

problema di miscelazione

July 10, 2021

1 problema di miscelazione

Si vuole produrre un nuovo tipo di latte ottenuto miscelando le varietà di latte L1, L2 e L3 provenienti da tre diverse fattorie. Ogni varietà ha una composizione specifica di sostanze e un costo dato (vedi tabella).

La nuova miscela deve avere le caratteristiche nutrizionali minime (esprese in g/litro di ogni sostanza) riportate nell'ultima colonna della tabella.

| | L1 | L2 | L3 | requisiti minimi (g/litro) |
|-----------------------|-----|-----|-----|-------------------------------|
| proteine (g/litro) | 31 | 35 | 32 | 32 |
| grassi (g/litro) | 48 | 40 | 50 | 45 |
| carboidrati (g/litro) | 35 | 38 | 32 | 36 |
| calcio (g/litro) | 1.2 | 1.5 | 1.4 | 1.3 |
| costo (€/litro) | 0.5 | 0.6 | 0.4 | |

Qual è la composizione delle varietà di latte che costituisce la miscela richiesta di costo minimo?

1.1 modello parametrico

- Si vuole ottenere una miscela utilizzando n prodotti P_1, \dots, P_n .
- Di ogni prodotto si conosce il costo unitario e la composizione in sostanze S_1, \dots, S_m , ognuna presente in una data quantità.
- La miscela deve contenere almeno una percentuale stabilita di ognuna delle m sostanze.

Quali prodotti utilizzo e in quali proporzioni al fine di ottenere la miscela con le caratteristiche richieste al costo minimo?

- n numero di prodotti
- m numero di sostanze componenti
- a_{ij} q.tà di sostanza j contenuta in una unità di prodotto i
- c_i costo unitario del prodotto i
- b_j q.tà minima di sostanza j richiesta nella composizione di una unità di miscela.

variabili decisionali

- $x_i \in \mathbf{R}$: frazione di miscela costituita dal prodotto i

funzione obiettivo:
minimizzazione del costo totale

vincoli:

- Per ogni sostanza occorre garantire la q.tà minima richiesta
- Le variabili rappresentano frazioni, la loro somma quindi deve essere pari a 1...
- ... e sono numeri non negativi

$$z = \min \sum_{i=1}^n c_i x_i$$
$$\sum_{i=1}^n a_{ij} x_i \geq b_j \quad \forall j = 1, \dots, m$$
$$\sum_{i=1}^n x_i = 1$$
$$x_i \geq 0 \quad \forall i = 1, \dots, n$$

```

[3]: from pyomo.environ import *
from pyomo.opt import SolverStatus, TerminationCondition

model = ConcreteModel()
model.name = "problema di miscelazione"

i = [ 'L1', 'L2', 'L3' ]
c = [ 0.5, 0.6, 0.4 ]

j = [ 'proteine', 'grassi', 'carboidrati', 'calcio' ]
b = [ 32, 45, 36, 1.3 ]

a = [[31, 35, 32],
      [48, 40, 50],
      [35, 38, 32],
      [1.2, 1.5, 1.4]]

model.i = Set(initialize=i)
model.j = Set(initialize=j)

a_dict = {}
for j, mj in enumerate(model.j):
    for i, mi in enumerate(model.i):
        a_dict[mj, mi] = a[j][i]

c = {mi: c[i] for i, mi in enumerate(model.i)}
b = {mj: b[j] for j, mj in enumerate(model.j)}

model.c = Param(model.i, initialize=c)
model.b = Param(model.j, initialize=b)
model.a = Param(model.j, model.i, initialize=a_dict)

model.x = Var(model.i, domain=NonNegativeReals, initialize=0)

obj_expr = sum(model.c[i]*model.x[i] for i in model.i)
model.cost = Objective(expr = obj_expr, sense=minimize)

model.constraints = ConstraintList()

for j in model.j:
    model.constraints.add(expr = sum(model.a[j, i]*model.x[i] for i in model.i)
        ->>= model.b[j])

model.constraints.add(expr = sum(model.x[i] for i in model.i) == 1)

results = SolverFactory('glpk').solve(model)
model.display()

```

Model problema di miscelazione

Variables:

```
x : Size=3, Index=i
  Key : Lower : Value           : Upper : Fixed : Stale : Domain
  L1 :      0 : 0.55555555555553 : None  : False : False : NonNegativeReals
  L2 :      0 : 0.38888888888889 : None  : False : False : NonNegativeReals
  L3 :      0 : 0.05555555555556 : None  : False : False : NonNegativeReals
```

Objectives:

```
cost : Size=1, Index=None, Active=True
  Key : Active : Value
  None :      True : 0.533333333333332
```

Constraints:

```
constraints : Size=5
  Key : Lower : Body           : Upper
  1 : 32.0 : 32.61111111111111 : None
  2 : 45.0 : 44.99999999999997 : None
  3 : 36.0 : 35.99999999999999 : None
  4 : 1.3 : 1.327777777777778 : None
  5 : 1.0 : 0.999999999999997 : 1.0
```

[]: