'.

Continuita' Qualunque valore delle variabili decisionali in R^n e' accettabile.

Certezza' I parametri di un problema di PL devono essere noti con certezza.

Divisibilita' Questa assunzione non e' sempre garantita, varia in base al problema. Le variabili decisionali possono assumere qualsiasi valore all'interno della Regione di Ammissibilita', inclusi i valori non interi che soddisfino i vincoli. Quindi le variabili decisionali sono variabili continue.
In certi problemi puo' essere necessario avere soluzioni intere perche' la logica non ci consente di spezzare unita' intrinsecamente e logicamente atomice: non si possono assumere 3.5 dipendenti!

Considerazioni Pratiche

Vincoli Prolissi Alcuni vincoli possono includerne altri, in quel caso e' inutile conservarli entrambi, si puo' lasciare quello piu' "forte".

Soluzioni Intere La soluzione del problema di PL non garantisce l'assunzione di divisibilità. In certi casi sono opportune o necessarie per senso logico soluzioni intere, ma non sempre le soluzione di un problema PL le possono garantire.

Se la soluzione ottimale del problema PL e' intera allora e' anche ottimale per il problema considerato. Altrimenti si presentano due strade:

Aggiungere vincoli che garantiscano che la variaibili di decisione siano interi, riducendo il problema PL in PLI (Programmazione Lineare Intera)

Arrotondare la soluzione (per eccesso o difetto, in modo opportuno), ma questo non garantisce l'ottimalita' della soluzione intera cosi' ottenuta.

Chapter 4

Algoritmo del Simplesso

4.1 Teoria

Introduzione Questo algoritmo di tipo greedy permette di risolvere problemi di Programmazione Lineare. E' uno degli algoritmi piu' efficienti che si conoscano. Il tempo nel caso medio e' $\Theta(n)$, lineare rispetto al numero di variabili. Il tempo nel caso peggiore e' $O(e^n)$, esponenziale rispetto al numero di variabili.

Definizioni

Frontiera La **Frontiera del Vincolo** e' un vincolo della regione ammissibile che abbia $\leq$ *oppure* $=$.

Vertice Un **Vertice** e' l'intersezione di due **Frontiere di Vincolo**. I **Vertici Ammissibili** sono quelli che stanno nella **Regione Ammissibile**. Due **Vertici** si dicono adiacenti se condividono $n-1$ **Frontiere di Vincolo**.

Spigolo Uno **Spigolo** e' il segmento che collega due **Vertici** adiacenti. Gli **Spigoli Ammissibili** sono quelli che stanno nella **Regione Ammissibile**.

Test di Ottimalita' Si consideri ogni problema PL tale da ammettere soluzioni ottimali, se una soluzione vertice non ammette soluzioni certice a lei adiacenti con valore della funzione obiettivo Z migliore, allora la soluzione in questione e' ottimale.

Algoritmo

Inizializzazione Scegliere il vertice $(0, 0)$ come soluzione iniziale, (vantaggiosa senza complicazione) se questa fa parte della Regione Ammissibile.

Passo Si valuta lo spostamento nei vertici ammissibili adiacenti. Con il test di ottimalita' si valuta se ci si puo' fermare. Ci si sposta nel vertice che garantisce il valore della funzione obiettivo migliore.

Concetti Chiave

1 Il metodo del simplesso non esplora, ma ispeziona solo i vertici ammissibili adiacenti. Per ogni problema PL trovare una soluzione richiede di trovare il vertice ammissibile migliore. Si richiede che la Regione Ammissibile sia limitata. Il numero di vertici sale esponenzialmente.

2 E' un algoritmo iterativo con due passi, Inizializzazione e Test di Ottimalita'.

3 Quando possibile l'inizializzazione a $(0,0)$ e' preferibile e "ottimale". Si possono applicare algoritmi appositi per garantire l'ammissibilita' della soluzione iniziale.

4 E' piu' vantaggioso ascoltare gli adiacenti che tentare di verificare vertici piu' lontani perche' in minore quantita'.

5 A partire dal vertice corrente compara i risultati ma non calcola ogni volta i valori della funzione, ma i tassi di miglioramento della funzione obiettivo.

6 E' assicurato che si ottiene ad ogni passo una soluzione migliore perche' si cerca il migliore tasso di crescita.

4.2 Procedura Algebrica

La forma standard di un problema PL comprende: opt = max, vincoli in $\geq$ e vincoli di non negativita'.
La forma aumentata consiste in: opt = max, vincoli in = e vincoli di non negativita', introducendo nuove variabili.

La soluzione del problema PL in forma aumentanta (chiamata **soluzione aumentata**) giace sulla frontiera di vincolo. Questo e' utile per utilizzare l'algoritmo del simplesso.

Forma Aumentata Si definisce modello in forma aumentata un modello di programmazione lineare formato da:

- operazione *max Z*
- vincoli in forma $\Sigma_{i=1}^{m} a_{i,j} \cdot x_{i,j} = b_j$
- $\forall i \in [1, m] \mid x_i >= 0$
- $\forall j \in [1, n] \mid b_j >= 0$

E' possibile ricondursi alla forma aumentata a partire da quella standard, ma non e' strettamente necessario.
E' preferibile ricondursi alla forma aumentata direttamente se il modello e' in forma non standard e non e' vitale.

Ottenimento Forma Aumentata Si introduce una **variabile slack** per convertire un vincolo di disuguaglianza in uguaglianza.

$$x_1 \leq 4 \tag{4.1}$$
$$x_1 \geq 0 \tag{4.2}$$

Si trasforma introducendo la variabile x_s.

$$x_1 + x_s = 4 \tag{4.3}$$
$$x_s \geq 0 \tag{4.4}$$
$$x_1 \geq 0 \tag{4.5}$$

E' possibile riscrivere anche la funzione obiettivo in questo modo:

$$max\ Z = 3 \cdot x_1 + 5 \cdot x_2 \tag{4.6}$$

Diventa:

$$max\ Z \tag{4.7}$$
$$Z - 3 \cdot x_1 - 5 \cdot x_2 = 0 \tag{4.8}$$
$$x_s 1, x_2, Z \geq 0 \tag{4.9}$$

Variabili di Base Detto un modello con z variabili (x decisionali, y slack) e w equazioni. Ho $g = z - grado(matrice\ vincoli)$ gradi di liberta'
Quindi fisso g variabili uguali a zero per calcolare le altre. Queste g variabili vengono dette **non di base**. Il resto vengono dette **di base**.
L'insieme delle variabili di base costituiscono una **base**.
Se le variabili di base soddisfano i requisiti di non negativita', la soluzione viene detta **soluzione di base ammissibile**.

Una soluzione di base e' una soluzione della forma aumentata dove le variabili possono essere di base o non di base.

Due soluzioni di base ammissibili sono adiacenti se condividono le stesse variaili non di base tranne una (non i valori, le variabili!).
Questo implica che una variabile non di base e una variabile di base si scambiano di posto.

4.3 Algoritmo

Inizializzazione Inizializzo le g variabili decisionali (e eventualmente slack) a 0.
Secondo il **concetto chiave 3**.
Quindi calcolo la soluzione di base ottimale.

Test di ottimalita' Uso il test di ottimalita' per verificare se la soluzione di base e' ottimale. Se la funzione obiettivo ha coefficienti positivi l'ottimizzazione non e' finita.

Ci si muove qunidi lungo uno degli spigoli, ovvero si scambia una variabile non di base con una di base. Si usano i tassi di crescita per individuare qual'e' lo spostamento piu' vantaggioso. Secondo i **concetti chiave 5 e 6**.

Poi bisogna determinare la quantita' di spostamento. Si usa il test del rapporto minimo.
In pratica si calcola la dipendenza delle variabili di base rispetto alla variabile non di base entrante. Si sceglie il minimo valore che manda a zero una variabile di base, quindi della variabile di base e' uscente. Quindi si ricalcolano i valori delle altre variabili di base risolvendo i vincoli con i nuovi valori delle variabili non di base, della variabile entrante e di quella uscente.
Questo si fa con la procedura di **eliminazione di Gauss-Jordan**.

Essendo la funzione obiettivo espressa tramite un vincolo, anch'essa viene modificata nella riassegnazione della soluzione di base. La variabile slack Z pero' rimane immutata. Z non e' una variabile di base.

4.4 Forma Tabellare

La forma migliore per rappresentare i vincoli della forma aumentata e' usare una matrice. Questa permette di rappresentare tutte le caratterische del problema.

Ho una tabella in cui ogni riga rappresenta un vincolo (esclusi quelli di non negativita') della forma aumentata. La prima riga e' Z - f(x) =0 .

Risoluzione con Deficit di Base La base e' formato da n variabili, dove n e' il numero di equazioni nei vincoli del modello.
Dette k il numero di variabili di slack, un modello necessita' di $m = n - k$ variabili "artificiali" per creare la soluzione di base ammissibile, ovvero creare il tableau iniziale.

Se $m > 0$ bisogna creare m variabili artificiali: $A_1, A_2...A_m$.
Una per ogni vincolo che non contiene variabili di slack.

Le variabili artificiali vengono quindi sommate dentro i vincoli. Quindi si riempiono le righe della tabella. La prima riga e' particolare, viene riempita con soli zero eccetto gli uni nelle colonne delle variabili artificiali.

Quindi alla prima riga della tabella si sottrae la somma delle righe contenenti variabili artificiali.

A questo punto si applica il normale algoritmo del simplesso. Nel momento in cui tutte le variabili artificiali saranno fuori dalla base, si procede a troncare la tabella, eliminando le colonne di tali variabili. Quindi si sostituisce la prima riga con una tradizionale $Z - f(x)$. Dopo la trasformazione, alla prima riga viene sottratta una combinazione lineare delle righe, quelle che avevano una variabile artificiale, che azzeri i coefficienti nella prima riga di tutte le variabili in base.

L'algoritmo del simplesso puo' riprendere normalmente. Questa operazione va effettuata nonappena si presenta l'occasione.

Passo 1 Se la prima riga ha almeno un coefficiente negativo, posso ottimizzare ulteriorimente. Quindi selezione la colonna con il coefficiente strettamente negativo piu' piccolo. Dico che e' la **colonna pivot**.

Passo 2 Cerco le righe che hanno nella **colonna pivot** un coefficiente strettamente positivo. Quindi, detti n_i e c_i il termine noto e il coefficiente di quella riga, cerco la riga tale per cui il rapporto $\frac{n_i}{c_i}$ piu' piccolo.
Chiamo quella riga **riga pivot**.
Il valore nella cella che e' incrocio tra **riga pivot** e **colonna pivot** viene detto **valore pivot**.

Passo 3 Moltiplico la **riga pivot** per $\frac{1}{valore\ pivot}$.
Quindi per ogni riga non pivot:

- se il $c_i > 0$, sottraggo alla riga: $|c_i| * (riga\ pivot)$.
- se il $c_i < 0$, sommo alla riga: $|c_i| * (riga\ pivot)$.

Passo 4 Torno a 1).

Multiple Variabili Entrante Puo' capitare che non ci siano variabili con coefficiente in prima riga piu' piccolo di tutte le altre. In caso di pareggio e' possibile selezionarne una in modo arbitrario. La soluzione ottimare sara' comunque ottenuta tramite l'algoritmo. Non e' possibile sapere in anticipo quale sia la scelta migliore per minimizzare il tempo di soluzione.

Multiple Variabili Uscenti Nel momento in cui nel calcolare il rapporto minimo non esiste una riga sola migliore delle altre, si apre un bivio. Il problema qui e' che fa differenza quale si sceglie. A differenza del caso MVE, qui la soluzione non e' garantita a prescindere dalla scelta della variabile uscente. E' possibile che il metodo del simplesso, a causa di una scelta "errata" in questa situazione, si riconduca ad un loop perpetuo in cui la funzione obiettivo resta costante.

Nessuna Variabile Uscente Se non esistono coefficienti strettamente positivi in una colonna eletta a variabile entrante, il problema PL e' illimitato e non ha soluzioni ottimali.

Molteplici Soluzioni Ottimali Puo' capitare che dopo l'arresto dell'algoritmo dovuto al ritrovo di una soluzione ottimale, sia possibile procedere oltre con il metodo del simplesso trovando una ulteriore soluzione ottimale diversa dalla precedente. Visto che la regione obiettivo e' un poliedro convesso, se esistono almeno due soluzioni ottimali, le soluzioni ottimali esistono e sono infinite. Sono Tutte quelle comprese nel segmento delimitato dalle due trovate in precedenza.

Chapter 5

Teoria della Dualita' e Analisi di Sensitivita'

5.1 Dualita'

Instroduzione Ogni problema di programmazione lineare ha associato un altro problema di programmazione lineare chiamato **duale**. E' utile in programmazione lineare studiare il rapporto tra il problema duale e il problema originale (chiamato **primale**). I **prezzi ombra** sono forniti dalla soluzione ottimale del problema duale. Ma esistono ulteriori applicazioni della teoria della dualita'.

Analisi di Sensitivita' L'**analisi di sensitivita'** e' lo studio dell'effetto dei parametri del problema sulle soluzioni del problema stesso. Infatti molti dei parametri usati in un modello PL sono costruiti con stime di condizioni future o probabili. Dalla teoria del simplesso uno dei prerequisiti e' la conoscenza a priori con ragionevole certezza dei parametri del problema. Questo studio aiuta a verificare e affinare il modello. Inoltre alcuni parametri, come le risorse rese disponibili, sono scelte simil-arbitrarie che vengono gestite dal compartimento manageriale, quindi sono anch'esse perfettibili sotto ogni punto di vista.

Problema Duale Prima di tutto puo' essere utile un confronto visivo tra il **primale** e il relativo **duale**:

$$\begin{array}{cc}
\textit{Primale} & \textit{Duale} \\
max\ Z = \Sigma_{j=1}^{n} c_j \cdot x_j & min\ Z = \Sigma_{i=1}^{m} b_i \cdot y_i \\
\Sigma_{j=1}^{n} a_{ij} \cdot x_j \leq b_i \ \ \forall i \in [1,m] & \Sigma_{i=1}^{m} a_{ij} \cdot y_i \leq c_j \ \ \forall j \in [1,n] \\
x_j \geq 0 \ \ \forall j \in [1,n] & y_i \geq 0 \ \ \forall i \in [1,m]
\end{array}$$

Come si puo' notare, se il problema primale e' di massimo, il duale e' di minimo e vice versa. I coefficienti del primale sono i termini noti del duale e i termini noti del primale sono i coefficienti del duale. I coefficienti di ogni variabile nei vincoli del primale corrisponde a un coefficiente del vincolo del duale. Da notare l'uso di simboli uguali per indicare lo spostamento di variabili.

Da un punto di vista puramente matematico, la matrice del duale e' composta in modo tale da fare supporre che se si sia calcolata la trasposta della matrice del problema primale, infatti:

	x_1	x_2	...	x_n		
y_1	a_{11}	a_{12}	...	a_{1n}	$\leq$	b_1
y_2	a_{21}	a_{22}	...	a_{2n}	$\leq$	b_2
...	...	...	...	...	...	...
y_m	a_{m1}	a_{m2}	...	a_{mn}	$\leq$	b_m
	VI	VI	...	VI		
	c_1	c_2	...	c_n		

Nella risoluzione con forma tabellare i valori delle variabili slack in riga 0 nel problema primale equivale alla soluzione del problema duale e vice versa.

Proprieta' di Dualita' Debole fonte: Teoria della Dualita' - slide prof Stella

se x è una soluzione ammissibile per il problema primale, e y è una soluzione ammissibile per il corrispondente problema duale, allora vale la seguente diseguaglianza:

$$cx \leq yb$$

Proprieta' di Dualita' Forte se x^* è una soluzione ottimale per il problema primale, e y^* è una soluzione ottimale per il corrispondente problema duale, allora vale la seguente eguaglianza:

$$cx^* = y^*b$$

Considerazioni Queste due proprietà, se considerate insieme, implicano che la diseguaglianza vale per soluzioni ammissibili se una o entrambe non sono ottimali per i corrispondenti problemi, l'uguaglianza vale solo se entrambe le soluzioni sono ottimali.

$$cx < yb$$

Ad ogni iterazione, il metodo del simplesso trova una specifica coppia di soluzioni per i due problemi, dove la soluzione del primale è ammissibile mentre quella del duale non è ammissibile, fatta eccezione per l'ultima iterazione, quella che vede l'arresto del metodo del simplesso.

Prolog Time Pericolo fallacia logica!

Per comodita' chiamo:
ammissibile(x) = f(x)
ottimale(x) = g(x)

$$f(x) \wedge f(y) \Leftrightarrow cx < yb \vee cx = yb$$
$$f(x) \wedge g(x) \wedge f(y) \wedge g(y) \Leftrightarrow cx = yb$$

$$A = \{f(x), f(y), \neg g(x), \neg g(y)\}$$
$$B = \{f(x), f(y), \neg g(x), g(y)\}$$
$$C = \{f(x), f(y), g(x), \neg g(y)\}$$

Definisco Modus Ponens = MP = $[(p \rightarrow q) \wedge p] \vdash q$

A Per A, applico MP:

$$(f(x) \wedge f(y) \Leftrightarrow cx < yb \vee cx = yb) \wedge (f(x) \wedge f(y) \wedge \neg g(x) \wedge \neg g(y)) \vdash cx < yb \vee cx = yb$$

$$(f(x) \wedge g(x) \wedge f(y) \wedge g(y) \Leftrightarrow cx = yb) \wedge (f(x) \wedge f(y) \wedge \neg g(x) \wedge \neg g(y)) \vdash \neg(cx = yb)$$

Quindi una semplice risoluzione:

$$((cx < yb) \vee (cx = yb)) \wedge \neg(cx = yb) \vdash (cx < yb)$$

B Per A, applico MP:

$$(f(x) \wedge f(y) \Leftrightarrow cx < yb \vee cx = yb) \wedge (f(x) \wedge f(y) \wedge \neg g(x) \wedge g(y)) \vdash cx < yb \vee cx = yb$$

$$(f(x) \wedge g(x) \wedge f(y) \wedge g(y) \Leftrightarrow cx = yb) \wedge (f(x) \wedge f(y) \wedge \neg g(x) \wedge g(y)) \vdash \neg(cx = yb)$$

Quindi una semplice risoluzione:

$$((cx < yb) \vee (cx = yb)) \wedge \neg(cx = yb) \vdash (cx < yb)$$

A Per A, applico MP:

$$(f(x) \wedge f(y) \Leftrightarrow cx < yb \vee cx = yb) \wedge (f(x) \wedge f(y) \wedge g(x) \wedge \neg g(y)) \vdash cx < yb \vee cx = yb$$

$$(f(x) \wedge g(x) \wedge f(y) \wedge g(y) \Leftrightarrow cx = yb) \wedge (f(x) \wedge f(y) \wedge g(x) \wedge \neg g(y)) \vdash \neg(cx = yb)$$

Quindi una semplice risoluzione:

$$((cx < yb) \vee (cx = yb)) \wedge \neg(cx = yb) \vdash (cx < yb)$$

Tasformazione Primale - Duale Per trovare il duale si consiglia di rifarsi alla seguente tabella:

Primale	Duale
Min Z(x) $= c^T x$	Max W(y) $= b^T y$
$a_i^T x \geq b_i$	$y_i \geq 0$
$a_i^T x \leq b_i$	$y_i \leq 0$
$a_i^T x = b_i$	y_i libera
$x_j \geq 0$	$y^T A_j \leq c_j$
$x_j \leq 0$	$y^T A_j \geq c_j$
x_j libera	$y^T A_j = c_j$

5.2 Sensitivita'

Analisi di termini noto B^{-1} e' la matrice delle colonne slack nel tableau finale.
La colonna dei termini noti del tableau finale e' esattamente $B^{-1}b$

$$\Delta b = \begin{pmatrix} 0 & ... & \Delta b_i & ... & 0 \end{pmatrix}$$

$$B^{-1}\Delta b \geq -B^{-1}b$$

Analisi di coefficienti di costo di variabili non in base y^* e' il vettore dei prezzi ombra.
A e' la matrice dei costi delle variabili dentro i vincoli funzionali. c_f e' il coefficiente della f-esima variabile non di base.

$$\Delta c_f \leq y^* A_j - c_f$$

Analisi di coefficienti di costo di variabili in base B^{-1} e' la matrice delle colonne slack nel tableau finale.
F e' la matrice delle colonne nel tableau iniziale, le quali nel tableau finale corrispondono a variabili che non sono in base.
c_f^t e' il vettore dei coefficienti in riga 0 delle variabili fuori base.

$$\Delta c_b^t = \begin{pmatrix} 0 & ... & \Delta c_i & ... & 0 \end{pmatrix}$$

$$\Delta c_b^t \, B^{-1} \, F \geq -c_f^t$$

Chapter 6

Programmazione Lineare Mista

6.1 Enunciazione

$$opt \;\; f(x) = c^T x$$
$$X = \left\{ x : g_i(x) \begin{Bmatrix} \geq \\ = \\ \leq \end{Bmatrix} 0, i = 1, ..., m \right\} \wedge g_i(x) = a_i^T - b_i$$
$$a \in \mathbb{R}^n, b \in \mathbb{R}^n, c \in \mathbb{R}^n, opt = \begin{Bmatrix} min \\ max \end{Bmatrix}$$

- Se $x \in \mathbb{Z}^n$ allora abbiamo un problema di **Programmazione Lineare Intera** (PLI)
- Se $x \in \{0, 1\}^n$ allora abbiamo un problema di **Programmazione Lineare Binaria** (PB)
- Se $x \in \mathbb{R}^p \times \mathbb{Z}^q$ allora abbiamo un problema di **Programmazione Lineare Mista** (PLM o MIP)

6.2 Variabili Binarie

Risolvono problemi nella modellazione del problema:

- Un problema decisionale del tipo Si'/No, es: Devo costruire o no una fabbrica a Gragnano Trebbiense?
- Un problema di selezione di siti o beni, es: Quale percorso o bene devo scegliere per un servizio?
- Un problema di tempistiche, es: Quando devo cominciare una certa attivita'?

Ma servono anche per includere particolari condizioni.

Vincoli di tipo Either-Or Ho due vincoli $f(x) \leq F$, $g(x) \leq G$ e sono uno dei due vincoli deve essere soddisfatto. Creo una variabile binaria $y \in \{0, 1\}$, considero poi una variabile $M = +\infty$ e formulo il problema cosi:

$$\begin{cases} f(x) \leq F + My \\ g(x) \leq G + M(1 - y) \end{cases}$$

In questo modo se y = 1, allora f(x) sara' sempre soddisfatta e rimarra' selezionata g(x) e vice versa.

K Vincoli su N Dati N vincoli solo K devono essere veri. La modellazione simile alla precedente.

$$\begin{cases} f_1(x) \leq F_1 + My_1 \\ f_2(x) \leq F_2 + My_2 \\ f_3(x) \leq F_3 + My_3 \\ ... \\ f_n(x) \leq F_n + My_n \\ \Sigma_{i=1}^{N} y_i = N - K \end{cases}$$

N possibili valori Nel caso in cui un vincolo abbia una risorsa con piu' di un valore possibile. Creo N variabili binarie y_i, corrispondente alle N risorse d_i.

$$\begin{cases} f(x) = \Sigma_{i=1}^{N} d_i y_i \Sigma_{i=1}^{N} y_i = 1 \end{cases}$$

Costo fisso Se avessi una funzione di costo di una attivita' j formulata cosi':

$$f(x) = \begin{cases} k + cx & x > 0 \\ 0 & x = 0 \end{cases}$$

La minimizzazione di questa funzione sarebbe la seguente:

$$min\ z = (ky + cx) \begin{cases} x - My \leq 0 \\ y \in \{0, 1\} \wedge x >= 0 \end{cases}$$

Rappresentazione binaria di variabili intere Voglio rappresentare x, che so essere nel range [0,u], chiamo N quell'intero tale che $2^N \leq u \leq 2^{N+1}$. Quindi scrivo x = $\Sigma_{i=0}^{N} 2^i y_i$.

6.3 Risoluzione

I problemi di Programmazione Lineare Intera e Binaria non possono essere risolti in modo piu' semplice rispetto alla Programmazione Lineare con Simplesso, anzi sono piu' complessi.

Rilassamento Lineare I problemi richiedono di subire un Rilassamento Lineare, una trasformazione che li "approssima" a problemi di Programmazione Lineare.

Una variabile $x \in \mathbb{Z}$ viene trasformata in:

$$\begin{cases} x \geq 0 \end{cases}$$

Una variabile $x \in \{0, 1\}$ viene trasformata in:

$$\begin{cases} x \leq 1 \\ x \geq 0 \end{cases}$$

Simplesso Quindi si calcola il simplesso del problema rilassato, e si controlla se la soluzione ottimale e' intera. Se non lo e' non e' possibile approssimare con efficacia questa soluzione in intera.

6.4 Matrice Totalmente Unimodulare

Una matrice si dice Totalmente Unimodulare (TUM) se $det(Q) \in \{1, 0, -1\}$ per ogni sua sottomatrice quadrata, di qualunque ordine.

Caso particolare Sia A una matrice tale che $\forall i, j \in A \mid a_{ij} \in \{1, 0, -1\}$ e tale che ogni colonna ha al più due elementi diversi da zero discordi. Questa e' una condizione sufficiente ma non necessaria per asserire che e' una matrice TUM.

esempio preso da slide di A. Gobbi:

$$\begin{array}{ccccccccccccccccccccc}
x_{11} & x_{21} & x_{31} & x_{41} & \dots & x_{p1} & x_{12} & x_{22} & x_{32} & \dots & x_{p2} & x_{13} & x_{23} & \dots & x_{p3} & x_{14} & \dots & x_{p4} & \dots & x_{pq} \\
1 & 1 & 1 & 1 & \dots & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 0 \\
0 & 0 & 0 & 0 & \dots & 0 & 1 & 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 0 \\
0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & 0 & \dots & 0 & \dots & 0 \\
0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 & \dots & 1 & \dots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 1 \\
-1 & 0 & 0 & 0 & \dots & 0 & -1 & 0 & 0 & \dots & 0 & -1 & 0 & \dots & 0 & -1 & \dots & 0 & \ddots & 0 \\
0 & -1 & 0 & 0 & \dots & 0 & 0 & -1 & 0 & \dots & 0 & 0 & -1 & \dots & 0 & 0 & \ddots & 0 & \ddots & 0 \\
0 & 0 & -1 & 0 & \dots & 0 & 0 & 0 & -1 & \dots & 0 & 0 & 0 & \ddots & 0 & 0 & \ddots & 0 & \ddots & 0 \\
0 & 0 & 0 & -1 & \dots & 0 & 0 & 0 & 0 & \ddots & 0 & 0 & 0 & \ddots & 0 & 0 & \ddots & 0 & \ddots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \dots & -1 & 0 & 0 & 0 & \dots & -1 & 0 & 0 & \dots & -1 & 0 & \dots & -1 & \dots & -1
\end{array}$$

6.5 TUM Interezza

Sia A una matrice TUM e b un intero. Il Poliedro $Ax \geq b \cap x \geq 0$ ha solo vertici interi. Quindi ogni sua soluzione ottimale e' automaticamente intera!

6.6 TUM Binarieta'

Il rilassamento lineare di una variabile binaria porta ai seguenti vincoli

$$\begin{cases} x \leq 1 \\ x \geq 0 \end{cases}$$

Quindi $1 \geq x \geq 0$. Se vale la TUM Interezza allora $x \in \mathbb{Z}$.
Questo assicura che x sia binaria, infatti $1 \geq x \geq 0 \wedge x \in \mathbb{Z} \Leftrightarrow x \in \{0, 1\}$

Chapter 7

Branch and Bound

Si generano diversi sottoproblemi e si esplorano per cercare la soluzione migliore. Un PLI o PLB e' di per se' un sottoproblema del PL.

7.1 Programmazione Lineare Binaria

Sia $z^* = f(x^*)$ la soluzione ottima del problema completo e $z = f(x)$ la soluzione ottima di un sotto problema, allora si ha che $f(x) \leq f(x^*)$

- la partizione rispetto al valore delle variabili (branching)
- la determinazione di un limite superiore (bounding)
- l'eliminazione dei sottoproblemi (fathoming)

Branching Si risolve col metodo del simplesso o metodi alternativi il problema corrente. Dopo di che' si osserva la soluzione.

Se la soluzione contiene una variabile con valore non binario (ex: $x = \frac{2}{3}$), si creano i due sottoproblemi fissando rispettivamente $x \leq \lfloor \frac{2}{3} \rfloor$ e $x \geq \lceil \frac{2}{3} \rceil$.

Questo ha l'effetto di fissare nei due sottoproblemi rispettivamente $x = 0$ e $x = 1$, visto che e' un PLI. Si risolvono quindi i sottoproblemi e si preleva la soluzione migliore.

Bounding Se la soluzione ottimale e' una soluzione binaria ammissibile allora e' considerabile la migliore tra tutti i suoi sotto problemi. Quindi questo nodo di esplorazione puo' considerarsi concluso.

Fathoming Se la soluzione e' una soluzione ammissibile non binaria il cui valore della funzione obiettivo e' inferiore della massima soluzione ottimale trovata fin'ora, e' possibile chiudere l'esplorazione di questo ramo ed eliminare la soluzione corrente.

Se il problema e' innammissibile vale lo stesso discorso.

7.2 Esplorazione dei sottoproblemi

Depth First Si esplorano i sottoproblemi in profondita' (con regola left-most o right-most).

Si rischia di esplorare prima tutti i sottoproblemi con soluzioni scadenti.

Best Bound First Si esplorano i sottoproblemi ordinandoli man mano che li si scopre e risolvendo ad ogni turno sempre quello con il miglior valore della funzione obiettivo.

Si ottengono tardi soluzioni ammissibili binarie, pero' si esplorano prima tutte le vie piu' "promettenti".

Mista I nodi vengono scelti alternando i diversi criteri, per evitarne gli svantaggi. Ad esempio, all'inizio si applica una strategia Depth First e, quando si ha una "buona" soluzione ammissibile, si passa alla strategia Best Bound First.

Chapter 8

Programmazione Non Lineare

8.1 Introduzione

Un problema di programmazione non lineare puo' essere scritto cosi':

$$\begin{aligned} & opt f(x) & \\ g_j(x) &\leq 0 & j = 1, ..., m \\ x_i &\geq 0 & i = 1, ..., n \end{aligned}$$

E' simile se non usuale a un problema di programmazione lineare. Tuttavia qua non vale l'assunzione che f e g_j siano lineare.

Esistono diversi modelli a seconda di quali siano le caratterische non lineari di funzione obiettivo e vincoli.

8.2 Ottimizzazione di Portafoglio

Nei problemi di gestione di un portafoglio di titoli finanziari l'obiettivo è massimizzare il rendimento atteso (guadagno) minimizzando il rischio associato all'investimento. La programmazione non lineare viene utilizzata per determinare un portafoglio che, in determinate ipotesi, fornisce un compromesso ottimale tra questi due fattori.

Supponiamo che il portafoglio possa essere formato da un mix di azioni (titoli) da scegliere tra n possibili. Le variabili di decisione x_j (j = 1, 2, ... , n) rappresentano il numero di azioni del titolo j da includere. Siano μ e δ la media (stimata) e la varianza, rispettivamente, del rendimento su ciascuna quota del titolo j, dove δ misura il rischio associato a questo titolo.

Per i = 1,...,n | ($i \neq j$), sia δ_{ij} la covarianza del rendimento dei titoli i e j.

$$\begin{aligned} R(x) &= \Sigma_{j=1}^{n} \mu_j x_j & \text{rendimento totale} \\ V(x) &= \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} \delta_{ij} x_i x_j & \text{rischio totale} \end{aligned}$$

Date queste due funzioni, si sceglie di usare una come vincolo e l'altra come funzione obiettivo. Questa scelta e' vincolata ad una visione filosofica del problema:

- spendere al massimo B, ottenere almeno rendimento L, con minimo rischio

- spendere al massimo B, rischiare al massimo A, con massima resa

In questo caso formalizzo il primo scenario.

$$MinV(x) = \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} \delta_{ij} x_i x_j$$
$$\Sigma_{j=1}^{n} \mu_j x_j \geq L$$
$$\Sigma_{j=1}^{n} P_j x_j \leq B$$
$$x_j \geq 0 \qquad j = 1, ..., n$$

8.3 Tipologie di PNL

Ottimizzazione non vincolata I problemi PNL di ott. non vincolata non hanno vincoli sulla regione ammissibile, quindi l'obiettivo e' semplicemente $maxf(x)$ o $minf(x)$.

Ottimizzazione con vincoli lineari I vincoli $g_j(x)$ sono lineare, ma la funzione obiettivo e' non lineare

l'Ottimizzazione di Portafoglio presentata prima e' un caso particolare di Programmazione Quadratica, che a sua volta e' un caso particolare di ottimizzazione con vincoli lineari.

Ottimizzazione convessa Ogni vincolo $g_j(x)$ e' una funzione convessa. La funzione obiettivo puo' essere sia concava che convessa.

Ottimizzazione non convessa Ogni vincolo $g_j(x)$ e la funzione obiettivo possono essere concavi. Sono i problemi piu' difficili da risolvere perche' possono presentare piu' di un minimo/massimo.

Risoluzione di PNL Non si puo' usare il metodo del simplesso, perche' la soluzione puo' non stare sulla frontiera.

8.4 Problemi Affini

Clustering Il clustering e' una serie di tecniche per analizzare e raggruppare dati in categorie omogenee inferendo delle proprieta' comuni. L'algoritmo piu' usato e' il K-Means.

Dato un insieme di n punti X = $\{x_i\}$ e un insieme di K punti $\{C_k\}$ detti centroidi, si definisce la distanza totale:

$$f(C_1, ..., C_k) = \Sigma_{k=1}^{K} \Sigma_{i=1}^{n} (||C_k - X_i||)^2$$

L'obiettivo del k-means e' minimizzare questa distanza totale. E' un problema PNL.

Classificazione Lineare Nel "Machine Learning" si deifniscono delle tecniche di classificazione che permettono a un programma di classificare degli input in k categorie accomunate da proprieta' che devono essere inferite (come nel Clustering).

Si condideri un insieme di n punti X = $\{x_i\}$, a cui sono associati i relativi target Y = $\{y_j\}$. I target sono le categorie di cui si sa gia' l'appartenenza. (X, Y) e' un dataset di training.

Un classificatore lineare cerca di identificare i parametri $w \in \mathbb{R}^d$ e $b \in R$ relativi ad una suprtficie di separazione $w^t x - b = 0$, tali che le osservazioni della classe A ricadano da un lato della superficie e le osservazioni della classe B vice versa ricadano nell'altra.

Genericamente catalogate in questo modo:

$$\begin{cases} w^t x < b & y_i = A \\ w^t x > b & y_j = B \end{cases}$$

Addestrare una Rete Neurale TODO: inserisci definizione formale di rete neurale

Si disponga del seguente modello di RN:

1. $x_1 \to x_1 * w_1$, $x_2 \to x_2 * w_2$
2. $s = x_1 * w_1 + x_2 * w_2$
3. $y = \frac{1}{1+e^{-s}}$

L'output e' y, gli input x_1 e x_2. Si cercano i parametri w_1, w_2 che diano le migliori previsioni.

Si supponga di avere la seguente tabella input-output (dataset di training)

x_1	x_2	y^r
0	1	0.88
1	0	0.73
0	0	0.5
1	1	0.6
0.5	0.7	0.86

La funzione di distanza ora e' $f(X_1, X_2, Z) = \Sigma_{i=1}^5 (y_i(w_1, w_2) - y_i^r)^2$.

Il problema PNL avra' funzione obiettivo $Minf(X_1, X_2, Y)$.

8.5 Risoluzione di PNL

L'idea e' quella di costruire una sequenza di punti $\{x_k\}$ tali che $lim_{k\to\infty} x_k = x^o$ e $f(x_{k+1}) \leq f(x_k)$. Ovvero ad ogni iterazione k si cerca un punto x_{k+1} che sia migliore di x_k.

A differenza dell'algoritmo del simplesso, questo metodo puo' non terminare in un numero finito di iterazioni. Quindi esistono una serie di criteri che e' possibile valutare per decidere quando occorre fermarsi.

- la soluzione e' sufficientemente accurata, ovvero $\frac{df(x_k)}{dx} \simeq 0$
- si fissa un numero massimo di iterazioni o di tempo computazionale
- il progresso della soluzione k-esima non e' abbastanza soddisfacente
- la soluzione diverge o si verificano cicli

Algoritmi

- dicotomici: algoritmi di ricerca per individuare un determinato valore (per il quale la funzione derivata si annulla) all'interno di un intervallo che ad ogni iterazione viene ridotto. Un esempio e' il metodo della Bisezione.
- di approssimazione: utilizzano approssimazioni locali della funzione. Un esempio e' il metodo di Newton.

8.6 Metodo della Bisezione

Se $f(x)$ e' continua e concava in un intervallo chiuso $[a, b]$, allora considerando un generico punto x_k.

- se $\frac{df(x_k)}{dx} < 0$ allora il punto di ottimo si trova a sinistra di x_k.
- se $\frac{df(x_k)}{dx} > 0$ allora il punto di ottimo si trova a destra di x_k.
- se $\frac{df(x_k)}{dx} \simeq 0$ allora il punto di ottimo e' appossimabile a x_k.

```
procedure BISEZIONE(a,b)
    while true do
        calcolo df = df(x_k)/dx
        if df ≃ 0 ∨ b − a ≤ ϵ then
            return x_k
        else
            if df < 0 then
                b ← x_k
            else
                a ← x_k
            end if
        end if
        x_{k+1} ← (a+b)/2
    end while
end procedure
```

8.7 Metodo di Newton

Si usa lo sviluppo di Taylor:

$$f(x_{k+1}) = f(x_k) + df(x_k)(x_{k+1} - x_k) + 0.5d^2 f(x_k)(x_{k+1} - x_k)^2 + O((x_{k+1} - x_k)^2)$$

$$f(x_{k+1}) \simeq f(x_k) + df(x_k)(x_{k+1} - x_k) + 0.5d^2 f(x_k)(x_{k+1} - x_k)^2$$

$$df(x_k) + d^2 f(x_k)(x_{k+1} - x_k) = 0 \Rightarrow x_{k+1} = x_k - \frac{df(x_k)}{d^2 f(x_k)}$$

---

procedure NEWTON
  while true **do**
    calcolo df $= \frac{df(x_k)}{dx}$
    calcolo d2f $= \frac{d^2f(x_k)}{dx}$
    $x_{k+1} \leftarrow x_k - \frac{df}{d2f}$
    if $|x_{k+1} - x_k| \simeq 0$ **then**
      return x_k
    end if
  end while
end procedure

8.8 Confronto dei due metodi

La bisezione richiesolo solo il calcolo della derivata prima e converge sempre, anche se e' un metodo lento. Mentre il metodo di Newton ha velocita' di convergenza quadratica, ma richiede la derivata seconda e potrebbe pure divergere. Esso infatti puo' fallire se il punto iniziale e' lontano dal punto di ottimo. Quindi deve ssere utilizzato con strategie di ottimizzazione globale.

Chapter 9

PNL Multivariata

9.1 Derivata direzionale

Sia $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una funzione continua e $v \in \mathbb{R}^n$ un vettore direzionale (con $|v| = 1$) di $\mathbb{R}^n$. Allora il limite: $\frac{\delta f(x)}{\delta v} = lim_{h \rightarrow 0} \frac{f(x+hv)-f(x)}{h}$ e' chiamato derivata direzionale di f nella direzione v.

Le derivate parziali sono derivate direzionali con v uguale ad un vettore della base canonica. Ovvero derivate direzionali con $v =< 0, ..., 1, ..., 0 >$.

9.2 Gradiente

Sia $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una funzione reale. Il gradiente di f corrisponde al vettore delle derivate parziali:

$\nabla f(x) = [\frac{\delta f(x)}{\delta x_1}, ..., \frac{\delta f(x)}{\delta x_n}]$.

9.3 Henessiano

La matrice Henessiana rappresenta l'estensione del concetto di derivata seconda per una funzione a piu' dimensioni:

$$\nabla^2 f(x) = \begin{pmatrix} \frac{\delta f(x)}{\delta x_1^2} & \frac{\delta f(x)}{\delta x_1 \delta x_2} & ... & \frac{\delta f(x)}{\delta x_1 \delta x_n} \\ \frac{\delta f(x)}{\delta x_2 \delta x_1} & \frac{\delta f(x)}{\delta x_2^2} & ... & \frac{\delta f(x)}{\delta x_2 \delta x_n} \\ ... & ... & ... & ... \\ \frac{\delta f(x)}{\delta x_n \delta x_1} & \frac{\delta f(x)}{\delta x_n \delta x_2} & ... & \frac{\delta f(x)}{\delta x_n^2} \end{pmatrix}$$

9.4 Matrici positive e negative

- una matrice M e' definita positiva se tutti gli autovalori della matrice sono positivi.
- una matrice M e' definita negativa se tutti gli autovalori della matrice sono negativi.
- una matrice M e' semi-definita positiva se tutti gli autovalori della matrice sono non-negativi.

- una matrice M e' semi-definita negativa se tutti gli autovalori della matrice sono non-positivi.
- per qualsiasi altra condizione la matrice M e' indefinita.

9.5 Legame tra Henessiano e Convessita'

Sia $f : \mathbb{R}^n \to \mathbb{R}$ una funzione reale due volte differenzialbile. Allora per ogni punti x_0 e' possibile calcolare il valore della matrice Henessiana in quel punto

$$\nabla^2 f(x_0) = \begin{pmatrix} \frac{\delta f(x_0)}{\delta x_1^2} & \frac{\delta f(x_0)}{\delta x_1 \delta x_2} & \cdots & \frac{\delta f(x_0)}{\delta x_1 \delta x_n} \\ \frac{\delta f(x_0)}{\delta x_2 \delta x_1} & \frac{\delta f(x_0)}{\delta x_2^2} & \cdots & \frac{\delta f(x_0)}{\delta x_2 \delta x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\delta f(x_0)}{\delta x_n \delta x_1} & \frac{\delta f(x_0)}{\delta x_n \delta x_2} & \cdots & \frac{\delta f(x_0)}{\delta x_n^2} \end{pmatrix}$$

- Se $H_f(x_0)$ e' definita positiva allora la funzione f(x) e' convessa in quel punto.
- Se $H_f(x_0)$ e' definita negativa allora la funzione f(x) e' concava in quel punto.
- Se la funzione f(x) e' convessa in x_0 allora $H_f(x_0)$ e' semi-definita positiva.
- Se la funzione f(x) e' concava in x_0 allora $H_f(x_0)$ e' semi-definita negativa.

9.6 Ricerca di punti di Massimo/Minimo

Poniamo il gradiente della funzione uguale a zero per cercare i punti stazionari, ovvero i punti di massimo e minimo.

$$\nabla f = 0$$

In questo caso si risolvono n equazioni di n incognite. Molto spesso pero' le equazioni che si ottengono sono non-lineari.

9.7 Algoritmi numerici per la ricerca non-lineare

Si prendono in considerazione due algoritmi: il gia' noto Metodo di Newton e il Metodo del Gradiente. Il metodo del gradiente e' concettualmente affine al metodo di newton.

Consideriamo una generica funzione $f : \mathbb{R}^n \to \mathbb{R}$, ed un generico punto $x_0 \in R^n$ in cui la funzione sia differenziabile.

$< x, y >= z$ e' il prodotto scalare $x\dot{y} = z$. Definisco il vettore di discesa un generico vettore $v \in \mathbb{R}^n$ tale che $< v, \nabla f(x_0) >< 0$. Definisco il vettore di salita un generico vettore $v \in \mathbb{R}^n$ tale che $< v, \nabla f(x_0) >> 0$.

Algoritmo generale per il minimo Se voglio cercare un punto di minimo per una funzione:

---

procedure GRADIENT(f)
  $k \leftarrow 0$
  $x_k \leftarrow$ **generico punto**
  $d_k \leftarrow$ **vettore di discesa**
  $a_k \leftarrow$ **step size**
  $x_{k+1} \leftarrow x_k \pm a_k d_k$
end procedure

Step Size ottimale voglio minimizzare/massimizzare il valore della funzione $f(x_{k+1}) = x_k + a_k d_k$ pongo

$$\frac{df(x_k + a_k d_k)}{da_k} = 0$$

uso quindi un metodo di minimizzazione/massimizzazione per pnl monovariati per trovare a_k ottimale.

Metodo del gradiente

$$g(x_k, d_k) = a_k \textbf{tale che} \frac{df(x_k + a_k d_k)}{da_k} = 0$$

---

procedure GRADIENT(f)
  $k \leftarrow 0$
  $x_k \leftarrow$ **punto iniziale**
  while true **do**
    $d_k \leftarrow \nabla f(x_k)$
    $a_k \leftarrow g(x_k, x_k)$
    $x_{k+1} \leftarrow x_k \pm a_k d_k$
    if $|f(x_{k+1}) - f(x_k)| < \epsilon_1 \vee |\nabla f(x_k)| < \epsilon_2$ **then**
      return x_k
    else
      $k \leftarrow k + 1$
    end if
  end while
end procedure

Metodo di newton Si adatta lo sviluppo di Taylor usato nel metodo monovariato in questo modo:

$$f(x_k + \Delta x) = f(x_k) + \nabla f(x_k)\Delta x + 0.5\Delta x \nabla^2 f(x_k)\Delta x$$

Come nel caso monovariato ho un rapporto tra derivate. Δx non e' altro che la direzione di salita/discesa.

$$\Delta x = -\frac{\nabla f(x_k)}{\nabla^2 f(x_k)}$$

Se il metodo di newton converge, lo fa in modo piu' veloce rispetto al metodo del gradiente, ma qua c'e' uno sforzo computazionale maggiore.

```
procedure GRADIENT(f)
    k ← 0
    x_k ← punto iniziale
    while true do
        x_{k+1} ← x_k − ∇f(x_k) / ∇²f(x_k)
        if |f(x_{k+1}) − f(x_k)| < ϵ_1 ∨ |∇f(x_k)| < ϵ_2 then
            return x_k
        else
            k ← k + 1
        end if
    end while
end procedure
```

Chapter 10

PNL Vincolata

10.1 Riduzione del numero di variabili libere

Supponiamo di avere un problema di ottimizzazione soggetto ad un certo numero n di vincoli di uguaglianza.

Nel caso in cui sia possibile esplicitare n variabili in funzione delle restanti $m-n$ variabili utilizzando i vincoli di ugualianza del problema allora possiamo trasformare tale problema in un problema di ottimizzazione non vincolata con $m-n$ variabili

Quindi esplicito le variabili nei vincoli e le sostituisco all'interno del resto del sistema (vincoli funzionali e funzione obiettivo)

Posso esplicitare una variabile solo se facendolo puo' assumere solo un valore.

10.2 Metodo dei moltiplicatori di Lagrange

Sia un generico pnl vincolato con solo vincoli di uguaglianza:

$$opt f(x_1, ..., x_n)$$
$$g_i(x_1, ..., x_n) = 0$$

Consideriamo la funzione L:

$$L(x_1, ..., x_n, \lambda_1, ..., \lambda_n) = f(x_1, ..., x_n) + \Sigma_{i=0}^m \lambda_i g_i(x_1, ..., x_n)$$

Tale funzione prende il nome di Lagrangiana e le variabili λ_i sono chiamate moltiplicatori di Lagrange

Sia $x_o = (x_{1o}, ..., x_{no})$ il punto stazionario di f, allora esistono m moltiplicatori di Lagrange tali che, detto $\lambda_o = (\lambda_{1o}, ..., \lambda_{mo})$, (x_o, λ_o) e' un punto stazionario della Lagrangiana associata.

Condizione di ottimalita' di primo grado Sia $x_o = (x_{1o}, ..., x_{no})$ il punto stazionario di f, allora esistono m moltiplicatori di Lagrange tali che, detto $\lambda_o = (\lambda_{1o}, ..., \lambda_{mo})$, nel punto (x_o, λ_o) il gradiente della Lagrangiana associata si annulla.

Condizione di ottimalita' di primo grado Sia J la matrice dei gradienti dei vincoli (matrice Jacobiana). Consideriamo l'insieme dei vettori $y \in \mathbb{R}^n$ tali che:

$$J(x_o) \cdot y = 0$$

Dove

$$J(x_o) = \begin{pmatrix} \frac{\delta g_1}{\delta x_1} & \dots & \frac{\delta g_1}{\delta x_n} \\ \dots & \dots & \dots \\ \frac{\delta g_m}{\delta x_1} & \dots & \frac{\delta g_m}{\delta x_n} \end{pmatrix}$$

Consideriamo ora la Henessiana della funzione Lagrangiana ristretta alle variabili x_i:

$$H(x_o, \lambda_o) = \begin{pmatrix} \frac{\delta^2 L}{\delta^2 x_1} & \dots & \frac{\delta^2 L}{\delta x_1 x_n} \\ \dots & \dots & \dots \\ \frac{\delta^2 L}{\delta x_1 x_n} & \dots & \frac{\delta^2 L}{\delta^2 x_n} \end{pmatrix}$$

- Se $y^t \cdot H_L(x, \lambda_o) \cdot y > 0$ allora x e' punto di minimo
- Se $y^t \cdot H_L(x, \lambda_o) \cdot y < 0$ allora x e' punto di massimo
- Se x e' un punto di minimo allora $y^t \cdot H_L(x, \lambda_o) \cdot y \geq 0$
- Se x e' un punto di massimo allora $y^t \cdot H_L(x, \lambda_o) \cdot y \leq 0$

10.3 Condizioni di Karush-Kuhn Tucker

Queste condizioni (necessarie ma non sufficienti) generalizzano il metodo dei moltiplicatori di Lagrange ai vincoli di disuguaglianza.

Sia un generico pnl vincolato:

$$\begin{aligned} & opt f(x_1, ..., x_n) \\ g_i(x_1, ..., x_n) &= 0 \\ h_j(x_1, ..., x_n) &\leq 0 \end{aligned}$$

Sia $x_o = (x_{o1}, ..., x_{on})$ un punto di ottimo di f, allora $h_j(x_o)$ puo' essere attivo ($h_j(x_o) = 0$) o inattivo ($h_j(x_o) < 0$).

Indico con $I(x_o)$ l'insieme degli indici dei vincoli attivi.

Sia $x_o = (x_{1o}, ..., x_{no})$ il punto di ottimo di f, allora esistono m moltiplicatori $\lambda_o = (\lambda_{1o}, ..., \lambda_{mo})$ e $u_o = (u_{o1}, ..., u_{op})$, tali che si verifichino le seguenti condizioni:

Condizioni di stazionarieta'

$$\nabla f(x_o) + \Sigma_{i=1}^{m} \lambda_o i \nabla g_i(x_o) + \Sigma_{j=1}^{p} u_o i \nabla h_j(x_o) = 0 \qquad problemidiminimo$$
$$\nabla f(x_o) - \Sigma_{i=1}^{m} \lambda_o i \nabla g_i(x_o) - \Sigma_{j=1}^{p} u_o i \nabla h_j(x_o) = 0 \qquad problemidimassimo$$

Ammissibilita' primale

$$g_i(x_o) = 0 \qquad \forall i = 1, ..., m$$
$$h_j(x_o) = 0 \qquad \forall j = 1, ..., p$$

Ammissibilita' duale

$$u_{oj} = 0 \qquad \forall j = 1, ..., p$$

Condizioni di complementarieta'

$$u_{oj} \cdot h_j(x_o) = 0 \qquad \forall j = 1, ..., p$$

Part II

Esercitazione

Chapter 11

Analisi di Sensitivita'

11.1 1

$$
\begin{aligned}
MaxZ &= 3x_1 + x_2 + 3x_3 \\
2x_1 + x_2 + 3x_3 &\leq 5 \\
x_1 + 2x_2 + 3x_3 &\leq 5 \\
2x_1 + 2x_2 + x_3 &\leq 6 \\
x_1, x_2, x_3 &\geq 0
\end{aligned}
$$

	Z	x_1	x_2	x_3	x_4	x_5	x_6	tn
Z	1	0	$\frac{7}{5}$	0	$\frac{6}{5}$	$\frac{3}{5}$	0	$\frac{27}{5}$
x_1	0	1	$\frac{1}{5}$	0	$\frac{3}{5}$	-$\frac{1}{5}$	0	$\frac{1}{5}$
x_3	0	0	$\frac{3}{5}$	1	-$\frac{1}{5}$	$\frac{2}{5}$	0	$\frac{8}{5}$
x_6	0	0	1	0	-1	0	1	4

$$F = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 0 & 1 \\ 2 & 0 & 0 \end{bmatrix}$$

$$B^{-1} = \begin{bmatrix} \frac{3}{5} & -\frac{1}{5} & 0 \\ -\frac{1}{5} & \frac{2}{5} & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

$$B^{-1}b = \begin{bmatrix} \frac{1}{5} & \frac{8}{5} & 4 \end{bmatrix}$$

$$c^t = \begin{bmatrix} 0 & \frac{7}{5} & 0 & \frac{6}{5} & \frac{3}{5} & 0 \end{bmatrix}$$

$$y^* = \begin{bmatrix} \frac{6}{5} & \frac{3}{5} & 0 \end{bmatrix}$$

Analisi su b_2

$$B^{-1}\Delta b \geq -B^{-1}b$$
$$\Delta b = \begin{pmatrix} 0 & \Delta b_2 & 0 \end{pmatrix}$$

$$\begin{cases} -\frac{1}{5} * \Delta b_2 \geq -\frac{1}{5} \\ \frac{2}{5} * \Delta b_2 \geq -\frac{8}{5} \\ 0 * \Delta b_2 \geq -4 \end{cases}$$

$$\begin{cases} \Delta b_2 \leq 1 \\ \Delta b_2 \geq -4 \end{cases}$$

Concludo che $-4 \leq \Delta b_2 \leq 1$

Analisi su c_1

$$B^{-1}\ F = \begin{bmatrix} \frac{1}{5} & \frac{3}{5} & -\frac{1}{5} \\ \frac{3}{5} & -\frac{1}{5} & \frac{2}{5} \\ 1 & -1 & 0 \end{bmatrix}$$

$$c_b^t\ B^{-1}\ F = \begin{bmatrix} \Delta c_1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{5} & \frac{3}{5} & -\frac{1}{5} \\ \frac{3}{5} & -\frac{1}{5} & \frac{2}{5} \\ 1 & -1 & 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{5}\Delta c_1 \\ \frac{3}{5}\Delta c_1 \\ -\frac{1}{5}\Delta c_1 \end{bmatrix}$$

$$c_f^t = \begin{bmatrix} \frac{7}{5} \\ \frac{6}{5} \\ \frac{3}{5} \end{bmatrix}$$

$$c_b^t\ B^{-1}\ F \geq -c_f^t = \begin{cases} \frac{1}{5}\Delta c_1 \geq -\frac{7}{5} \\ \frac{3}{5}\Delta c_1 \geq -\frac{6}{5} \\ -\frac{1}{5}\Delta c_1 \geq -\frac{3}{5} \end{cases} = \begin{cases} \Delta c_1 \geq -7 \\ \Delta c_1 \geq -2 \\ \Delta c_1 \leq 3 \end{cases}$$

Quindi $-2 \leq \Delta c_1 \leq 3$

Analisi su c_2

$$\Delta c_2 \leq y^* A_2 - c_2$$

$$c_2 \leq \begin{bmatrix} \frac{6}{5} & \frac{3}{5} & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix} = \frac{6}{5} + \frac{6}{5} = \frac{12}{5}$$

$$\Delta c_2 \leq \frac{12}{5} - c_2$$

$$\Delta c_2 \leq \frac{7}{5}$$

Chapter 12

Programmazione Lineare Mista

12.1 Rifornimento d'Acqua

Consegna Una catena di ristoranti ha stipulato un contratto commerciale con un'industria di acque minerali per la fornitura esclusiva di bottiglie d'acqua. L'industria ha a disposizione quattro impianti di S1, S2, S3, S4 con cui dovrà rifornire i tre ristoranti R1, R2 e R3 della catena. Vista la differente distanza tra gli impianti e i ristoranti e i differenti mezzi di trasporto utilizzati, i costi di trasporto euro/bottiglia di acqua da un impianto ad un bar risultano differenti e sono riassunti nella tabella a lato. Sapendo che:

- gli impianti producono giornalmente 125, 150, 130 e 110 bottiglie;
- i tre ristoranti necessitano di 160, 175, 180 bottiglie

	R1	R2	R3
S1	0.4	0.3	0.2
S2	0.2	0.3	0.5
S3	0.1	0.6	0.2
S4	0.5	0.1	0.3

Elaborare un modello di Programmazione Lineare che minimizzi i costi totali di trasporto.

Svolgimento Sia c_{ij} il costo del trasporto tra il i-esimo stabilimento e il j-esimo ristorante. Sia q_{ij} la quantita' di acqua trasportata tra il i-esimo stabilimento e il j-esimo ristorante. Sia p_i la quantita' di acqua prodotta dall'i-esimo stabilimento. Sia a_j la quantita' di acqua accolta dal j-esimo ristorante.

$$
\begin{aligned}
minZ &= \Sigma_{i=1}^{4}\Sigma_{j=1}^{3} c_{ij} q_{ij} & \\
\Sigma_{i=1}^{4} q_{ij} &= a_j & \forall j \\
\Sigma_{j=1}^{3} q_{ij} &= p_i & \forall i \\
q_{ij} &\geq 0 & \forall i, j
\end{aligned}
$$

12.2 Rifornimento d'Acqua 2.0

Riprendiamo il problema dell' ESERCIZIO 1. Supponiamo ora che l'industria voglia che ciascun impianto si prenda carico di almeno il 15% delle spedizioni e che il costo di trasporto generato dall'impianto S1 sia al massimo 1/4 del costo totale. Come possiamo formulare questi vincoli aggiuntivi? Quale altra modifica al modello precedente è necessario imporre?

- gli impianti producono giornalmente 125, 150, 130 e 110 bottiglie;
- i tre ristoranti necessitano di 160, 175, 180 bottiglie

	R1	R2	R3
S1	0.4	0.3	0.2
S2	0.2	0.3	0.5
S3	0.1	0.6	0.2
S4	0.5	0.1	0.3

Svolgimento Sia c_{ij} il costo del trasporto tra il i-esimo stabilimento e il j-esimo ristorante. Sia q_{ij} la quantita' di acqua trasportata tra il i-esimo stabilimento e il j-esimo ristorante. Sia p_i la quantita' di acqua prodotta dall'i-esimo stabilimento. Sia a_j la quantita' di acqua accolta dal j-esimo ristorante.

$$\begin{aligned}
minZ &= \Sigma_{i=1}^{4}\Sigma_{j=1}^{3}c_{ij}q_{ij} & \\
\Sigma_{i=1}^{4}q_{ij} &= a_j & \forall j\\
\Sigma_{j=1}^{3}q_{ij} &= p_i & \forall i\\
4\Sigma_{j=1}^{3}c_{1j}q_{1j} - \Sigma_{i=1}^{4}\Sigma_{j=1}^{3}c_{ij}q_{ij} &\leq 0 & \\
100\Sigma_{j=1}^{3}q_{ij} - 15\Sigma_{i=1}^{4}\Sigma_{j=1}^{3}q_{ij} &\geq 0 & \forall i\\
q_{ij} &\geq 0 & \forall i,j
\end{aligned}$$

12.3 Assegnazione Progetti

Il manager di un'azienda di consulenza deve decidere su quale nuovo progetto far lavorare i suoi tre dipendenti Maria, Carlo e Andrea. I tre progetti (P1, P2, P3) richiedono diverse abilità ed esperienza e il manager ha quindi stimato in quanti giorni ciascuno dei suoi dipendenti potrebbe portare a termine il progetto (tabella a lato). Supposto che ciascun dipendente possa prendersi in carico un solo progetto e vice versa, costruire un modello di Programmazione Lineare che decida come assegnare il personale ai progetti, minimizzando il tempo complessivo per portare a termine i tre progetti.

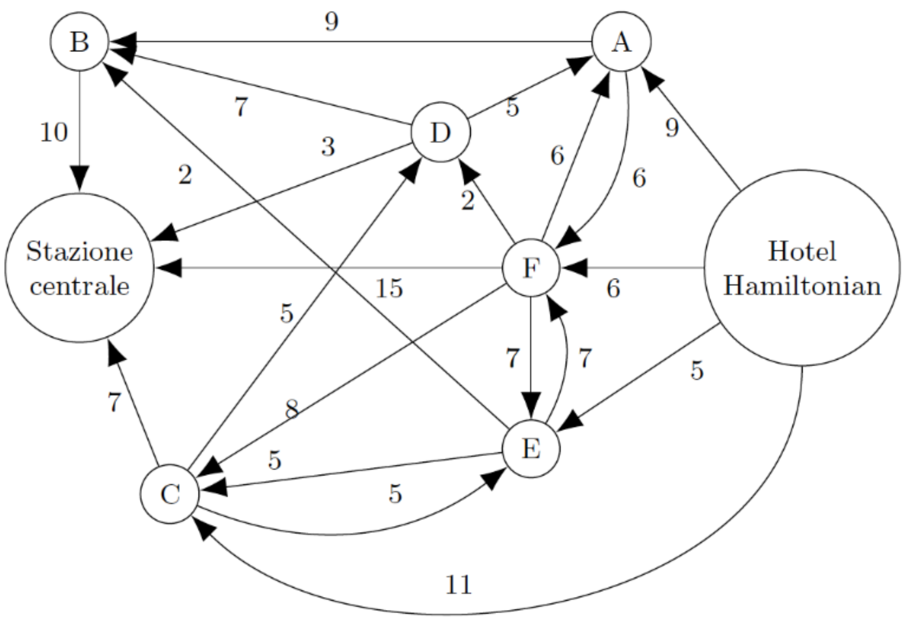
	P1	P2	P3
Maria	15	10	8
Carlo	14	9	4
Andrea	12	6	5

Svolgimento Sia t_{ij} il tempo di completamento per l'i-esimo impiegato e il j-esimo progetto. Sia p_{ij} una variabile booleana che indica che l'i-esimo impiegato lavora sul j-esimo progetto.

$$
\begin{aligned}
minZ &= \Sigma_{i=1}^{3}\Sigma_{j=1}^{3}t_{ij}p_{ij} & \\
\Sigma_{j=1}^{3}p_{ij} &= 1 & \forall i \\
\Sigma_{i=1}^{3}p_{ij} &= 1 & \forall j \\
p_{ij} &\in \{0,1\} & \forall i,j
\end{aligned}
$$

12.4 Corsa in Stazione

È giorno di partenze all'Hotel Hamiltonian e Remo deve correre in Stazione Centrale per prendere il treno che lo riporterà a casa. Il grafo nella slide successiva sintetizza la parte di mappa che interessa a Remo, dove ogni incrocio è un nodo del grafo e ogni arco è un tratto di strada. Supponendo di conoscere per ogni possibile tratto (trai nodi v_i e v_j) il tempo di attraversamento t_{ij} (misurato in minuti, indicato nel grafo) elaborare un modello di Programmazione Lineare che permetta a Remo di arrivare in stazione il prima possibile.



Svolgimento

$$
\begin{aligned}
minZ &= \Sigma_{(i,j)\in E}t_{ij}x_{ij} & \\
\Sigma_{z|(i,z)\in E}x_{iz} - \Sigma_{z|(z,i)\in E}x_{zi} &= 0 & \forall i \in V \setminus \{s,t\} \\
\Sigma_{z|(s,z)\in E}x_{sz} - \Sigma_{z|(z,s)\in E}x_{zs} &= 1 & \\
\Sigma_{z|(t,z)\in E}x_{tz} - \Sigma_{z|(z,t)\in E}x_{zt} &= -1 & \\
x_{ij} &\in \{0,1\} & \forall i,j \in V
\end{aligned}
$$

12.5 Colonnine di Ricarica

Svolgimento dichiaro la funzione $V : Q \times Q \to \{1, 0\}$
che descrive con $V(q_x, q_y)$ se q_x e q_y sono adiacenti.

$$minZ = \Sigma_{i=1}^{|Q|} x_i$$
$$\Sigma_{i=1}^{|Q|} V(q_i, q_j) x_i \geq 1 \qquad \forall j \in [1, |Q|]$$
$$\forall i \in [1, Q] \to x_i \in \{1, 0\}$$

12.6 Gita al Parco Divertimenti

dichiaro la funzione $P : A \to N^+$
che descrive con $P(a_x)$ il punteggio della relativa attrazione.
dichiaro la funzione $T : A \to N^+$
che descrive con $T(a_x)$ il tempo di visita della relativa attrazione.
dichiaro la funzione $F : A \to \{1, 0\}$
che descrive con $F(a_x)$ se la relativa attrazione e' Adrenaline.
dichiaro la funzione $G : A \to \{1, 0\}$
che descrive con $F(a_x)$ se la relativa attrazione e' Fantasy.

$$maxZ = \Sigma_{i=1}^{|A|} P(a_i) * x_i$$
$$\Sigma_{i=1}^{|A|} T(a_i) * x_i \leq 540$$
$$\Sigma_{i=1}^{|A|} F(a_i) * x_i \geq 3$$
$$\Sigma_{i=1}^{|A|} G(a_i) * x_i \leq 1$$
$$x_{11} \leq x_1$$
$$x_i \leq 1 - x_5 \qquad \forall i \in [1, |A|] \wedge F(x_i)$$
$$\forall i \in [1, A] \to x_i \in \{1, 0\}$$

Chapter 13

PNL - Bisezione e Newton

13.1 1

Si determini nell'intervallo $[-3, 0]$ il punto massimo della funzione $f(x) = x^3 - 3x + 5$.

Analiticamente La derivata $g = \frac{df(x)}{dx} = 3x^2 - 3$. Le soluzioni di g sono $(x = \pm 1)$.
La derivata $h = \frac{dg(x)}{dx} = 6x$. Applico le soluzioni di g in h.
$x = 1 \notin [-3, 0]$. $h(-1) = -6$, $x = -1$ e' il punto di massimo.

Con bisezione e massimo 4 iterazioni La derivata $g = \frac{df(x)}{dx} = 3x^2 - 3$.

1. $x_k = -\frac{3}{2}$. $g(x_k) = \frac{15}{4}$. L'ottimo e' a destra di x_k. Il nuovo range e' $[-\frac{3}{2}, 0]$.
2. $x_k = -\frac{3}{4}$. $g(x_k) = -\frac{21}{16}$. L'ottimo e' a sinistra di x_k. Il nuovo range e' $[-\frac{3}{2}, -\frac{3}{4}]$.
3. $x_k = -\frac{9}{8}$. $g(x_k) = \frac{51}{64}$. L'ottimo e' a destra di x_k. Il nuovo range e' $[-\frac{9}{8}, -\frac{3}{4}]$.
4. $x_k = -\frac{15}{16}$. $g(x_k) = -\frac{93}{256}$. L'ottimo e' a sinistra di x_k. Il nuovo range e' $[-\frac{9}{8}, -\frac{15}{16}]$.

Dopo 4 iterazioni $x_k = -\frac{33}{32}$.

13.2 2

Si determini il punto minimo con il metodo della bisezione della funzione $f(x) = x^4 - 14x^3 + 60x^2 - 70x$, con range $[0, 1]$ e precisione $\epsilon = 0.1$.
La derivata $g = \frac{df(x)}{dx} = 4x^3 - 42x^2 + 120x - 70$.

1. $x_k = \frac{1}{2}$. $g(x_k) = -20$. L'ottimo e a di x_k. Il nuovo range e' $[1, \frac{1}{2}]$.
2. $x_k = \frac{1}{4}$. $g(x_k) = -\frac{681}{16}$. L'ottimo e a di x_k. Il nuovo range e' $[0, \frac{1}{4}]$.
3. $x_k = \frac{1}{8}$. $g(x_k) = -\frac{7123}{128}$. L'ottimo e a di x_k. Il nuovo range e' $[0, \frac{1}{8}]$.
4. $x_k = \frac{1}{16}$. $g(x_k) = -\frac{64167}{1024}$. L'ottimo e a di x_k. Il nuovo range e' $[0, \frac{1}{16}]$.
5. $x_k = \frac{1}{32}$. $g(x_k) = -\frac{-543055}{8192}$. L'ottimo e a di x_k. Il nuovo range e' $[0, \frac{1}{32}]$.

13.3 3

Si determini il punto di massimo della funzione $f(x) = -5x^2 + 5x + 4$.

La derivata $g = \frac{df(x)}{dx} = -10x + 5$. La derivata $h = \frac{dg(x)}{dx} = -10$. Il rapporto $r = \frac{g}{h} = \frac{10x-5}{10} = x - \frac{1}{2}$

Newton da $x_0 = 0$

- $x_k = 0$. $r(x_k) = -\frac{1}{2}$. Diventa $x_k = \frac{1}{2}$.
- $x_k = \frac{1}{2}$. $r(x_k) = 0$. Resta $x_k = \frac{1}{2}$.

Mi fermo con $x_k = \frac{1}{2}$.

Newton da $x_0 = 6$

- $x_k = 6$. $r(x_k) = \frac{11}{2}$. Diventa $x_k = \frac{1}{2}$.
- $x_k = \frac{1}{2}$. $r(x_k) = 0$. Resta $x_k = \frac{1}{2}$.

Mi fermo con $x_k = \frac{1}{2}$.

13.4 4

Si determini il punto di massimo della funzione $f(x) = -e^{-x^2}$.

La derivata $g = \frac{df(x)}{dx} = 2e^{-x^2}(x)$. La derivata $h = \frac{dg(x)}{dx} = 2e^{-x^2}(1 - 2x^2)$. Il rapporto $r = \frac{g}{h} = \frac{2e^{-x^2}(x)}{2e^{-x^2}(1-2x^2)} = \frac{x}{1-2x^2}$

Newton con $x_0 = \frac{1}{4}$

- $x_k = \frac{1}{4}$. $r(x_k) = \frac{2}{7}$. Diventa $x_k = -\frac{1}{28}$.
- $x_k = -\frac{1}{28}$. $r(x_k) = \frac{14}{391}$. Diventa $x_k = -1/28 - 14/391 = -\frac{783}{10948}$.

Mi fermo con $x_k = -\frac{783}{10948}$.

Newton con $x_0 = 1$

- $x_k = 1$. $r(x_k) = -1$. Diventa $x_k = 2$.
- $x_k = 2$. $r(x_k) = -\frac{2}{7}$. Diventa $x_k = \frac{16}{7}$.

Mi fermo con $x_k = \frac{16}{7}$.

Chapter 14

Esame

14.1 Metodo del gradiente

Sia la funzione f(x) con un problema di minimo.

$$f(x) = x_1^2 - 2x_1x_2 + 3x_2$$

Si applichi una iterazione del metodo del gradiente con punto di partenza A = (1,1) e con line search in modo esatto.

- Calcolo il gradiente di f(x), ovvero $\nabla f(x) = [2x_1 - 2x_2, -2x_1 + 6x_2]$.
- Si considera il punto di partenza $x_0 = A = (1, 1)$.
- Calcolo la direzione di discesa d_0, applicando il punto di partenza nel gradiente, $d_0 = \nabla f(x_0) = [0, 4]$.
- Definisco la funzione temporanea $g(\alpha) = f(x_0 - \alpha d_0) = 48\alpha^2 - 16\alpha + 4$.
- Calcolo la derivata prima dell funzione $g(\alpha)$ rispetto ad α, $dg(\alpha) = 96\alpha - 16$.
- Calcolo la derivata seconda dell funzione $g(\alpha)$ rispetto ad α, $d^2g(\alpha) = 96$.
- Per $dg(\alpha) = 0$, $\alpha = \frac{1}{6}$.
- Calcolo il nuovo punto $x_1 = x_0 + \alpha d_0 = (a, \frac{1}{3})$.

14.2 Metodo di Newton

Sia la funzione f(x) con un problema di minimo.

$$f(x) = x_1^2 - 2x_1x_2 + 3x_2$$

Si applichi una iterazione del metodo di Newton.

- Calcolo il gradiente di f(x), ovvero $\nabla f(x) = [2x_1 - 2x_2, -2x_1 + 6x_2]$.
- Calcolo il gradiente henessiano di f(x), ovvero $[\nabla^2 f(x)] = \begin{pmatrix} 2 & -2 \\ -2 & 6 \end{pmatrix}$.

- Calcolo l'inverso del gradiente henessiano di f(x), ovvero $[\nabla^2 f(x)]^{-1} = \begin{pmatrix} \frac{3}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{pmatrix}$.
- Calcolo lo step di newton, $d(x) = [\nabla^2 f(x)]^{-1}[\nabla f(x)] = \begin{pmatrix} x_1 & 0 \\ 0 & x_2 \end{pmatrix} = (x_1, x_2)$.
- Si considera il punto di partenza $x_0 = A = (1,1)$.
- Il nuovo punto e' $x_1 = x_0 - d(x_0) = (1,1) - (1,1) = (0,0)$.

14.3 Scrittura delle condizioni KKT

Sia la funzione f(x) con un problema di minimo.

$$f(x) = x_1^2 - 2x_1x_2 + 3x_2$$

Siano i vincoli $g_i(x) \leq 0$

- $x_1^2 + x_2^2 - 4 \leq 0$
- $x_1 - x_2 - 2 \leq 0$
- $-x_1^2 + x_2 + 2 \leq 0$

Si scrivano le condizioni di KKT per il problema vincolato.

- L'equazione di KKT per un problema di minimo con solo vincoli $\leq$ e' $\nabla f(x) + \Sigma\lambda_i \nabla g_i(x) = 0$.
- Calcolo il gradiente di f(x), ovvero $\nabla f(x) = [2x_1 - 2x_2, -2x_1 + 6x_2]$.
- Calcolo il gradiente di $g_1(x)$, ovvero $\nabla g_1(x) = [2x_1, 2x_2]$.
- Calcolo il gradiente di $g_2(x)$, ovvero $\nabla g_2(x) = [1, -1]$.
- Calcolo il gradiente di $g_3(x)$, ovvero $\nabla g_3(x) = [-2x_1, +1]$.
- L'equazione di KKT e' $[2x_1 - 2x_2, -2x_1 + 6x_2] + \lambda_1[2x_1, 2x_2] + \lambda_2[1, -1] + \lambda_3[-2x_1, 1] = [0, 0]$.
- Avendo due variabili avro' due equazioni: 1) $2x_1 - 2x_2 + \lambda_1 2x_1 + \lambda_2 - \lambda_3 2x_1 = 0$.
- Avendo due variabili avro' due equazioni: 2) $-2x_1 - 6x_2 + \lambda_1 2x_2 - \lambda_2 + \lambda_3 = 0$.
- Inoltre valgono le condizioni base per i moltiplicatori lagrangiani, $\lambda_i \geq 0$.
- Quindi trascivo i vincoli funzionali.
- Quindi aggiungo le condizioni di annullamento dei moltiplicatori di Lagrange, $\lambda_i g_i(x) = 0$.

In totale avro'

- $2x_1 - 2x_2 + \lambda_1 2x_1 + \lambda_2 - \lambda_3 2x_1 = 0$

- $-2x_1 - 6x_2 + \lambda_1 2x_2 - \lambda_2 + \lambda_3 = 0$
- $\lambda_i \geq 0$
- $x_1^2 + x_2^2 - 4 \leq 0$
- $x_1 - x_2 - 2 \leq 0$
- $-x_1^2 + x_2 + 2 \leq 0$
- $\lambda_1(x_1^2 + x_2^2 - 4) = 0$
- $\lambda_2(x_1 - x_2 - 2) = 0$
- $\lambda_3(-x_1^2 + x_2 + 2) = 0$

14.4 Applicazione delle condizioni KKT

Part III

Laboratorio

Chapter 15

11-10-2022

15.1

Outlet	Boys	Women	Men	Cost
TV	5	1	3	600
Mag	2	6	3	500
Target	24	18	24	

$$5x + 2y \geq 24 \tag{15.1}$$
$$x + 6y \geq 18 \tag{15.2}$$
$$3x + 3y \geq 24 \tag{15.3}$$
$$C(x, y) = 600x + 500y \tag{15.4}$$

15.2

Gasoline	Vapor	Octane	Price
Regular	≤ 7	≥ 80	9.80
Premium	≤ 6	≥ 100	12

Stock	Vapor	Octane	Barrels
A	8	83	2700
B	20	109	1350
C	4	74	4100

$$\frac{A_i * V_A + B_i * V_B + C_i * V_C}{A_i + B_i + C_i} \leq V_i \tag{15.5}$$
$$\frac{A_i * O_A + B_i * O_B + C_i * O_C}{A_i + B_i + C_i} \geq O_i \tag{15.6}$$
$$\Sigma_{i=0}^{n} A_i \leq Q_A \tag{15.7}$$
$$\Sigma_{i=0}^{n} B_i \leq Q_B \tag{15.8}$$
$$\Sigma_{i=0}^{n} C_i \leq Q_C \tag{15.9}$$
$$C_{scarto} = P_{scarto} * (Q_A - \Sigma_{i=0}^{n} A_i + Q_B - \Sigma_{i=0}^{n} B_i + Q_C - \Sigma_{i=0}^{n} C_i) \tag{15.10}$$
$$C_i = P_i * (A_i + B_i + C_i) \tag{15.11}$$
$$max\ C = C_{scarto} + \Sigma_{i=0}^{n} C_i \tag{15.12}$$

15.3

Zone

1	2	3	4	5	6
12	11	10	9	8	7
13	14	15	16	17	18
24	23	22	21	20	19
25	26	27	28	29	30
36	35	34	33	32	31

Ogni ripetitore copre anche le zone adiacenti.

Chapter 16

18-10-2022

$$Z - 40x - 50y = 0 \tag{16.1}$$
$$x + 2y + s1 = 40 \tag{16.2}$$
$$4x + 3y + s2 = 120 \tag{16.3}$$

$$\left|\begin{array}{c|c|c|c|c} -40 & -50 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 40 \\ 4 & 3 & 0 & 1 & 120 \end{array}\right|$$

1

$$\left|\begin{array}{c|c|c|c|c} -40 & -50 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & \frac{1}{2} & 0 & 20 \\ 4 & 3 & 0 & 1 & 120 \end{array}\right|$$

$$\left|\begin{array}{c|c|c|c|c} -15 & 0 & 50 & 0 & 1000 \\ \frac{1}{2} & 1 & \frac{1}{2} & 0 & 20 \\ \frac{5}{2} & 0 & \frac{3}{2} & 1 & 60 \end{array}\right|$$

2

$$\left|\begin{array}{c|c|c|c|c} -15 & 0 & 50 & 0 & 1000 \\ \frac{1}{2} & 1 & \frac{1}{2} & 0 & 20 \\ 1 & 0 & \frac{3}{5} & \frac{2}{5} & 24 \end{array}\right|$$

$$\left|\begin{array}{c|c|c|c|c} 0 & 0 & 59 & 6 & 1360 \\ 0 & 1 & \frac{1}{5} & \frac{1}{5} & 32 \\ 1 & 0 & \frac{3}{5} & \frac{2}{5} & 24 \end{array}\right|$$

16.1 Esercizio 2

max 3x + 5y x - y <= 1 2x - y <= 4 -2x + y <= 1

Chapter 17

25-10-2022

$$max 3x + 5y \tag{17.1}$$
$$x - y \leq 1 \tag{17.2}$$
$$2x - y \geq 4 \tag{17.3}$$
$$-2x + y = 1 \tag{17.4}$$
$$x, y \geq 0 \tag{17.5}$$

$$min a + 4b + 3c \tag{17.6}$$
$$a - 2b - 2c \geq 3 \tag{17.7}$$
$$-a - b + c \geq 5 \tag{17.8}$$
$$a \geq 0 \tag{17.9}$$
$$b \leq 0 \tag{17.10}$$

2

$$Min : x_1 + 2x_2 - 9x_3 + 5x_4 + 6x_5 \tag{17.11}$$
$$x_1 - 2x_2 + 3x_3 - x_4 + 2x_5 = 30 \tag{17.12}$$
$$x_1 + 3x_2 + 5x_3 + 2x_4 - x_5 <= 10 \tag{17.13}$$
$$x_1, x_2, x_4, x_5 \geq 0, x_3 \leq 0 \tag{17.14}$$

$$Min : x_1 + 2x_2 + 9x_3 + 5x_4 + 6x_5 \tag{17.15}$$
$$x_1 - 2x_2 - 3x_3 - x_4 + 2x_5 = 30 \tag{17.16}$$
$$x_1 + 3x_2 - 5x_3 + 2x_4 - x_5 <= 10 \tag{17.17}$$
$$x_1, x_2, x_3, x_4, x_5 \geq 0 \tag{17.18}$$

$$Max : 30y_1 + 10y_2 \tag{17.19}$$
$$y_1 + y_2 \geq 1 \tag{17.20}$$
$$-2y_1 + 3y_2 \geq 2 \tag{17.21}$$
$$-3y_1 - 5y_2 \geq 9 \tag{17.22}$$
$$-y_1 + 2y_2 \geq 5 \tag{17.23}$$
$$2y_1 - y_2 \geq 6 \tag{17.24}$$
$$y_2 \geq 0 \tag{17.25}$$