Ricerca Operativa

Mascherpa Matteo

a.a. 2024-2025

Indice

1 Introduzione

Con lo scalare improvviso delle imprese della rivoluzione industriale è diventata preponderante la necessità di allocare al meglio le risorse disponibili. Spesso se un'operazione cresce velocemente è facile che ogni compartimento diventi un'entità a sé stante rendendo complessa la gestione. Da questa necessità è emersa la *Ricerca operativa* che è maturata ed è stata sfruttata per la prima volta nella seconda guerra mondiale. Servì per allocare nel miglior modo le poche risorse inglesi per evitare di soccombere ai tedeschi. Con l'avvenire dei computer poi questa materia ha avuto una spinta che la ha portata alla situazione attuale.

1.1 Attributi di una decisione

Una buona decisione è caratterizzata da:

1. ***Efficace***: Raggiunge lo scopo prefissato.
2. ***Efficente***: Nel raggiungere lo scopo prefissato lo fa in modo da sprecare meno risorse possibile
3. ***Tempestiva***: Viene attuata in un arco di tempo coerente con l'orizzonte degli eventi prefissato(strategico(immediato), tattico(medio peridio), operativo(a lungo periodo)).
4. ***Robusta***: Rimane la una buona scelta anche in caso in caso di variazioni trascurabili.
5. ***Giustificabile***: Se si ha il compito di spiegare il motivo della decisione si può fare senza dover usare sotterfugi e depistaggi.

2 Panoramica all'approccio della modellazione

Nonostante la parte più preponderante della materia sia composta da metodi matematici non si deve dimenticare che una parte dello sforzo è la *raccolta dei dati*. Un riassunto delle varie fasi di uno studio sono:

1. Definire il problema e ottenere i dati.

2. Formulare il modello matematico che rappresenti il problema.
3. Sviluppare la procedura per derivare la soluzione del problema dal modello.
4. Test, miglioramento, rifinatura fino a quando si è soddisfatti.
5. Preparare l'applicazione operativa della soluzione.
6. Implementazione della soluzione trovata.

2.1 Definire il problema

Solitamente il problema da risolvere è descritto in termini vaghi, va quindi trovato un modo per formalizzarlo. Solitamente, non si deve dare la "soluzione" ma un insieme di diverse alternative dove il cliente sceglierà quella che più lo aggrada in base a quale soluzione lo attrae di più(Alcuni potrebbero volere una estrema rapidità altri una maggiore robustezza). Per prima cosa si deve discernere i **principi fondamentali**, che saranno quelli che guideranno la scelta finale. Generalmente per le imprese i punti salienti sono: *Proprietario, Imppiegati, Clienti, Fornitori, Governo/nazione.*

2.2 Formulo il modello matematico

Il modello matematico sono versioni idealizzate della realtà per una più facile lavorazione. Esso è composto da ***variabili decisionali***, $(x_1, \ldots, x_n)$ che rappresentano i valori come valori determinati. Le misure delle prestazioni con le ***funzioni obiettivo*** espresse come funzioni aritmetiche, $(P = 3x_1 + 2x_2 + \cdots + mx_n)$. Ci sono anche modi per avere delle restrizioni, chiamate ***vincoli***, espresse come equazioni, $(3x_1+2x_2+\cdots+mx_n < 10)$. Le costanti del modello vengono chiamate ***parametro del modello***. Solitamente, per motivo di incertezza, i parametri non sono altro che grezze stime, concetto a cui ci si riferisce come ***sensitività d'analisi***. Una volta modellato il problema esso non diventa altro che un problema di programmazione lineare. Bisogna evitare però di creare un modello insoddisfaciibile essendo il modello astratto e idealizzato ma con un utilizzo concreto. Un criterio per giudicare un modello è la relativa correttezza in base alle scelte prese. Nello sviluppo del modello è buona pratica da cominciare da un nucleo ristretto e arricchirlo, questo processo si chiama ***arricchimento del modello***. In caso gli obiettivi fossero molteplici essi vengono trasformati in una misura composita chiamata: ***Misura globale delle prestazioni***.

2.3 Sviluppare la procedura

Solitamente un passo semplice dove applicare al modello sviluppato precedentemente un *algoritmo standard*, la parte più complessa è l'*analisi postottimale*. Gli algoritmi infatti restituiscono la soluzione *ottime* per il modello dato, il problema è che il modello dato essendo un astrazione può quasi sempre essere migliorato per ottenere una nuova soluzione migliore. ***Herbert Simon***, nobel per l'economia, ha definito "*satisficing*" (*soddisfacente*) la tendenza dei manager a cercare una soluzione "*abbastanza buona*" e non per forza ottima. Bisogna, nella modifica del modello porre attenzione alla *sensitività delle analisi*, cioè determinare quale parametro modifica considerevomente la soluzione e la sposta di più verso un obiettivo che più si confà alla realtà del problema.

2.4 Test, Miglioramento, e rifinatura

Essendo i casi della realtà, al contrario degli esempi didattici, enormi è delirante aspettarsi che la prima soluzione trovata sia quella che alla fine si andrà ad attuare. Per questo esiste questa fase dove il modello viene messo sotto stress fino ad arrivare ad un livello soddisfacente. Non ci si può neanche aspettare di arrivare alla versione perfetta, si punta perciò alla minimizzazione dei bachi. Questo processo si chiama ***validazione del modello***. Visto l'ammontare di tempo speso nelle fasi precedenti è facile trascurare, per dimenticanza, vari dettagli. Per questo è spesso consigliato che a questo processo venga aggiunto nel team un elemento totalmente avulso dal progetto. Tra gli errori più frequenti ci sono gli errori di incosistenze nelle unità dei valori, *consistenza dimensionale*. Un approccio sistematico è il *test a retrospettiva*. Ciò consiste nell'utilizzo di dati passati per ricostruire vecchi scenari simili e determinare la risposta del modello. Lo svantaggio è che i dati usati per i test sono gli stessi usati per creare il modello quindi non i migliori per trovare le falle.

2.5 Preparazione dell'applicazione

Una volta testato il modello ed essere arrivati ad una versione da adottare allora si deve creare una documentazione estesa che spieghi ogni dettaglio del modello. Il sistema che verrà creato sarà quasi certamente informatizzato. In questa fase si creano le *basi di dati* e *sistemi di gestione delle informazioni*. In caso le decisioni da prendere siano interattive si crea un *sistema di supporto alle decisioni* in modo da avere un rapido accesso alle possibili migliori decisioni.

2.6 Implementazione

La fase critica in cui le scelte prese in precedenza vanno, finalmente, attuate nel modo più accurato possibile al modello ricavato dai mesi di lavoro. Tasselli importanti nell'applicazione sono la comunicazione tra reparti. se tutto viene svolto in cordinazione tra i vari capi reparto il modello potrà essere utilizzato per anni, per questo è importante un riscontro costante per potere avere aggiustamenti minori che vanno documentati e motivati. Grazie a questo si potrà andare avanti fino a quando non cambieranno le necessità che il modello soddisfaceva.

3 Programmazione lineare

In breve l'applicazione più comune per la programmazione lineare è l'allocazione di *risorse limitate*. La *programmazione lineare* usa modelli matematici, dove lineare indica che le funzioni utilizzate devono essere lineari. Un metodo efficente per trovare la soluzione è l'***algoritmo del simplesso***.

3.1 Prototipo esempio

3.1.1 Il problema

Il problema è così formulato:

Problema di PL Schiopparelli & Co

Schiopparelli & CO produce vetri di alta qualità e ha 3 stabilimenti. Lo scopo è massimizzare il guadagno bilanciando la quantità dei vari prodotti.

- **Stabilimento 1:** Crea gli infissi in alluminio.
- **Stabilimento 2/3:** Creano e assemblano le fenistre.

I prodotti sono:

- **Portafinistra:** Alta $2,4m$ con infissi in alluminio.
  Necessita *Stabilimento 1 e 3*
- **Doppia finestra:** $1,2m$ x $1,8m$ con infissi in legno.
  Necessita *Stabilimento 2 e 3*

I dati necessari:

- Numero di ore di lavoro disponibili a settimana in ogni stabilimento per ogni prodotto.
- Numero di ore necessarie per assemblare ogni prodotto.
- Profitto per ogni insieme di prodotti

Modello risultante:

	Tempi di produzione per lotto		
Stabilimento	**Prod. 1**	**Prod. 2**	**Produzione a settimana (h)**
1	1	0	4
2	0	2	12
3	3	2	18
Profitto per lotto	$3,000	$5,000	

3.1.2 Formulazione per la programmazione lineare

Il modello allora per il problema.

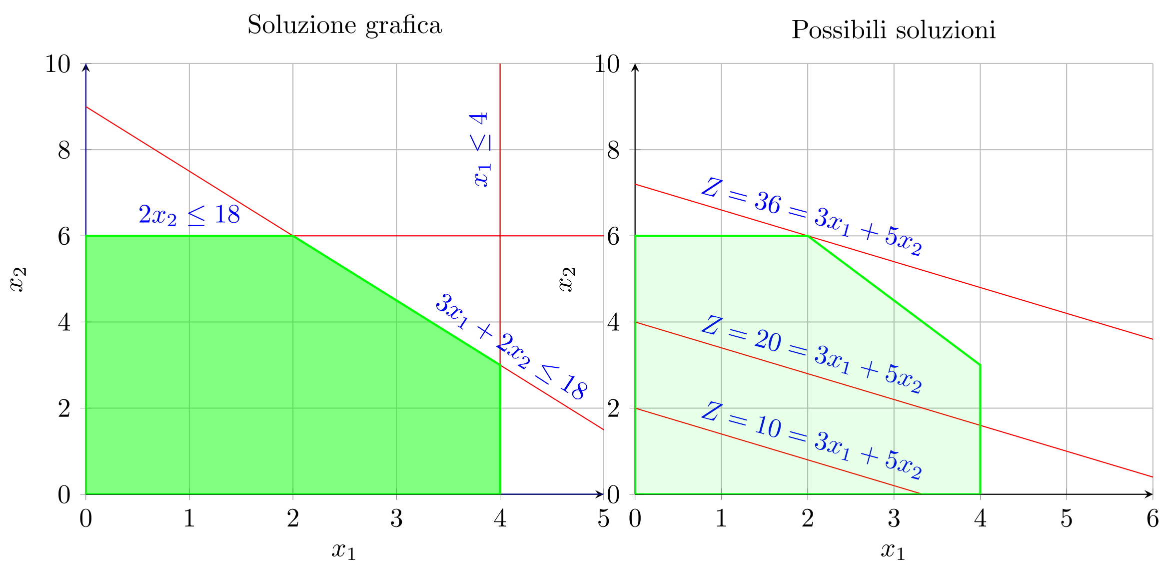
$$\begin{cases} x_1 = \text{N° lotti prod. 1 per settimana} \\ x_2 = \text{N° lotti prod. 2 per settimana} \\ Z = \text{Profitto totale per settimana(in migliaia di \$)} \end{cases}$$

Quindi x_1,x_2 sono *variabili decisionali* del modello. Dalla tabella precedente si ottiene $Z = 3x_1 + tx_2$. L'obiettivo è trovare i valori x_1,x_2 tale che Z abbia il valore massimo. Introduciamo ora i vincoli degli orari dei vari stabilimenti. Lo stabilimento 1: $x_1 \leq 4$, stabilimento 2: $2x_2 \leq 12$, stabilimento 3: $3x_1 + 2x_2 \leq 18$, e infine i *limiti di positività*. Si arriva quindi al sistema:

$$\text{Modello} = \begin{cases} x_1 \leq 4 \\ 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_1 \geq 0 \\ x_2 \geq 0 \\ Z = 3x_1 + 5x_2 \end{cases}$$

3.2 Soluzione dei grafici

Avendo questo problema due *variabili decisionali* su può usare un grafico bidimensionale. Analizziamo la soluzione passo passo. In *blu* ci sono i vincoli di positività $x_1 \geq 0$, $x_2 \geq 0$. In *rosso* il inseriamo i vincoli $x_1 \leq 4$ e $2x_2 \leq 12$ e $3x_1 + 2x_2 \leq 18$. In *verde* l'area delle possibili soluzioni. Si arriva quindi al punto di scegliere un punto che massimizzi $Z = 3x_1 + 5x_2$. Da qui si avanza con il metodo *trial and error* prendendo un valore arbitrario.



(a) Soluzione grafica

(b) Possibili soluzioni

Si noti nel grafico a sinistra che tutte le rette sono parallele, infatti sono tutte forme di una funzione chiamata **slope-intercept form**, in questo caso è $x_2 = -\frac{3}{5}x_1 + \frac{1}{5}Z$. Nel grafico a destra si può vedere che la soluzione massima sia $(x_1 = 2, x_2 = 6)$ portando $Z = 36$.

ATTENZIONE

Nell'esempio fornito si può utilizzare un piano per la scaristà di parametri, ma con il crescere del numero di parametri cresce il numero di dimensioni del grafico. In caso la *regione ammissibile* diventa un poliedro che può essere: *vuoto* in caso non esistano soluzioni ammissibili, *illimitato* in caso in almeno una dimensione non esiste il *valore ottimo*. Esiste poi un almeno un vertice di tale poliedro che corrisponde ad un *valore ottimo*. In ogni caso il poliedro è convesso.

3.3 Modello di programmazione lineare

Abbiamo visto come trovare la soluzione graficamente, ora vediamo l'approccio di programmazione lineare. Cominciamo ora con termini necessari e notazione. *Risorse* e *attività*, dove m indica il numero di risorse utilizzabili e n il numero di attività considerate. Lo scopo in questo caso è trovare i *livelli* delle attività per avere la *migliore prestazione globale*. Di seguito la lista di simboli e il grafico generici per la programmazione lineare.

- Z: Valore totale delle prestazioni.
- x_j: Livello dell'attività $j \in [1, n]$.
- c_j: Incremento in Z per ogni incremento unitario dell'attività j.
- b_i: Ammontare delle risorse i disponibili per l'attività. $i \in [1, m]$.
- a_{ij}: Valore della risorsa i consumata da ogni unità di attività j.

Risorse usate per Unità di attività					
Risorse	**Attività**				**Quantità di risorse disponibili**
	1	**2**	$\cdots$	n	
1	a_{11}	a_{12}	$\cdots$	a_{1n}	b_1
2	a_{21}	a_{22}	$\cdots$	a_{2n}	b_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
m	a_{m1}	a_{m2}	$\cdots$	a_{mn}	b_m
Contributo a Z per unità					
	c_1	c_2	$\cdots$	c_n	

3.4 Forma standard del modello

Dalla tabella precedente si deriva il modello generale(**forma standard**) dal quale si può derivare il modello per il problema che abbia già risolto graficamente.

3.4.1 Modello generale

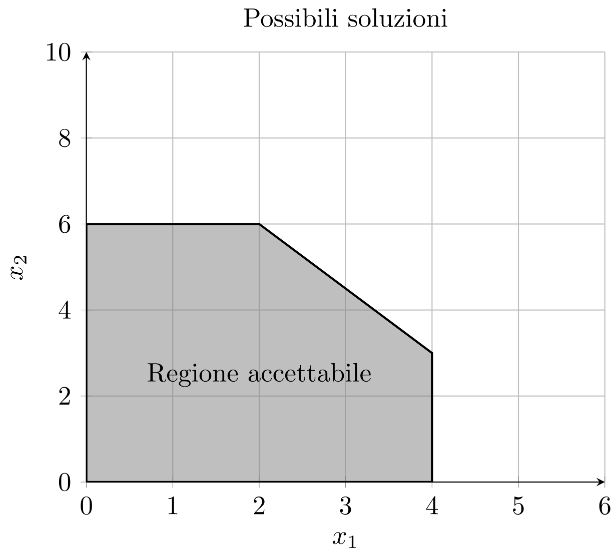
$$
\begin{cases}
\left.Z = c_1x_1 + \cdots + c_nx_n \right\} \text{Funzione obiettivo, da massimizzare} \\
\left.\begin{array}{l}
a_{11}x_1 + a_{12} + \cdots + a_{1n}x_n \leq b_1 \\
a_{21}x_1 + a_{22} + \cdots + a_{2n}x_n \leq b_2 \\
\vdots \\
a_{m1}x_1 + a_{m2} + \cdots + a_{mn}x_n \leq b_m
\end{array}\right\} \text{Vincoli funzionali} \\
\left.\begin{array}{l}
x_1 \geq 0 \\
\vdots \\
x_n \geq 0
\end{array}\right\} \text{Vincoli di non negatività}
\end{cases}
$$

Versioni alternative

- Minimizzare la *funzione obiettivo*.
- Rivoltare i *vincoli funzionali* in modo da avere $\geq$.
- Porre un *vincolo funzionale* a un ugualianza.
- Eliminare *vincoli di non negatività*.

3.4.2 Soluzione del modello

Il termine è utilizzato per indicare il metodo di soluzione del modello. Una **soluzione accettabile** è una soluzione dove tutti vincoli sono soddisfatti. Una **soluzione inaccettabile** è una soluzione dove almeno un vincolo non è soddisfatto. La **regione accettabile** è l'insieme di tutte le possibili soluzioni. La **soluzione ottima** è la soluzione accettabile più favorevole alla funzione obiettivo. É possibile che non esista una soluzione. La maggior parte dei modelli ha una soluzione ottima, ma non è impossibile che ne esistano più di una. Potrebbe accadere che non esistano soluzioni ottimali, che succede in caso non esista una soluzione accettabile o se i limiti non prevengono la funzione obiettivo(Z), al secondo caso ci si riferisce come Z **svincolata** o **obiettivo svincolato**. Una soluzione che si trova a un angolo della regione accettabile si chiama **soluzione accettabile ad angolo**, (CPF).



3.5 Soluzione aritmetica del modello

Metodo per risolvere il modello evitando di utilizzare il grafico. Si parte quindi dal *modello lineare* per portarlo in forma standard e poi risolverlo. Per portare il modello in forma standard tutti i vincoli di *disuguaglianza* devono essere trasformati in *vincoli di ugualianza* e la funzione obiettivo Z vengono poste in forma di minizzazione.
Per poter portare ogni vincolo di *disuguaglianza* ad un vincolo di *ugualianza* si aggiunge una variabile di *slack* per ogni vincolo in gorma di disuguaglianza.

$$\begin{cases} Z = x_1 + 3x_2 - 3x_3 \\ -3x_1 + 2x_2 - 2x_3 \leq 9 \\ -4x_1 - 3x_2 + 3x_3 \leq 8 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 \end{cases} \rightarrow \begin{cases} Z = x_1 + 3x_2 - 3x_3 \\ -3x_1 + 2x_2 - 2x_3 + x_4 = 9 \\ -4x_1 - 3x_2 + 3x_3 + x_5 = 8 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 \\ x_4 \geq 0, x_5 \geq 0 \end{cases}$$

Si passa così da un sistema da m vincoli e n variabili ad un modello con m vincoli e $n + m$ variabili. Si rimane quindi con una matrice che, a meno di vincoli ridondanti, ha **Rango** m. Il sistema ha una soluzione univoca se eliminando gli n gradi di libertà usando n variabili.

Teorema di Rouché Capelli

Data una matrice in forma $A' = [A|b]$ dove A è una matrice m x n e b un vettore di n elementi allora:

1. Se il sistema è risolubile la soluzione è unica se, e solo se, $r(A') = k$ dove k è il numero di incognite del sistema.

2. Se il sistema è risolubile si hanno ∞^j soluzioni(j gradi di infinito) se, e solo se, $j =$

$k - r(A')$.

La soluzione poi si trova estrapolando la base(B) dal modello dove il resto del modello lo chiamiamo N. Il modello allora lo si può riscrivere in forma $[B|N]$. Riscrivo quindi il modello nella forma $B_{x_B} + N_{x_N} = b$ dove x_B e x_N sono le soluzioni del sistema. La soluzione si ottiene dopo aver posto a 0 tutte le n variabili che non appartengono a B. Per farlo si inverte la matrice B, quindi si ottiene: $x_B = B^{-1}b - B^{-1}N_{X_N}$ da qui si comprende il motivo di porre X_N a 0 visto che la soluzione è:$X_N = 0, X_B = B^{-1}b$.

3.5.1 Degenerazione del modello

La degenerazione si ottiene quando si pone a 0 un parametro interno alla base B. Questo porta ad avere più soluzioni basiche decidibili.

3.5.2 Teorema fondamentale della PL

Dato il problema lineare in forma standard

$$Z = \min\{c^T x : Ax = b, \geq 0\}$$

con $A|r(A) = m$

1. Se esiste una *soluzione ammissibile* esiste anche una soluzione di base
2. Se esiste una *soluzione ottima* esiste una soluzione ottima di base.

3.6 Assunzioni sulla programmazione lineare

3.6.1 Proporzionalità

Il contributo di ogni attività al *valore della funzione obiettivo* Z è proporzionale al livello dell'attività x_j rappresentata da $c_j x_j$ in Z. Come il contributo nei vincoli funzionali è proporzionale al livello di x_j rappresentato da prodotto $a_{ij}x_j$. L'assunzione dice dunque esclude ogni esponente diverso da quello neutro(1) per ogni variabile nel modello lineare.

3.6.2 Addività

Ogni funzione nel modello lineare è la somma dei contributi individuali delle rispettive attività.

3.6.3 Divisibilità

Le variabili di decisione in un modello di programmazione lineare possono avere un qualsiasi valore, (anche non interi) per soddisfare i vincoli. Quindi i valori non sono obbligatoriamente interi, quindi un attività può essere svolta a *livelli frazionati*.

3.6.4 Certezza

Il valore assegnato ad ogni parametro si assume sia una costante conosciuta.

4 Risolvere i problemi di PL con l'algoritmo del simplesso:

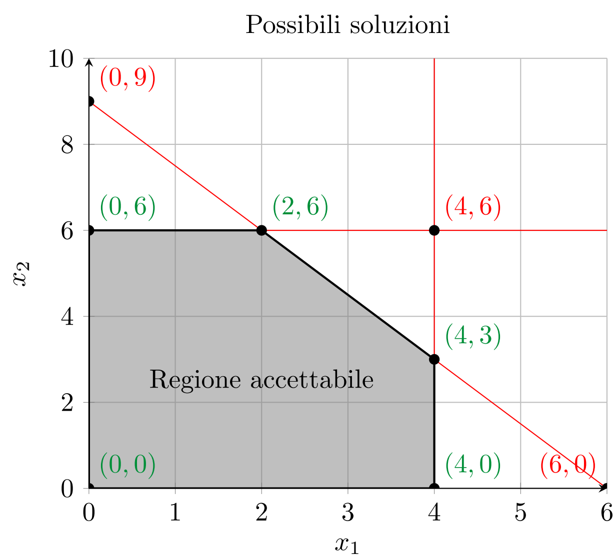
Arrivati a questo punti si deve introdurre l'*algoritmo del simplesso* una procedura generale per risolvere problemi di programmazione lineare sviluppato da *George Dantzig* nel 1947.

4.1 Il concetto

Il metodo del simplesso è un metodo algebrico che si basa su un concetto geometrico. Quindi una prefazione geometrica aiuta a comprendere il concetto sotto. Per evitare di definire un nuovo esempio riprenderemo quello del capitolo precendente di "Schiopparelli & CO".

$$\text{Modello} = \begin{cases} x_1 \leq 4 \\ 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_1 \geq 0 \\ x_2 \geq 0 \\ Z = 3x_1 + 5x_2 \end{cases}$$

(a) Modello del problema "Schiopparelli & CO"



(b) Soluzioni grafiche

Si noti come i punti evidenziati(CFP) non siano altro che l'intersezione di due *vincoli*, alcuni sono al limitare e altri no, rispettivamente quelli *verdi* e quelli *rossi*, se ora questa affermazione lascia il tempo che trova più tardi sarà importante.

Definizione del simplesso

Per ogni problema lineare con n variabili di decisione due CPF sono *adiacenti* se condividono $n-1$ vincoli. Due CPF adiacenti sono uniti da un segmento sulle dimensioni dei vincoli condivisi. Ci si riferisce a questo segmento come: **Spigolo** della regione accettabile.

Per la soluzione si usa il **test di ottimalità**:

Test di ottimalità

Considerando un qualsiasi problema di programmazione lineare che possiede almeno una soluzione ottimale. Se una soluzione CPF non ha *adiacenze* migliori allora quella soluzione è la soluzione migliore.

Test di illimitatezza

Si dice illimitato un problema se non esistono candidati pivot su una colonna con costo ridotto negativo.

4.1.1 L'algoritmo

Algoritmo del simplesso

while (¬ inaccettabile(b,c))∧ (¬ Baseaccettabile(b)) do
Pivot(A,b,c)
if(inaccettabile(b,c)) then
Stop: Problema inammissibile
else
while(¬ Ottimale(c))∧ (¬ Illimitato(A,c)) do
Pivot(A,b,c)
if(Ottimale(c)) then
Stop: Soluzione ottima
else
Stop: Problema illimitato

4.2 Variabili artificiali:

Dato un modello di programmazione lineare in forma standard con n variabili e m vincoli nella forma: Z =min$\{C^T x : Ax = b\}$. Si può introdurre una variabile artificiale $u_i \geq 0$ $\forall i \in [1, m]$. Così si ottiene il problema artificiale Z^a =min$\{e^T u : Ax + Iu = b\}$ con $e^T = [1, \ldots, 1]$. Si può dimostrare che il problema $Z^a \cong Z$ ma la differenza fondamentale è che trovare una base iniziale in Z^a è estremamente semplice e da lì si può risolvere il modello tramite l'algoritmo del simplesso. Infatti con la sostituzione $u_i = b_i - \sum_{j=1}^{n} a_{ij} x_j$ si ottiene una riga del *tablue* con una riga tutta uguale a zero. Si arriva quindi a una soluzione iniziale dove $x = 0$ e $u = b$ ma ammissibile. Si è trovata una soluzione ottimale se si trova $Z^a = 0$ altrimenti Z è inammissibile.

4.2.1 Evitare m passi per rendere non basiche m variabili

Per evitare gli m passi si possono usare dele variabili di slack.

$$\begin{cases} \text{Min}(Z = \sum_{j=1}^{n} c_j x_j) \\ \sum_{j=1}^{n} a_{ij} x_j \geq b_i \forall i \in I_1 \\ \sum_{j=1}^{n} a_{ij} x_j \leq b_i \forall i \in I_2 \\ \sum_{j=1}^{n} a_{ij} x_j = b_i \forall i \in I_3 \\ x \geq 0 \end{cases} \rightarrow \begin{cases} \text{Min}(Z = \sum_{j=1}^{n} c_j x_j) \\ \sum_{j=1}^{n} a_{ij} x_j + \hat{x_i} \geq b_i \forall i \in I_1 \\ \sum_{j=1}^{n} a_{ij} x_j - \hat{x_i} \leq b_i \forall i \in I_2 \\ \sum_{j=1}^{n} a_{ij} x_j = b_i \forall i \in I_3 \\ x, \hat{x} \geq 0 \end{cases}$$

Le variabili $\hat{x_i}$ per $i \in I_1$ soddisfano la prima condizione della forma canonica. Invece sia $h \in I_2 | h = \max_{i \in I_2} b_i$ sottraendo ogni riga nella forma $\leq$ si ottengono i vincoli equivalenti con termine noto non-negativo e variabile $\hat{x_i}$. Si necessita quindi di inserire variabili artificiali soòo per il vincolo h e per i vincoli in I_3.

4.2.2 Metodo "*Big M*"

Al posti di eliminare dal problema il problema delle varibiali aritificiali x è possibile le variabili u e penalizzarle. Si risolve il problema $Z^a = \min(c^T x + w^T u : Ax + Iu = b) | w^T = [M, \dots, M]$ sia un vettore di coefficenti abbastanza grandi in modo che ogni soluzione con u_i sia penalizzata a tal punto da rendere favorevoli solo le soluzioni influenzate da c^T cioè quelle che appartengono anche a Z originale.

4.2.3 Balinsky-Gomery

Quest'algoritmo trascura temporaneamente la funzione obiettivo e minimizza l'inammissibilità del vincolo violato finché si trova una base ammissibile o si scopre che il problema è inammissibile.

4.3 Risolvere il problema d'esempio:

Dato che nel nostro problema $n = 2$ i CPF adiacenti devono condividere un vincolo.

CPF	**Adiacenze**
$(0,0)$	$(0,6), (4,0)$
$(0,6)$	$(2,6), (0,0)$
$(2,6)$	$(4,3), (0,6)$
$(4,3)$	$(4,0), (2,6)$
$(4,0)$	$(0,0), (4,3)$

Esempio di algoritmo del simplesso

Inizializzazione:
Prendo $(0,0)$ come soluzione CPF da esaminare, non c'è un vero e proprio criterio per scegliere in questo caso era solo il più semplice.
1 **Test di ottimalità**:
Fallito, non è ottimale, esiste una soluzione migliore adiacente.
Iterazione 1: Mi muovo verso $(0,6)$ usando i seguenti step.

1. Considero(nella tabella) i due CPF adiacenti $(0,6)$ e $(4,0)$, da cui vien fuori: $(0,6)$:

$Z = 3 \cdot 0 + 6 \cdot 5 = 30$ e $(4, 0) : Z = 3 \cdot 4 + 5 \cdot 0 = 12$. Ora prendo il valore che massimizza Z, quindi $(0, 6)$ mi muoverò nella direzione di x_2 fino a trovare il primo CPF.

2. Mi fermo all'incontrare primo vincolo incontrato: $2x_2 \leq 12$ per evitare di uscire dalla *regione accettabile*.

3. Risolvo l'intersezione del nuovo insieme di vincoli.

2 **Test di ottimalità**:
Fallito, non è ottimale, esiste una soluzione migliore adiacente.

Iterazione 2: Mi muovo verso $(2, 6)$ usando i seguenti step.

1. Considero(nella tabella) i due CPF adiacenti $(2, 6)$ e $(0, 0)$, da cui vien fuori: $(0, 0) : Z = 3 \cdot 0 + 6 \cdot 0 = 0$ e $(2, 6) : Z = 3 \cdot 2 + 5 \cdot 6 = 36$. Ora prendo il valore che massimizza Z, quindi $(2, 6)$ mi muoverò nella direzione di x_1 fino a trovare il primo CPF.

2. Mi fermo all'incontrare primo vincolo incontrato: $3x_1 + 2x_2 \leq 18$ per evitare di uscire dalla *regione accettabile*.

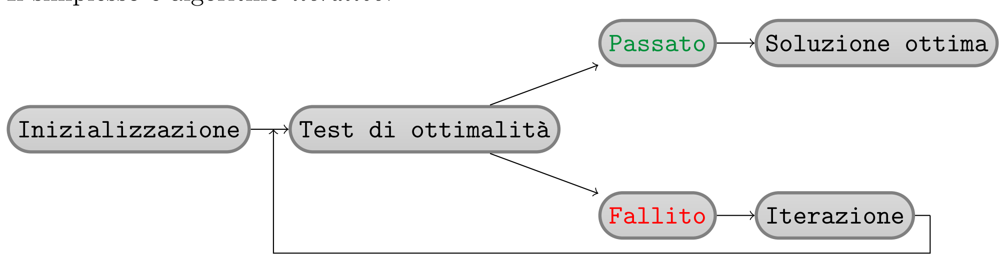
3. Risolvo l'intersezione del nuovo insieme di vincoli.

3 **Test di ottimalità**:
Passato, è ottimale, non esiste una soluzione migliore adiacente. Ho trovato la soluzione migliore.

4.3.1 Concetti della soluzione:

1. Il simplesso si concentra esclusivamnete sui CPF perché se esite una soluzione migliore ne esiste una che coincide con un CPF.

2. Il simplesso è algoritmo *iterativo*.



3. Se possibile per l'inizializzazione si sceglie l'origine in modo da eliminare entropia iniziale.

4. Computazionalmente è più efficente confrontare il CPF adiacenti.

5. Dopo aver trovato un CPF controlla se ne esiste uno adiacente migliore, cerca per ogni paramatro quale deve essere modificato per incrementare di più la funzione obiettivo. O per meglio dire quale incrementa di più il *rateo di incremento di Z*

4.4 Il metodo del simplesso

Per quanto graficamente sia più facile da comprendere queseto metodo non si può tradurre computazionalmente, per poterlo rendere computabile lo si deve formalizzare aritmeticamente. Come nel capitolo precedente per la soluzione aritmetica si pone il modello in forma aumentata.

$$\begin{cases} Z = 3x_1 + 5x_2 \\ x_1 \leq 4 \\ 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_1 \geq 0, x_2 \geq 0 \end{cases} \rightarrow \begin{cases} Z = 3x_1 + 5x_2 \\ x_1 + x_3 = 4 \\ 2x_2 + x_4 = 12 \\ 3x_1 + 2x_2 + x_5 = 18 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0 \end{cases}$$

Si può notare dal nuovo modello che se una delle variabili di slack(*in rosso*) sono poste a zero allora la soluzione si sul limite di un vincolo. Un valore maggiore di 0 invece lo porta nella *regione accettabile* e un valore negativo lo porta nella *regione inaccettabile*. Per meglio comprendere viene comodo un esempio, ma prima si deve introdurre un po di terminologia.

- **Soluzione aumentata:**
  Soluzione per le variabili originali, *di decisione*, che devono essere aumentate dal valore corrispondente della *variabile di slack*.

> **Esempio**
>
> Una soluzione normale sarebbe: $(x_1 : 3, x_2 : 2) \rightarrow (x_1 : 3, x_2 : 2, x_3 : 1, x_4 : 8, x_5 : 5)$

- **Soluzione basica:**
  È una soluzione CPF aumentata che può trovarsi all'interno o all'esterno della regione accettabile, che porta al condetto di *soluzione basica accettabile* una soluzione basica ha le seguenti proprietà:

  1. Ogni variabile può essere o *basica* o *non basica*.
     L'insieme dei coefficenti delle variabili basiche si scrive B_{x_B}, l'insieme dei coefficenti delle variabili non basiche si scrive N_{x_N}. Questo porta il modello a poter essere riscritto così: $B_{x_B} + N_{x_N} = b$
  2. Il numero di *variabili basiche* è uguale al numero di vincoli funzionali, parimenti il numero di *variabili non basiche* sono il grado di libertà.
  3. Le *variabili non basiche* sono settate a 0.
  4. I valori delle variabili basiche sono ottenute simultaneamente.
  5. Se le *variabili basiche* soddisfano i vincoli di non negatività alla la soluzione basica è accettabile.

- **Soluzione basica accettabile:**
  Come da nome una *soluzione basica* ma con il vincolo di trovarsi nella regione accettabile.

- **Gradi di libertà:**
  Differenza tra in numeri di variabili, (*variabili di decisione + variabili di slack*) e il numero di *vincoli funzionali.* Indica il numero di variabili che possono essere poste ad un valore aribtrario, 0, per poter ottenere una soluzione per le variabili rimanenti.

4.5 L'algebra dell'algoritmo del simplesso

In questo paragrafo verranno rielaborati i punti precedenti per illustrare la connessione tra interpretazione geometrica e algebrica. Ci si riferisce allo stesso problema d'esempio usato precedentemente, in caso di dubbi sui passaggi si faccia riferimento lì.

Algebra dell'esempio del simplesso

- **Inizializzazione**:
  Scelgo di mettere x_1, x_2 al di fuori della base, in modo da trovare immediatamente i valori di x_3, x_4, x_5.

$$\begin{cases} Z = 3x_1 + 5x_2 \\ 0 + x_3 = 4 \rightarrow x_3 = 4 \\ 0 + x_4 = 12 \rightarrow x_4 = 12 \\ 0 + 0 + x_5 = 18 \rightarrow x_5 = 18 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0 \end{cases}$$

  Si trova quindi la soluzione $(0, 0, 4, 12, 18)$.

- 1**Test di ottimalità**
  La funzione obiettivo $Z = 3x_1 + 5x_2 = 0$. Essendo però i ratei di incremenento positivi i valori possono incrementare. Visto che tutte le altre BF esaminate hanno almeno un parametro maggiore di 0 allora il valore attuale non è il migliore.

- **Iterazione 1:**
  1. **Determinare la direzione**: L'incrementare di una variabile non basica corrisponde al muoversi in una delle dimensioni. Data la funzione obiettivo $Z = 3x_1 + 5x_2$ so che il coefficente di x_1 è 3 e x_2 è 5, prendo il maggiore: 5.
  2. **Dove fermarsi**
     Si vuole incrementare il più possibile il valore di un parametro ma rimanendo nella regione accettabile.

$$\begin{cases} Z = 3x_1 + 5x_2 \\ x_1 + x_3 = 4 \rightarrow x_3 = 4 \\ 2x_2 + x_4 = 12 \rightarrow x_4 = 12 - 2x_2 \\ 3x_1 + 2x_2 + x_5 = 18 \rightarrow x_5 = 18 - 2x_2 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0 \end{cases}$$

     Dal quale si estrapolano i vincoli:
     $x_4 = 12 - 2x_2 \geq 0 \Rightarrow x_2 \leq \frac{12}{2} = 6$

$x_5 = 18 - 2x \geq 0 \Rightarrow x_2 \leq \frac{18}{2} = 9$ Porto x_4 a 0 ottengo $x_2 = 6$ incrementarlo ulteriormente violerebbe i vincoli. Ci si riferisce a questi calcoli come **test di rateo minimo**. Quando una variabile viene portata "esce" dalla base e lascia entrare la variabile che viene incrementata.

3. **Risolvere per la nuova BF**:
L'incrementare di $x_2 = 0 \rightarrow x_2 = 6$ sposta il problema ad una nuova BF.

	Soluzione BF iniziale	Nuova soluzione BF
Variabili non-basice	$x_1 = 0, x_2 = 0$	x_1, x_4
Variabili basiche	$x_3 = 4, x_4 = 12, x_5 = 18$	$x_3 = ?, x_2 = 6, x_5 = ?$

Una volta svolta questa operazione di sostituzione, x_2 ha sostituito x_4 nella base. Per ottenere lo stesso risultato nella forma algebrica si svolgono semplici operazioni algebriche. *Moltiplicazione* e *divisioni* per un numero diverso da 0 e *Sommare* o *sottrarre* un'equazione per un altra. Si arriva al risultato desiderato, cioè, il modello trasformato, tramite l'*eliminazione gaussiana* arrivando a:

$$\begin{cases} Z - 3x_1 + \frac{5}{2}x_4 = 30 \\ x_1 + x_3 = 4 \\ x_2 + \frac{1}{2}x_4 = 6 \\ 3x_1 - x_4 + x_5 = 6 \end{cases}$$

- 2 **Test di ottimalità** Data la funzione obiettivo $Z = 30 + 3x_1 - \frac{5}{2}x_4$ tutti e due i valori non basici posso incrementati ma scelgo x_1 visto che ha un coefficente positivo.

- **Iterazione 2:** Come nell'iterazione precedente si sceglie una direzione, forzata dalla negatività del coefficente di x_4. Dal modello si estrapola:
$x_3 = 4 - x_1 \geq 0 \Rightarrow x_1 \leq 4 = 4$
$x_2 = 6 \geq 0 \Rightarrow$ Senza limiti
$x_5 = 6 - 3x_1 \geq 0 \Rightarrow x_1 \leq \frac{6}{3} = 2$
Nuovamente al terzo step dell'iterazione si usano l'*eliminazione Gaussiana* per scambiare x_1 e x_5 che restistuisce:

$$\begin{cases} Z - \frac{3}{2}x_4 + x_5 = 36 \\ x_3 + \frac{1}{3}x_4 - \frac{1}{3}x_5 = 2 \\ x_2 + \frac{1}{2}x_4 = 6 \\ x_1 - \frac{1}{3}x_4 + \frac{1}{3}x_5 = 2 \end{cases}$$

La soluzione attuale è quindi $Z = -36 - \frac{3}{2}x_4 - x_5$, visto che incrementando un qualsiasi valore della funzione obiettivo peggioreremmo il valore di Z possiamo dire con cercettezza che abbiamo trovato una soluzione ottima.

4.6 Soluzione tabulare

Nonostante ne si sia discusso estensivamente c'è un altro modo per visualizzare la risoluzione del modello. La soluzione *geometrica* serve ad avere una rapida visualizzazione del problema, mentre la versione aritmetica è la più adatta per una risoluzione tramite un programma. C'è una via di mezzo che concilia la semplicità di comprensione tra uomo e computer, la versione **tabulare**. In questo contesto la tabella che useremo si chiama *tablue*.

$$\begin{cases} Z - 3x_1 - 5x_2 = 0 \\ x_1 + x_3 = 4 \\ 2x_2 + x_4 = 12 \\ 3x_1 + 2x_2 + x_5 = 18 \end{cases} \Rightarrow$$

Variabili basiche	Coefficenti						Vincoli
	Z	$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{x_3}$	$\mathbf{x_4}$	$\mathbf{x_5}$	
Z	1	-3	-5	0	0	0	0
x_3	0	1	0	1	0	0	4
x_4	0	0	2	0	1	0	12
x_5	0	3	2	0	0	1	18

4.6.1 Riassunto del metodo del simplesso

Visto che sarebbe inutile ripetere quello che è stato detto nel paragrafo precedente si darà per scontato che si sia compreso i concetti e si evidenzieranno solo riassuntivamente i passaggi che non mutano.

Inizializzazione:

Si introducono le variabili di slack e si decide che le variabili di decisione siano al di fuori della base(rosso) e complementarmente le variabili di slack siano nella base(blue).

Test di ottimalità:

La soluzione BF è ottima se e solo se ogni coefficente nella funzione obiettivo è non negativa, ergo il test di ottimalità non è passato.

Variabili basiche	Coefficenti						Vincoli	Rateo
	Z	$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{x_3}$	$\mathbf{x_4}$	$\mathbf{x_5}$		
Z	1	-3	-5	0	0	0	0	
x_3	0	1	0	1	0	0	4	
x_4	0	0	2	0	1	0	12	$\frac{12}{2} = 6$
x_5	0	3	2	0	0	1	18	$\frac{18}{2} = 9$

I passaggi da svolgere sono i segunti e generano il tablue successivo:

1. Dividere la *riga pivot* per il valore del *numero di pivot*.
2. Sommare la nuova riga di pivot per il coeffincente di pivot, ($|$n pivot$| \cdot$ riga), ad ogni altra riga che abbia nella colonna di pivot un valore negativo.
3. Per ogni riga che nella colonna di pivot ha un valore positivo, sottrarre la nuova riga di pivot per il coeffincente di pivot, ($|$n pivot$| \cdot$ riga).

Attenzione

Si può notare, che le variabile basiche nel *tablue* hanno la forma della matrice I di identità. Infatti dopo la prima iterazione dove x_2 prende il posto di x_4 i vettori che formano la base della matrice sono le nuove variabili di base: (x_2, x_3, x_5)

Variabili basiche	**Coefficenti**						**Vincoli**
	Z	$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{x_3}$	$\mathbf{x_4}$	$\mathbf{x_5}$	
Z	1	-3	0	0	$\frac{5}{2}$	0	30
x_3	0	1	0	1	0	0	4
x_2	0	0	1	0	$\frac{1}{2}$	0	6
x_5	0	3	0	0	-1	1	6

Da qui, si continua con il normale svoglimento dell'algoritmo del simplesso, nello stesso modo evidenziato precendentemente, quindi si eviteranno commenti che ormai potrebbero essere ridondanti. Nell'ultimo tablue, visto che x_1, x_2 sono nella base si è arrivati alla soluzione ottima che il valore del vincolo, in verde.

Numero iterazione	**Variabili basiche**	**Coefficenti**						**Vincoli**
		Z	$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{x_3}$	$\mathbf{x_4}$	$\mathbf{x_5}$	
0	Z	1	-3	5	0	0	0	0
	x_3	0	1	0	1	0	0	4
	x_4	0	0	2	0	1	0	12
	x_5	0	3	2	0	0	1	18
1	Z	1	-3	0	0	$\frac{5}{2}$	0	30
	x_3	0	1	0	1	0	0	4
	x_2	0	0	1	0	$\frac{1}{2}$	0	6
	x_5	0	3	0	0	0	1	6
2	Z	1	0	0	0	$\frac{3}{2}$	1	36
	x_3	0	0	0	1	$\frac{1}{3}$	-$\frac{1}{3}$	2
	x_2	0	0	1	0	$\frac{1}{2}$	0	6
	x_1	0	1	0	0	-$\frac{1}{3}$	$\frac{1}{3}$	2

4.7 Forma canonica

Dato un modello in forma standard, nella forma di una funzione obiettivo ($Z = C_B^T x_B + C_N^T x_N$) e un matrice che rappresenta le funzioni di vincolo($B_{X_B} + N_{X_N} = b$) si può riscrive il modello in modo di poter trovare x_B.

$$\begin{cases} Z = C_B^T x_B + C_N^T x_N \\ B_{X_B} + N_{X_N} = b \\ X_B, X_N \geq 0 \end{cases} \xRightarrow{\text{Moltiplico per } B^{-1}} \begin{cases} Z = C_B^T x_B + C_N^T x_N \\ I x_B + (B^{-1}N) x_N = B^{-1}b \quad \rightarrow x_B = (B^{-1}N)x_n = B^{-1}b \\ X_B, X_N \geq 0 \end{cases}$$

Si è ora trovata la funzione per trovare il valore di x_B. Ora l'idea è vedere cosa succede se insieriamo nella funzione obiettivo il valore dei coefficenti della base.

$$\begin{cases} Z = C_B^T x_B + C_N^T x_N \\ I x_B + (B^{-1}N) x_N = B^{-1} b \\ X_B, X_N \geq 0 \end{cases} \xRightarrow{\text{Inserisco } x_B} \begin{cases} Z = C_B^T B^{-1} b + (C_N^T - C_B^T B^{-1} N) x_N \\ I x_B + (B^{-1}N) x_N = B^{-1} b \\ X_B, X_N \geq 0 \end{cases} \Rightarrow \begin{cases} Z = Z_B + \overline{C}_N^T x_N \\ I x_B + \overline{N} x_N = \overline{b} \\ X_B, X_N \geq 0 \end{cases}$$

5 Teoria della dualità

Uno dei concetti più importanti nella programmazione lineare è il concetto di **dualità**. Ogni problema di programmazione lineare ha associato un altro problema chiamato **dualità**. La relazione tra due problemi duali, dove quello originale si chiama **primale** è molto utile in vari ambiti. Uno degli usi fondamentali è l'implentazioine della *analisi della sensitività*. Questa connessione deriva dal fatto che i valori del modelli primale sono stime di condizioni future, l'effetto sulle soluzioni prevale quando in realtà andrebbe ultreriormente investigato.

5.1 Essenza della teoria della dualità:

Dato un problema primale il problema duale si trova invertendo: funzione obiettivo e vincoli funzionali. (Nota bene: n è il numero di parametri e m è il numero di vincoli)

$$\overbrace{\begin{cases} \text{Max}(Z = \sum_{j=1}^{n} c_j x_j) \\ \sum_{j=1}^{n} a_{ij} x_j \leq b_b | i \in [1, m] \\ x_j \geq 0 | j \in [1, n] \end{cases}}^{\text{Prob. primale}} \qquad \overbrace{\begin{cases} \text{Min}(Z = \sum_{j=1}^{m} b_i y_i) \\ \sum_{i=1}^{m} a_{ij} y_i \geq c_j | j \in [1, n] \\ y_i \geq 0 | j \in [1, n] \end{cases}}^{\text{Prob. Duale}}$$

Si noti che dove il problema priamrio è in forma di massimizzazione allora il problema duale è in forma di minimizzazione. Il problema duale usa anche gli stessi parametri. Le differenze sono le seguenti.

- I coefficenti della funzione obiettivo del problema primale si trovato come parametri nelle funzioni di vincolo del problema duale.
- I parametri del problema primale sono i coefficenti delle funzioni obiettivo del problema duale.
- I coefficenti di una variabile nelle funzioni di vincolo di un problema primale sono i coefficenti della funzione vincolo del problema duale.

Queste differenze si notato al meglio in notazione matriciale.

$$\overbrace{\begin{cases} \text{Max}(Z = {\color{red}c}x) \\ {\color{blue}A}x \leq {\color{green}b} \\ x \geq 0 \end{cases}}^{\text{Prob. primale}} \qquad \overbrace{\begin{cases} \text{Min}(W = y{\color{green}b}) \\ {\color{blue}A^T} y \geq {\color{red}c} \\ y \geq 0 \end{cases}}^{\text{Prob. Duale}}$$

Esempio

Creiamo il problema duale dell'esercizio della Schiopparelli & co. Dato il solito modello. Troviamo prima di tutto $\overline{x}, \overline{c}, \overline{b}, A$.

$$\begin{cases} \text{Max } Z = 3x_1 + 5x_2 \\ x_1 \leq 4 \\ 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

$$\overline{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}, \quad \overline{c} = \begin{bmatrix} 3 & 5 \end{bmatrix}, \quad \overline{b} = \begin{bmatrix} 4 & 12 & 18 \end{bmatrix} \quad A = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 3 & 2 \end{bmatrix}$$

$$\begin{cases} \text{Min}(W = y{\color{green}b}) \\ {\color{blue}A^T} y \geq {\color{red}c} \\ y \geq 0 \end{cases} \quad \to \begin{cases} \text{Min}(W = 4y_1 + 12y_2 + 18y_3) \\ y_1 + 3y_3 \geq 3 \\ 2y_2 + 2y_3 \geq 5 \\ y \geq 0 \end{cases}$$

Si può ora notare la relazione tra il problema primale e il problema duale.

- I parametri per un vincolo in uno dei due problemi sono i coefficenti di una variabile nell'altro problema
- I coefficenti della funzione obiettivo di uno dei due problemi sono i parametri delle funzioni di vincolo dell'altro problema.
- Si può anche notare che su un problema ha n parametri nella funzione obiettivo e m vincoli allora il problema duale avrà m parametri nella funzione obiettivo e n vincoli allora il problema duale

5.1.1 Relazione dei problemi primali-duali

Proprietà di dualità debole
Se $\overline{x}$ è accetabile per la soluzione per problema primale e $\overline{y}$ è accettabile per la soluzione di del problema duale allora $\overline{cx} \leq \overline{y}\overline{b}$. Quest'affermazione deve valore per ogni coppia di soluzione accetabili, il valore massimo di $Z = cx$ è uguale al valore minimo possibile per $W = yb$.

Dimostrazione

Dimostrazione:
Data una coppia di problema composti da uno *primale* e uno *duale* così formulati:
P: Max $Z = c^T x, Ax \leq b, x(\in X) \geq 0$
D: Min $W = b^T y, A^T y \geq c, y(\in X) \geq 0$

$$\begin{cases} A^T_{n\ m}\overline{y}_m \geq \overline{c}_n \\ \overline{x}_n \geq 0 \end{cases} \Rightarrow \underset{\overline{x}_{n\ 1}\cdot A_{n\ m}=\alpha_{1\ m}\alpha_{1m}\cdot\overline{y}_{1\ m}=k}{\overline{x}^T_n A^T_{n\ m}\overline{y}^T_m} \geq \underset{\text{Scalare}}{\overline{x}^T_n\overline{c}_n} \Leftrightarrow \underbrace{\overline{c}^T_n\overline{x}_n}_{Z} \leq \overline{x}^T_n A^T_{n\ m}\overline{y_m}^T$$

$$\begin{cases} A\overline{x} \leq \overline{b} \\ \overline{y} \geq 0 \end{cases} \Rightarrow \overline{y}^T_m A_{m\ n}\overline{x}_n \geq \overline{y}^T_m b_m \Leftrightarrow \underbrace{\overline{b}^T_m\overline{y}_m}_{W} \geq \overline{y}^T_m A_{m\ n}\overline{x}_n$$

In entrambi casi il secondo membro è uno scalare che porta a poter scrivere $\overline{x}^T A^T\overline{y} = (\overline{x}^T A^T\overline{y})^T$ visto che il trasposto di uno scalare è sé stesso. Ma per la proprietà $(BA)^T = B^T A^T$ posso scrivere $(\overline{x}^T A^T\overline{y})^T = \overline{y}^T(A^T)^T(\overline{x}^T)^T$. Tolgo le trasposte ridontanti e ottengo $(\overline{x}^T A^T\overline{y})^T = \overline{y}^T A\overline{x}$ arrivando allora a poter riassumere tutto così:

$$\overline{x}^T A^T\overline{y} = (\overline{x}^T A^T\overline{y})^T = \overline{y}^T(A^T)^T(\overline{x}^T)^T = \overline{y}^T A\overline{x}$$

Assemblando il tutto:

$$\underset{Z(\overline{x})}{c^T\overline{x}} \leq \overline{x}^T A^T\overline{y} = \overline{y}^T A\overline{x} \leq \underset{W(\overline{y})}{b^T\overline{y}}$$

A questo teorema seguono due corollari:

- Se $\overline{x} \in X, \overline{y} \in Y, z(\overline{x}) = w(\overline{y})$ allora $\overline{x} = x^*, \overline{y} = y^*, z(\overline{x}) = z^*, w(\overline{y}) = w^*$
- Se il problema prilame è *illimitato* allora il suo duale è *inammissibile*.

Lemma di Farkas:
Il lemma di Farkas dice che dato $A_{m\ n}\overline{x}_m = \overline{b}_m$ allora una sola delle seguenti è vera:

1. $\exists\overline{x} \in R^n | A\overline{x} = \overline{b}, \overline{x} \geq 0$
2. $\exists\overline{y} \in R^m | A^T\overline{y} \geq 0, \overline{b}^T\overline{y} < 0$

Questo significa che dato un sistema di equazioni o esiste una soluzione o non esiste.

Dimostrazione

1. 1) è vera, 2) è falsa.
   Riscrivo le premesse:

$$\begin{cases} A\overline{x} = \overline{b} \\ \exists\overline{x} \in R^n | A\overline{x} = \overline{b} \\ \overline{x} \geq \overline{0} \end{cases} \quad \text{Date le premesse:}$$

   - $A^T\overline{y} \geq 0 \Rightarrow \underset{\text{scalare}}{\overline{x}^T A^T y} > 0 \forall\overline{y} \in R^m$
   - $\overline{x}^T A^T = \overline{b}^T \Rightarrow \overline{b}^T\overline{y} \geq 0 \forall\overline{y} \in R^m$
   - $\nexists\overline{y} \in R^m | A^T\overline{y} \geq 0, b^T\overline{y} < 0$ perché nel punto precedente abbiamo dimostrato che $\overline{b}^T\overline{y} \geq 0 \forall\overline{y} \in R^m$.

2. 1) è false, 2) è vera.

Convessità

Uno spazio vettoriale C è convesso se per ogni coppia di punti x, y:
$\forall x, y \in C, \forall \lambda \in [0,1] \rightarrow \lambda x + (1-\lambda)y \in C$. In poche parole l'insieme è convesso se per ogni coppia di punti contiene tutti i segmenti li uniscono.

Teorema dell'iperpiano separatore

Siamo A e B due insiemi convessi in una spazio vettoriali R^n Allora $\exists H, \exists a$: $a \cdot x \leq c \forall x \in A, a \cdot y \geq c \forall y \in B$

Assumiamo che 1) sia falsa e definiamo il cono $C =$ $\overline{q} \in R^m | \exists X(\overline{q} \in R^n | Ax(q) = q, x(q) \geq 0)$.
C è convesso, essendo 1) falsa $b \notin C$.
Per il teorema del iperpiano separatore:

- $\exists \overline{y} \in R^m - 0 : \overline{q}^T \overline{y} \geq 0 \ \forall \overline{q} \in C, \overline{b}^T \overline{y} < 0$.
- Dato: $q = Ax(\overline{q}) \ \forall q \in C$ si ha: $\overline{q}^T \overline{y} \geq 0 \ \forall q \in C \Rightarrow x(q)^T A^T \overline{y} \geq 0 \forall q \in C$
- $x(q) \geq 0 \forall q \in C \rightarrow x(q)^T AT\overline{y} \geq 0 \ \forall q \in C \Rightarrow A^T \overline{y} \geq 0$
- Abbiamo dimostrato che $A^T \overline{y} \geq 0$.

Dimostrazione inutilmente complicata

Versione alternativa di Farkas

Dato un sistema di equazioni lineari $A_{m\ n} x \leq b_m, x \geq 0$, allora una sola è vera

- $\exists \overline{x} \in R^n : A\overline{x} \geq b, \overline{x} \geq 0$
- $\exists \overline{y} \in R^m : A^T_{n\ m} \overline{y} \geq 0, b^T \overline{y} < 0, \overline{y} \geq 0$

Introduciamo un vettore $\overline{s} \geq 0$. La prima condizione del **Lemma** diventa:

$$\exists \overline{x} \in R^n, \overline{s} \in R^m : A\overline{x} + Is = b, \overline{x} \geq 0, \overline{s} \geq 0$$

Definiamo $A' = [A|I]_{m,(n+m)}$ e $x' = \begin{bmatrix} \overline{x} \\ \overline{s} \end{bmatrix}_{1,(n+m)}$
Le due condizioni diventano:
$\exists x' \in R^{n+m} : A'x' = b, x' \geq 0$ che diventa per il lemma di Farkas alternativa a:
$\exists y \in R^m : A'^T y \geq 0, b^T y < 0, y \geq 0$.

Proprietà di dualità forte

Se x^* è una soluzione ottima per il problema ottimale e y^* allora $cx^* = y^*b$. Questo si può dedurre dal punto precendente. Perché tra tutte le soluzioni per il problema primale x^* è il massimo dell'insieme delle soluzioni accettabili, per y^* è il minore delle soluzioni accettabili per il problema duale. Avendo il vincolo della **proprietà di dualità debole** $cx \leq yb$ l'unico caso in cui le due soluzioni possano essere uguali è avere la soluzione ottima.

Dimostrazione

Sia $y^* \in R^m$ una soluzione ottima per il problema **duale** di valore: $w^* = b^T y^*$, il valore è uno scalare.
Lo scopo della dimostrazione è arrivare a dimostrare $\exists \overline{x}^* \in R^n : A\overline{x}^* \leq \bar{b}, \overline{x}^*$ che soddisfa $c^T \overline{x}^* \geq b^T \overline{y}^*$ (che sono scalari).
Prima parte

- Assumiamo che $\nexists \in R^n : Ax \leq b, x \geq 0, c^T \overline{x} \geq \overline{w}^*$
- Allora posso scrivere $\nexists \in R^n : Ax \leq b, -c^T x \leq -w^*, x \geq 0$
- $A' = \begin{bmatrix} A \\ -c^T \end{bmatrix}_{(m+1)\text{ x }n} \quad b' = \begin{bmatrix} b \\ -w^* \end{bmatrix}_{(m+1)\text{ x }1}$
- Equivale $\nexists x \in R^n : A'x \leq b', x \geq 0$ che implica $\exists y' \in R^{m+1} : A'^T y' \geq 0, b'^T y' < 0, y' \geq 0$.

Seconda parte

- $\exists y' \in R^{m+1} : A'^T y' \geq 0, b'^T y' < 0, y' \geq 0$ sia $y' = \begin{bmatrix} \overline{y} \in R^m \\ \lambda \in R \end{bmatrix}$
- Allora $A'^T \overline{y}' \geq 0, b'^T \overline{y}' < 0, \overline{y}' \geq 0 \Leftrightarrow A^T \overline{y} - c\delta \geq 0, b^T \overline{y} - w^*\delta < 0, \overline{y} \geq 0, \delta \geq 0$

$A^T \overline{y} - c\lambda \geq 0, b^T \overline{y} - w^*\lambda < 0, \overline{y} \geq 0, \lambda \geq 0$.
Caso I:$(\lambda > 0)$
$\hat{y} = \frac{\overline{y}}{\lambda}$ si ha $\begin{cases} A^T \hat{y} \geq c \\ b^T \hat{y} < w^* \\ \hat{y} \geq 0 \end{cases}$
Caso II:$(\lambda = 0)$
$A^T \overline{y} - c\lambda \geq 0, b^T \overline{y} - w^*\delta < 0, \overline{y} \geq 0 \rightarrow A^T \overline{y} \geq 0 b^T \overline{y} < 0, \overline{y} \geq 0$
Se $\hat{y} = y^* + \overline{y}$. Da cui ottiene $A^T \overline{y}^* \geq c, b^T \overline{y}^* = w^*, y^* \geq 0$
Da questo si ottiene $A^T \hat{y} \geq c, b^T \hat{y} < w^*, \hat{y} \geq 0$
Quindi possiamo dimostrare per assurdo: $\exists x^* \in R^n : Ax^* \leq b, x^* \geq 0, c^T x^* \geq w^* \rightarrow z^* = c^T x^* = b^T y^* = w^*$

Proprietà della soluzione complementare

Ad ogni iterazione dell'algoritmo del simplesso l'algoritmo identifica simultaneamente i CPF in x per il problema primale e la soluzione complementare di y per il problema duale tale che $cx = yb$. Da questo si può dedurre che se x non è ottima allora neanche y sarà ottima.

Proprietà della soluzione ottima complementare
All'iterazione finale, l'algoritmo del simplesso identifica simultaneamente la soluzione ottima per il problema primale(x^*) e per il problema duale(y^*) tale che $cx^* = y^*b$.

Teorema Fondamentale Dualità Lineare
Data una coppia di problemi.

$$P : \text{Max } z(x) \; x \in X$$
$$D : \text{Min } w(x) \; y \in Y$$

Allora una delle quattro è vera.

- Esiste la soluzione ottima per P e D
- Esiste la soluzione ottime per P e D è inammissibile
- Esiste la soluzione ottime per D e P è inammissibile
- P,D sono inammissibili.

Teorema dello scarto complementare
Dato una coppia primale-duale

$$P : \text{Max } z = c^T x, Ax \geq b, x \geq 0$$
$$D : \text{Min } w = b^T y, A^T y \geq c, y \geq 0$$

La condizione per l'ottimalità è:

$$\overline{y}^T (b - A\overline{x}) = 0$$
$$(A^T \overline{y} - \overline{c})^T \overline{x} = 0$$

DA QUI LA QUALITA' CALA
Teorema dello scarto complementare
Data la coppia:

$$P : max(z) = c^T x : Ax \leq b, x \geq 0$$
$$D : mix(w) = b^T y : A^T y \geq c, y \geq 0$$

Allora date le soluzioni ottime $\overline{x}^*$ $\overline{y}^*$ sono valide le seguenti equazioni:

$$\overline{y}^T (b - A\overline{x}) = 0$$
$$(A^T \overline{y} - c)^T \overline{x} = 0$$

Dimostrazione

Sappiamo che dati due problemi duali P e D che hanno entrambi soluzioni ottime vale l'equazione: $z^* = c^T \overline{x}^* = \overline{b}^T y^* = \overline{w}^*$
Sappiamo anche che nella soluzione: $A\overline{x} = b$ e $A^T \overline{y} = c$, da questi si può ricavare:
$A\overline{x} - b = 0 \rightarrow b - A\overline{x}$=0 & $A^T \overline{y} - c = 0$
Sostituiamo nelle premesse e ci viene: $\overline{y}^T (0) = 0$ e $(0)^T \overline{x} = 0$. Quindi il teorema è dimostrato.

5.2 Algoritmo simplesso duale

Dato i coefficenti di P e D sono gli stessi cioé A e A^T quindi si possono rappresentare con lo stesso tablueat.
Problema Primale: Conserva l'ottimalità e persegue l'ottimalità.
Problema Duale: Conserva l'ottimalità e persegue l'ammissibilità.

- La riga di pivot viene nella prima colonna
- Il pivot deve essere non negativa
- La colonna di Pivot viene scelta minimizzando il valore assoluto del rapporto tra il coefficente di costo ridotto ed il candidato pivot.

6 Sezione post-ottima:

Controllo della robustezza del progetto. Esso viene fatto tramite l'analisi post'ottima.
Input: $A, \overline{b}, \overline{c}$
Output: $B^*, \overline{x}^*, \overline{z}^*$.
L'Analisi post-ottima ha lo scopo di capir quanto può variare ogni coefficente(dei vettori $\overline{c}$ e $\overline{b}$) c_j, b_i senza che cambi la base ottima B^*.
La Base B^* rimane ottima finché:

- Ammissibilità: $x_B = B^{-1}b \geq 0$
- Ottimalità: $c_N = c_N - c_B B^{-1} N \geq 0$

Come effettuare l'analisi post'ottima va eseguita nella forma:

- Funzione obiettivo massimizzato.
- Vincoli in forma di disuguaglianza $\geq$

Consideriamo una colonna $\overline{j}$ con due casi:

1. ($\overline{j} \in B$ e $\overline{r}$):
   $\max\{-\infty, \max_{j \in N^+}\{\frac{-c_j^*}{a_{\overline{r}j}^{*+}}\}\} \leq \Delta c_{\overline{j}} \leq \min\{\min_{j \in N^-}\{\frac{-c_j^*}{a_{\overline{r}j}^{*-}}\}, +\infty\}$
2. ($\overline{j} \in N$): $\Delta c_{\overline{j}} \leq c_{\overline{j}^*}$