

# Programmazione Lineare

## Metodo del Simplexso

MAX  $cx$  = MIN  $-cx$

Problema di MIN

$b$  = colonna termini noti

Variabile NB = Variabile non **in** base

Variabile B = Variabile **in** base

```
PickEP {  
    CP=c <<; c<0  
    RP=b_i / c_i <<; c_i > 0  
    # Se tutti i rapporti b_i/c_i sono negativi,  
    # il problema è illimitato  
    ElementoPivot = (RP,CP)  
}
```

```
prepareSimplex {  
    # I termini noti devono essere >= 0  
    # Tutti i vincoli del tipo <= (Min) o >= (Max)  
    # La matrice aggiunta delle variabili di Slack  
    # deve essere una matrice sidentità  
    # Sopra la matrice identità, i coefficienti  
    # devono essere 0 (due fasi)  
    # Se il problema è di Massimo,  
    # invertire i coefficienti sulla riga 0
```

```
}
```

```
prepareSimplex()  
while(!(coeff su Riga0 tutti  $\geq 0$ )) {  
    PickEP()  
    # La variabile sulla colonna CP va in base,  
    # sostituendo quella precedente sulla RP.  
    RP = RP / EP  
    for each (riga i con  $(i, CP) > 0$ ):  
         $R_i = R_i + |(i, CP)| * RP$   
    foreach (riga i con  $(i, CP) < 0$ ):  
         $R_i = R_i - |(i, CP)| * RP$   
    foreach (riga i con  $(i, CP) = 0$ ):  
         $R_i = R_i$   
    }1  
BFS = (Variabili NB = 0, i-esima variabile B =  $b_i$ )
```

## Problemi di PL

**Var non di base** var uguale a 0

**Var di base** var di qualsiasi valore le cui colonne formano una matrice identità

**Sol aumentata** tutte le variabili

**Sol di base** sol aumentata che sta su un vertice

**Sol di base degenere** sol di base con qualche var di base = 0

**Sol non di base** sol aumentata che non sta su un vertice (senza variabili di slack)

**Una soluzione NON di base di un problema di PL può appartenere alla regione ammissibile?**

**Vero**, la regione ammissibile è semplicemente definita dai vincoli. Una soluzione di base soddisfa tutti i vincoli, ma non è su uno dei vertici.

**Quante variabili possono assumere valore diverso da zero in una soluzione di base?**

Sia  $n$  il numero di vincoli funzionali del problema di PL,  $n$  variabili sono di base e dunque potranno assumere valori non nulli.

**In una soluzione di base ammissibile, tutte le variabili nulle sono necessariamente fuori base?**

**Falso**, le variabili di base possono assumere valore nullo. In particolare, se qualche variabile di base vale 0 in una soluzione di base ammissibile, essa si dice *degenere*.

**Dare la definizione di soluzione di base**

**Soluzione ammissibile** è una soluzione che soddisfa il sistema di equazioni della forma aumentata e che ha tutte le variabili non negative.

**Soluzione di base** È un vertice a cui sono stati aggiunti i corrispondenti valori delle variabili di slack. Godrà delle seguenti proprietà:

- Ogni variabile è una variabile di base o non di base;

- Il numero delle variabili di base è uguale al numero di vincoli funzionali.  
Numero di variabili non di base = numero delle variabili - numero vincoli funzionali;
- Le variabili non di base sono poste a 0;
- I valori delle variabili di base sono le soluzioni del sistema di equazioni.
- Se le variabili di base soddisfano i vincoli di non negatività, la soluzione di base è una BFS.

**Soluzione aumentata** È una soluzione per la quale alle variabili originali sono aggiunte le variabili di slack

**Soluzione di base ammissibile (BFS)** è un vertice ammissibile cui sono stati aggiunti i corrispondenti valori delle variabili di slack

Una soluzione di base con le variabili di base che soddisfano i vincoli di non negatività è una BFS.

**Dire se la soluzione ottimale di un problema di PL può essere non di base, giustificando la risposta**

**Vero**, potrebbero esserci più di una BFS collegate da un segmento che danno vita dunque a infinite combinazioni convesse dei due vertici.

## Sensitività

**Quale è il tasso di variazione della funzione obiettivo al variare dei termini noti?**

Il tasso di variazione coincide con i prezzi ombra.

Il prezzo ombra  $i$  può variare di una quantità  $\delta$  definita dalla seguente disequazione, mantenendo la soluzione attuale ottima.

Sia  $CS_i$  la colonna della variabile di slack  $i$  (esclusa riga 0) e  $b$  la colonna di termini noti:

$$b + \delta CS_i \geq 0$$

**Quali sono gli intervalli di variazione di ciascuna risorsa affinché la soluzione ottima rimanga ammissibile?**

$$b + \delta CS_i \geq 0$$

**Intervalli di variazione del costo dell'attività  $x_i$  affinché la soluzione ottima rimanga tale:**

Sia  $R$  la riga che sulla colonna di  $x_i$  ha coefficiente 1:

Per tutti i coeff  $\neq 0, 1$  su  $R$ :

$$coeff_i(R_0) + \delta coeff_i(R) \geq 0$$

**Data una soluzione ottima corrente, determinare se è opportuno introdurre una nuova attività  $x_6$ . Se sì, calcolare la nuova soluzione ottima.**

Sia  $(x)$  la soluzione ottima corrente, controllare che  $(x, 0)$  sia ancora valida e ottima:

- Validità: sostituire nei vincoli
- Ottimalità: il costo ridotto (coefficiente sulla riga 0) di  $x_6$  deve risultare non negativo

Sia

- $c_b B^{-1}$  vettore riga 0 variabili ombra
- $A_i$  vettore colonna tecnologico nuova attività  $i$
- $c_i$  costo unitario nuova attività  $i$

Costo ridotto  $x_6 = c_b B^{-1} A_6 - c_6$

Se il costo ridotto è  $\geq 0$ , significa che la soluzione  $(x, 0)$ , dove **non** si svolge  $x_6$  ( $x_6 = 0$ ) è ancora ottima, dunque non è conveniente aggiungerla.

Se il costo ridotto risultasse  $< 0$ , non svolgere l'attività non sarebbe più una soluzione ottima, quindi è conveniente aggiungerla.

Per calcolare la nuova soluzione, si ripete il simplesso aggiornando il tableau in questo modo:

- Sulla riga 0, il coefficiente della nuova attività è il costo ridotto calcolato prima
- La colonna della nuova variabile si calcola moltiplicando la matrice delle variabili di slack (escludendo sempre la riga 0) per il vettore colonna tecnologico della nuova attività:  $B^{-1} * A_6$

**(???) Calcolare il vettore colonna relativo all'introduzione di una nuova attività  $x_c$  con costo ridotto 3 e vettore variabili tecnologiche  $A = (3, 2, 1)$**

Vettore Variabili Ombra = Vettore coefficienti variabili

CR = Costo Ridotto

A = Vettore variabili tecnologiche

VVO \* A - CR = Vettore Colonna nuova attività  $x_c$

# Dualità

**Impostare le relazioni di complementarità che le soluzioni ottime del primale e del duale devono soddisfare**

Sia:

- $x_i$  le variabili del primale;
- $u_j$  le variabili del duale;
- $b_j$  il j-esimo termine noto del primale;
- $Ax_j$  i termini del j-esimo vincolo funzionale del primale;
- $c_i$  l'i-esimo termine noto del duale;
- $yA_i$  i termini dell'i-esimo vincolo funzionale del duale.

$$u_j(b_j - Ax_j) = 0$$

$$(yA_i - c_i)x_i = 0$$

## Teorema di dualità debole

Data una coppia Primale-Duale  $\min c^T x : Ax \geq d, x \geq 0$ ,  $\max u^T d : u^T A \leq c^T, u \geq 0$ :

Sia  $X$  la regione d'ammissibilità di  $P$  ed  $U$  quella di  $D$ , per ogni  $x \in X$ ,  $u \in U$  risulta:  $c^T x \geq u^T d$

## Teorema di dualità forte

Se il primale ha soluzione ottima finita:

- 1) anche il suo duale ha soluzione ottima finita;
- 2) i valori della due soluzioni sono uguali.

## Dare un esempio di utilizzo del teorema di dualità

Le proprietà del problema duale definite dal teorema di dualità ci interessano perchè:

- Il problema duale può essere più facile da risolvere (meno vincoli, conviene)
- Il problema duale corrisponde ad una diversa visione dello stesso problema
- Molti algoritmi utilizzano aspetti della dualità, quali il Simplex Duale, Primale/Duale, alternativi al Simplex utili per certe classi di problemi
- Il problema duale fornisce bounds utili per risolvere problemi a variabili intere (Branch and Bound)
- Condizioni di ottimalità

**Dato il grafo delle precedenze tra le attività di un progetto e la loro durata media, come si determinano le loro attività critiche?**

$t_{\min}$  di un'attività è il tempo minimo entro cui si possono terminare tutte le fasi necessarie per iniziirla. Le prime hanno  $t_{\min} = 0$ , per ogni altra attività sequente  $A$ ,  $t_{\min}(A)$  è la massima somma tra  $t_{\min}(P) + \text{durata}(P)$  dei predecessori, con  $P$  predecessore con tale massimo valore.

$t_{\max}$  di un'attività è il massimo tempo entro cui devo iniziare l'attività stessa, pena un aumento del tempo minimo per completare il progetto. L'ultima attività ha  $t_{\min} = t_{\max}$ , per ogni altra  $t_{\max}$  è la differenza minima tra la durata di  $P$  e  $t_{\max}$  di un successore  $A$ .

Un'**attività critica** è un'attività che ha  $\text{slack} = 0$ .  $\text{Slack} = t_{\max} - t_{\min}$ . Un percorso critico è la sequenza più lunga di attività critiche.



## Perchè sono dette critiche?

Un'attività è detta *critica* è un'attività il cui inizio non può essere ritardato nemmeno di un'unità di tempo, altrimenti l'intera durata minima del progetto in esame verrebbe aumentata.

## Programmazione non lineare

### Elencare le condizioni di ottimalità di KKT

1. Ad ogni vincolo, associare una  $\delta_i$ .
2. Vincoli di *ammissibilità duale* relativi ai  $\delta_i$  (inverto segni, = variabile senza vincoli)
3. Condizioni di *complementarietà*  
 $\delta_i$  (i-esimo vincolo) = 0
4. Condizioni sul *gradiente*  
 $(dFO/dx, dFO/dy) - \delta_i (dVi/dx, dVi/dy) = 0$  (per tutte le  $i$ )
5. Definire  $2^n$  sviluppi, combinando le condizioni

### Qual è la differenza tra clustering e classificazione? (Descrivere la differenza tra classificazione supervisionata e non supervisionata)

La classificazione è un tipo di apprendimento supervisionato che descrive il problema di identificare a quale di un insieme di categorie appartiene una

nuova osservazione, basandosi su di un training set che contiene osservazioni già correttamente identificate.

La classificazione può essere implementata con l'algoritmo K-Nearest Neighbor o SVM.

Il clustering (classificazione non supervisionata) è una procedura di apprendimento non supervisionato che consiste nel raggruppare un insieme di oggetti in un modo tale che oggetti dello stesso gruppo (chiamato cluster) siano più *simili* tra loro rispetto che a quelli di altri gruppi.

Il clustering può essere implementato con l'algoritmo K-means.

## Differenze

In particolare, il clustering è una tecnica di apprendimento non supervisionata, mentre la classificazione è supervisionata: nella prima non è presente un training set, nella seconda sì. Inoltre, nel clustering vengono usati concetti statistici gli oggetti vengono divisi in cluster con caratteristiche simili. La classificazione usa invece algoritmi che tengono conto del training set. Infine, nel clustering non si fa ricorso ad etichette, mentre nella classificazione sì, per alcuni punti.

## **Descrivere il metodo del K-Means per il problema di clustering illustrandone vantaggi e svantaggi**

K-Means si prepone di minimizzare la varianza totale intra-cluster. Ogni cluster viene identificato tramite un punto medio o centroide. Vengono create K partizioni e ad ognuna assegnati punti di ingresso (casualmente o usando informazioni euristiche). Viene calcolato il centroide di ogni gruppo e costruita una nuova partizione associando ogni punto d'ingresso al cluster

il cui centroide è più vicino. Vengono ricalcolati i centroidi e ripetuta iterativamente la procedura, finchè l'algoritmo non converge.

Pro e Contro:

- È facilmente implementabile/utilizzabile
- Converge molto velocemente e generalmente il numero di iterazioni è minore del numero di punti. Tuttavia può essere molto lento del caso peggiore (superpolinomiale/esponenziale)
- Non garantisce il raggiungimento dell'ottimo globale, la qualità della soluzione finale dipende dal set di cluster iniziale e può ottenere una soluzione ben peggiore dell'ottimo globale. Tuttavia, essendo molto veloce, è possibile applicarlo più volte e scegliere tra le soluzioni quella più soddisfacente.
- Svantaggio: richiede di scegliere il numero di cluster da trovare. Se i dati non sono naturalmente partizionati si ottengono risultati non desiderabili.
- L'algoritmo funziona bene solo quando sono individuabili cluster sferici di dati.

Formalmente:

```
1: KM(n observations, k) → k clusters
2:   cs ← Initialize(k)
3:   clusters ← {}
4:   clusters0 ← Partition(observations, cs)
5:   if clusters = clusters0 then
6:     return clusters
7:   else
8:     clusters ← clusters0
9:     cs ← RecomputeCentroids(clusters)
10:    goto 4
```

```
11:     end if
```

## **Illustrare e formalizzare il problema di clustering (classificazione non supervisionata) come problema di ottimizzazione**

Gli algoritmi di clustering partizionale creatno una partizione delle osser-  
vazioni minimizzando una certa funzzone di costo:

$$\sum_{j=1}^k E(C_j)$$

Dove

- $k$  è il numero dei cluster;
- $C_j$  è il  $j$ -esimo cluster
- $E : C \rightarrow \mathbb{R}^+$  è la funzione di costo associata al singolo cluster.

## **Descrivere brevemente il metodo SVM per la classificazione su- pervisionata**

TODO