PROGETTO RICERCA OPERATIVA

Un'applicazione del modello di set covering

Di Lapi Sergio, mat. : 747808
Atzori Guendalina, mat. : 745416

Consegna del progetto asseganto

L'amministrazione comunale deve decidere l'apertura delle stazioni dei vigili del fuoco tra 6 possibili siti in modo da garantire che ognuna delle 5 zone in cui è suddiviso il comune sia raggiungibile entro 8 minuti.

Nella seguente tabella sono contrassegnate con "*" le zone raggiungibili entro 8 minuti da ciascuna stazione e sono inclusi i costi di apertura per ogni stazione.

Zone/Stazioni	1	2	3	4	5	6
A		*	*	*	*	
B			*	*		
C	*	*				*
D					*	*
E	*	*	*			
Costi di apertura	4	6	10	14	5	6

Si decida quali centri di emergenza aprire con costo totale minimo in modo da garantire che ogni zona sia servita da almeno un centro di emergenza.

Intuizione grafica

Per poter rappresentare e visualizzare graficamente il nostro problema supponiamo che siano vere le seguenti condizioni:
Fig_A)

- da ogni generica stazione dei viglili (quad. rosso) può partire un'auto (cerchio giallo) a velocità costante x, senza accelerazione e può compiere un tagitto lineare (non può sterzare)
- non esistono le strade, ciò vuol dire che ogni auto può percorre il tragitto in qualsiasi direzione (o angolo)
- tutte le auto dei vigili del fuoco hanno le stesse caratteristiche (e velocità)

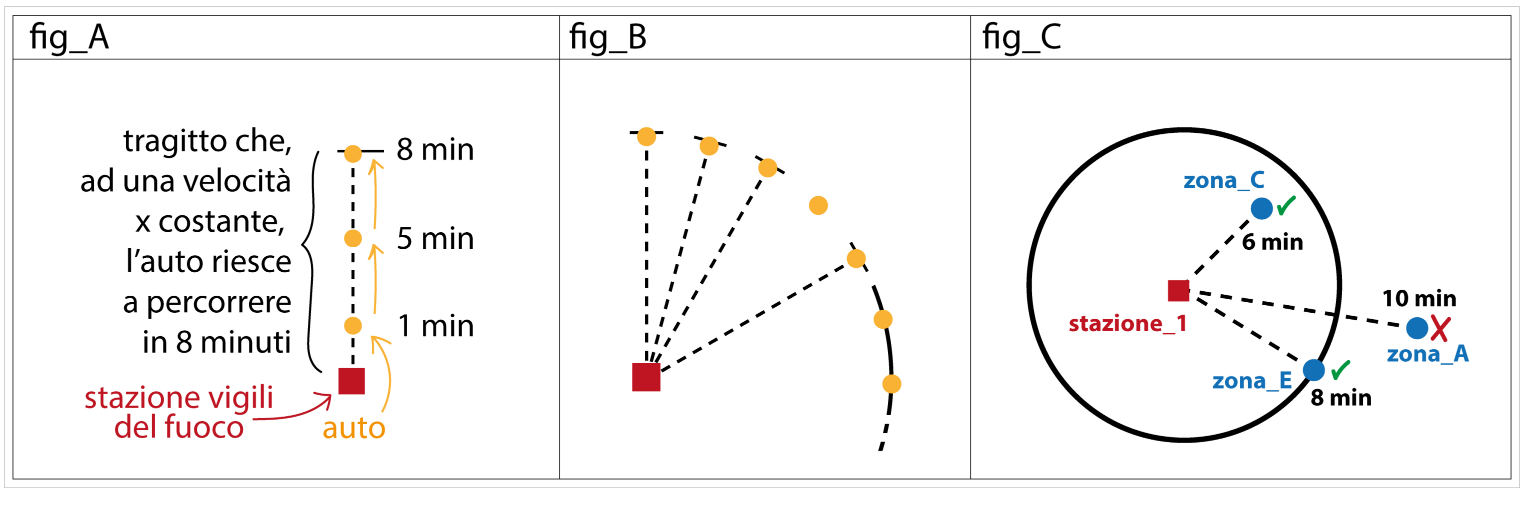
Da tali condizioni ne deriva:

Fig_B)

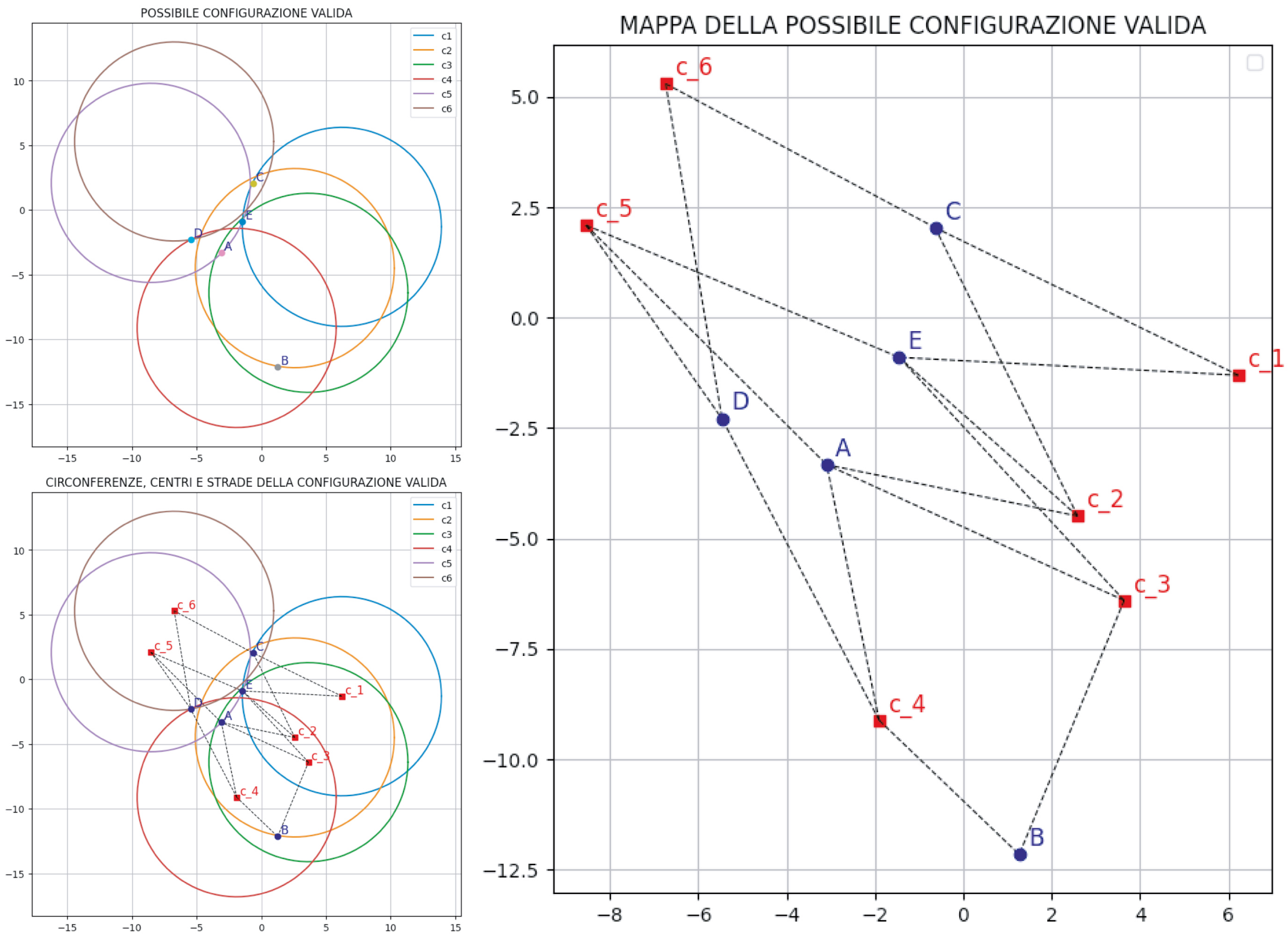
- l'insieme dei "punti limiti" (ovvero il punto l'auto che percorre il tragitto i si trova al minuto 8:00) descrive una circonferenza

Fig_C)

- la zona da servire (cerchio blu) viene servito entro 8 minuti s.s.s. si trova all'interno della circonferenza (circonferenza inclusa)



Ponendo a sistema quanto premesso sopra, ricaviamo un grafico coerente (di una possibile delle infinite soluzioni) della disposizione di zone da servire e la posizione di eventuale apertura delle stazioni dei vigili del fuoco.



N.B.:
Il grafico sopra (mappa della possibile configurazione valida) non ci fornisce nessuna soluzione!!!

Ci è tuttavia utile per visualizzare meglio il problema.

Il modello - set covering

Indici:

$$
\begin{aligned}
i &= \mathrm{A, B, C, D, E} && \text{(zone)} \\
j &= 1, 2, 3, 4, 5, 6 && \text{(stazioni)}
\end{aligned}
$$

Dati:

$$\text{costo}_j = [\,4,\ 6,\ 10,\ 14,\ 5,\ 6\,]$$

$$\text{servizio}_{ij} = \begin{bmatrix} & * & * & * & * & \\ & & * & * & & \\ * & * & & & & * \\ & & & & * & * \\ * & * & * & & & \end{bmatrix} \xrightarrow{\text{trasformazione}} \text{servizio}_{ij} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Il vettore *costo_j* rappresenta il costo di apertura (che può dipendere da vari fattori, vedi sotto "variazioni sui dati") della stazione *j* espresso in milioni di euro.
Vale a dire 4 mln per aprire la stazione 1, 6 mln per la stazione 2 ...

La matrice *servizio_ij* invece rappresenta la possibilità della stazione *j* di servire la zona *i* o meno.
Dato che gli asterischi non sono direttamente modellabili matematicamente trasformiamo la matrice in una matrice di 0 e 1:

- 1 se la stazione *j* riuscirebbe a servire la zona *i*
- 0 altrimenti

n.b.: la possibilità eventuale di servire il centro *i* qualora la stazione *j* fosse aperta!

Variabili decisionali:

$$y_j \in \{0,1\} \quad \text{dove} \quad \begin{cases} 1 & \text{se aperto} \\ 0 & \text{altrimenti} \end{cases}$$

La decisione da prendere è:

- 1, aprire il centro
- 0, altrimenti

Ci è necessaria perciò una variabile binaria, che può assumere valori tra 0 e 1, ossia può assumere valori in uno specifico sottoinsieme di Z+.
Un problema di set-covering è pertanto uno particolare problema ILP.

Funzione obiettivo:

$$\min z = 4y_1 + 6y_2 + 10y_3 + 14y_4 + 5y_5 + 6y_5$$

La funzione obiettivo è molto semplice, pretendiamo banalmente che il costo sia minore possiblie.

Vincoli, l'idea

$$\text{e.g. per la stazione 1...} \quad \rightarrow \quad 0y_1 + 1y_2 + \ldots + 1y_5 + 0y_6 \leq 1$$

Questo vincolo ci garantisce per che la zona 1 sia coperta al peggio da un solo centro.
n.b.: come nel caso in cui la stazione *j* non sarebbe comunque in grado di garantire gli 8 miuti al centro 1, *(0 * y_j)* il valore sarebbe 0 sia se *y_j* assumesse valore 1 che se assumesse valore 0.
E così via per ogni zona *i* ...

Modello matematico in forma compatta:

$$\begin{aligned} \min \quad & z = \sum_{j=1}^{6} \text{costo}_j \cdot y_j \\ \text{s.t.} \quad & \sum_{j=1}^{6} \text{servizio}_{ij} \cdot y_j \geq 1 \qquad \forall \quad i \\ & y_j \in \{0,1\} \end{aligned}$$

Modello AMPL

Adesso mettiamo in codice, quanto formalizzato sopra.

Per poter risolvere il modello in forma compatta, creeremo 2 file:

- file.mod --> che conterrà il modello
- file.dat --> che conterrà i dati

```
# === progetto_35.mod ===========================================================================================

# --- indici ----------------------------------------------------------------------------------------------------
set zone := A B C D E;
set stazioni := 1 2 3 4 5 6;

# --- costo_j ---------------------------------------------------------------------------------------------------
param costo :=
1 4
2 6
3 10
4 14
5 5
6 6;

# --- servizio_ij -----------------------------------------------------------------------------------------------
param servizio: 1 2 3 4 5 6   :=

A               0 1 1 1 1 0
B               0 0 1 1 0 0
C               1 1 0 0 0 1
D               0 0 0 0 1 1
E               1 1 1 0 0 0;

# --------------------------------------------------------------------------------------------------------------
```

```
# === progetto_35.dat ===========================================================================================

# --- indici ----------------------------------------------------------------------------------------------------
set zone;
set stazioni;

# --- costo_j ---------------------------------------------------------------------------------------------------
param costo{stazioni};

# --- servizio_ij -----------------------------------------------------------------------------------------------
param servizio{zone, stazioni};

# --- variabile binaria y_j -------------------------------------------------------------------------------------
var y{j in stazioni} binary;

# --- funzione obiettivo ----------------------------------------------------------------------------------------
minimize CostoTotale:
    sum {j in stazioni} costo[j] * y[j];

# --- vincoli ---------------------------------------------------------------------------------------------------
subject to CoperturaZone {i in zone}:
    sum {j in stazioni} servizio[i,j] * y[j] >= 1;

# --------------------------------------------------------------------------------------------------------------
```

Soluzione

Risolvendo il modello troviamo:

- Costo minimo: 16 milioni
- Stazioni aperte: stazione_3, stazione_6

Variazione sui dati

Premessa

Adesso, come anticipato prima, supponiamo che il costo dei di apertura delle stazioni dipenda da prezzo dei materiali, che a loro volta dipendono da altri fattori...

Per costruire le stazioni occorrono tre tipologie di materiali.
Inoltre supponiamo di riuscire a prevedere esattamente il prezzo dei materiali nel futuro perchè siamo in possesso di informazioni sensibili.
Nello sspecifico sappiamo che:

- materiale_1 --> a causa della reperibilità legata alle stagioni, ha un prezzo oscillante annualmente
- materiale_2 --> a causa della sua dipendenza dal costo dell'energia, ha un prezzo che è schizzato in un certo periodo a causa di una crisi del petrolio per poi ritornare alla normallità
- materiale_3 --> a causa della scarsità, essendo una risorsa che sta per terminare, ha un prezzo che cresce linearmente.

Ipotizziamo che si riunisca un consiglio per decidere all'unanimità se costruire o meno le stazioni.
Se tutti i componenti del consiglio votassero 'SI' alla costruzione allora è chiaro che si opterebbe per minimizzare i costi di costruzione a parità di servizio (nessuno vuole pagare più del minimo per il medesimo servizio).
Se non si raggiunge l'unanimità, allora il consiglio si riunirà nuovamente dopo 6 mesi e si siederà allo stesso tavolo per ridiscutere la costruzione o meno delle stazioni.

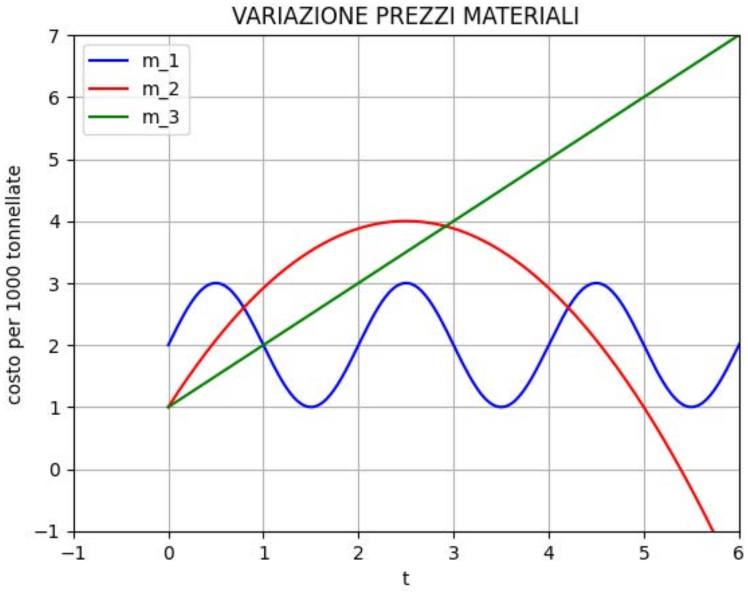
Nel frattempo però i prezzi dei materiali cambiano...

Dati sui prezzi e dipendeza della matrice costo

Siano le variazioni dei prezzi per ogni 1000 tonnellate di materiale in milioni di euro approssimati dalle seguenti funzioni:

$$\begin{cases} \text{p.p.1000t. } m_1(t) = \sin(\pi t) + 2 \\ \text{p.p.1000t. } m_2(t) = -\dfrac{12}{25}x^2 + \dfrac{12}{5}x + 1 \\ \text{p.p.1000t. } m_3(t) = x + 1 \end{cases}$$

n.b.: da qui in poi semplificheremo la notazione in *m_1* , *m_2* , *m_3* , ma esprimeranno sempre lo stesso dato.



Dopo aver osservato come variano i prezzi dei materiali, introduciamo la funzione di dipendenza del vettore *costo_j* dalle funzioni dei materiali

$$\overrightarrow{costo}_j(t)^\top = \overrightarrow{costo}_j(m_1(t), m_2(t), m_3(t))^\top = \begin{bmatrix} 1 \cdot m_1(t) + 1 \cdot m_2(t) + 1 \cdot m_3(t) \\ 2 \cdot m_1(t) + 1 \cdot m_2(t) + 1 \cdot m_3(t) \\ 3 \cdot m_1(t) + 3 \cdot m_2(t) + 1 \cdot m_3(t) \\ 2 \cdot m_1(t) + 8 \cdot m_2(t) + 2 \cdot m_3(t) \\ 1 \cdot m_1(t) + 3 \cdot m_2(t) + 0 \cdot m_3(t) \\ 2 \cdot m_1(t) + 1 \cdot m_2(t) + 1 \cdot m_3(t) \end{bmatrix}$$

Secondo questa supposizione, il problema che abbiamo risolto prima è stato risolto in *t = 0*, anche se non ne eravamo consapevoli.

Essendo:

$$\begin{cases} m_1(0) = \sin(\pi \cdot 0) + 2 = 2 \\ m_2(0) = -\frac{12}{25} \cdot 0^2 + \frac{12}{5} \cdot 0 + 1 = 1 \\ m_3(0) = 0 + 1 = 1 \end{cases}$$

... è banale verificare che:

$$\overrightarrow{costo}_j(0)^\top = \overrightarrow{costo}_j(m_1(0), m_2(0), m_3(0))^\top = \begin{bmatrix} 1 \cdot m_1(0) + 1 \cdot m_2(0) + 1 \cdot m_3(0) \\ 2 \cdot m_1(0) + 1 \cdot m_2(0) + 1 \cdot m_3(0) \\ 3 \cdot m_1(0) + 3 \cdot m_2(0) + 1 \cdot m_3(0) \\ 2 \cdot m_1(0) + 8 \cdot m_2(0) + 2 \cdot m_3(0) \\ 1 \cdot m_1(0) + 3 \cdot m_2(0) + 0 \cdot m_3(0) \\ 2 \cdot m_1(0) + 1 \cdot m_2(0) + 1 \cdot m_3(0) \end{bmatrix} = \begin{bmatrix} 1 \cdot 2 + 1 \cdot 1 + 1 \cdot 1 \\ 2 \cdot 2 + 1 \cdot 1 + 1 \cdot 1 \\ 3 \cdot 2 + 3 \cdot 1 + 1 \cdot 1 \\ 2 \cdot 2 + 8 \cdot 1 + 2 \cdot 1 \\ 1 \cdot 2 + 3 \cdot 1 + 0 \cdot 1 \\ 2 \cdot 2 + 1 \cdot 1 + 1 \cdot 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 6 \\ 10 \\ 14 \\ 5 \\ 6 \end{bmatrix}$$

e così via...

Come potrebbe essere solito, è difficoltoso arrivare all'unanimità. Pertanto risolviamo il problema per ogni *t/2* (sei mesi, ossia per ogni possibile cosiglio di amministrazione nel caso in cui non si fosse arrivati ad una decisione precedentemente) per *0 < t < 5* (5 anni).

Con la stessa procedura utilizzata per calocalre il vettore *costo_j in t(0)* ricaviamo:

$$\begin{aligned}
\text{costo}_j(0.0) &= [\,4.00,\ 6.00,\ 10.00,\ 14.00,\ 5.00,\ 6.00\,] \\
\text{costo}_j(0.5) &= [\,6.58,\ 9.58,\ 16.64,\ 25.64,\ 9.24,\ 4.50\,] \\
\text{costo}_j(1.0) &= [\,6.92,\ 8.92,\ 16.76,\ 31.36,\ 10.76,\ 4.00\,] \\
\text{costo}_j(1.5) &= [\,7.02,\ 8.02,\ 16.06,\ 35.16,\ 11.56,\ 3.50\,] \\
\text{costo}_j(2.0) &= [\,8.88,\ 10.88,\ 20.64,\ 41.04,\ 13.64,\ 5.00\,] \\
\text{costo}_j(2.5) &= [\,10.5,\ 13.50,\ 24.50,\ 45.00,\ 15.00,\ 6.50\,] \\
\text{costo}_j(3.0) &= [\,9.88,\ 11.88,\ 21.64,\ 43.04,\ 13.64,\ 6.00\,] \\
\text{costo}_j(3.5) &= [\,9.02,\ 10.02,\ 18.06,\ 39.16,\ 11.56,\ 5.50\,] \\
\text{costo}_j(4.0) &= [\,9.92,\ 11.92,\ 19.76,\ 37.36,\ 10.76,\ 7.00\,] \\
\text{costo}_j(4.5) &= [\,10.58,\ 13.58,\ 20.74,\ 33.64,\ 9.24,\ 8.50\,] \\
\text{costo}_j(5.0) &= [\,9.00,\ 11.00,\ 15.00,\ 24.00,\ 5.00,\ 8.00\,]
\end{aligned}$$

Soluzioni

Essendo particolarmente noioso risolvere manualmente 11 modelli diversi 'manualmente', aggrediremo i problemi nel seguente modo:

- utilizziamo lo stesso file.mod del problema originale per poi variare solo il file.dat con il vettore costo corrispondente alla variazione di pertinenza.
- Scriviamo uno script (launcher.run) che resetta il modello dopo averlo risolto, per poi ricaricare l'unico file.mod + file.dat della variazione successiva
- Raccogliamo in un file.txt i dati raccolti

Sotto un'esempio della logica utilizzata per scrivere launcher.run:

```
# === file.run ==========================================================================================
# ======================================================================================================


# === problema originale ================================================================================

# --- carica i file----------------------------------------------------------------------------------------
model 'file_path/file.mod';
data 'file_path/file.dat';

# --- risolve ---------------------------------------------------------------------------------------------
solve;
display y;
```

```
# --- scrive sul file.txt ---------------------------------------------------------------------------------------------
printf 'Problema originale: \nStazioni aperte:  ' > 'file_path/file.txt';
for {j in stazioni: y[j] = 1} {
    printf '%s  ', j >> 'file_path/file.txt';
}
printf '  ==>  Costo totale = %f\n\n', CostoTotale >> 'file_path/file.txt';

# --- reset -----------------------------------------------------------------------------------------------------------
reset;



# === variazione 1 ====================================================================================================

# --- stessa logica di prima ... --------------------------------------------------------------------------------------
model 'file_path/file.mod';
data 'file_path/file_variazione_0.dat';

solve;
display y;

printf 'Problema originale: \nStazioni aperte:  ' > 'file_path/file.txt';
for {j in stazioni: y[j] = 1} {
    printf '%s  ', j >> 'file_path/file.txt';
}
printf '  ==>  Costo totale = %f\n\n', CostoTotale >> 'file_path/file.txt';

reset;


# ---------------------------------------------------------------------------------------------------- e così via ... -
```

Il file 'risultati.txt' ha raccolto tutti i risultati dei problemi derivati dalla varazione di *costo_j* .

```
# === risultati.txt ===================================================================================================
# ====================================================================================================================

Problema originale:
Stazioni aperte:  3  6    ==>  Costo totale = 16.000000

Variazione 0:
Stazioni aperte:  3  6    ==>  Costo totale = 26.320000

Variazione 1:
Stazioni aperte:  3  6    ==>  Costo totale = 25.680000

Variazione 2:
Stazioni aperte:  3  6    ==>  Costo totale = 24.080000

Variazione 3:
Stazioni aperte:  3  6    ==>  Costo totale = 31.520000

Variazione 4:
Stazioni aperte:  3  6    ==>  Costo totale = 38.000000

Variazione 5:
Stazioni aperte:  3  6    ==>  Costo totale = 33.520000

Variazione 6:
Stazioni aperte:  3  6    ==>  Costo totale = 28.080000

Variazione 7:
Stazioni aperte:  3  6    ==>  Costo totale = 31.680000

Variazione 8:
Stazioni aperte:  3  6    ==>  Costo totale = 34.320000

Variazione 9:
Stazioni aperte:  3  6    ==>  Costo totale = 26.000000
```

Conclusioni

Dall'analisi si evince (con sorpresa) che cambia il valore della funzione obiettivo per ogni variazione, ma non cambia mai le stazioni che verranno eventualmente aperte!!!

Non solo !... si potrebbe estendendere il problema e verificare che per ogni valore delle funzioni dei prezzi dei materiali NON cambiano le stazioni che verranno aperte (data come fissa la dipendenza di *costo_j* dai materiali *m_k* e la matrice *servizio_ij*).

Questo si potrebbe dimostrare empiricamente facendo n simulazioni dove n è un numero molto grande. L'idea sarebbe di analizzare come varia la funzione obiettivo rispetto al tempo nelle n simulazioni.

Se la funzione 'non fa salti' (cioè il delta tra la la f(x) ob. della *i-esima* variazione e la f(x) ob. della *(i + 1)-esima* è infinitesimale) allora 'probabilmente' non cambierano mai le stazioni da aprire.

Oppure si potrebbe dimostrare formalmente se non richiede troppo sforzo. A volte è meglio una soluzione approssimata piuttosto che una esatta.

...ma cosa vogliono dire in concreto i risultati trovati???

Vuol dire che non solo la soluzione (stazioni: 3, 6) è ottima per il problema originale, ma è particolarmente resistente alle fluttuazioni di mercati.
Anche se i prezzi fluttuano abbastanza anche in maniera non proporzionale la decisione non è molto influenzata da questi ultimi.

Servirebbe, esempio, una catastrofe naturale sul territorio dove dovrebbe essere costruita la stazione 6 che renderebbe molto costoso (es.100 mln) la costruzione di quest'ultima per far diventare le *stazioni_ottime: 1, 3, 5* .

Tale risultato però è intuitivo, perchè osservando bene la coppia (3, 6) è l'unica complementare. Pertanto la seconda soluzione ottima è composta da un trio di stazioni. In altre parole dovrebbe esserci una variazione di costo così importante da far diventare la costruzione di 3 stazioni più conveniente di costruirne 2.