

1 Background

We are faced with the following question:

How to minimize the cost of food eaten during one day, subject to the requirements that the diet satisfy a person's nutritional requirements and that not too much of any one food be eaten?

Suppose we have the following requirements

Food	Cost/Serving	Vitamin A	Calories	Maxium Servings
Corn	\$ 0.18	107	72	10
2% Milk	\$ 0.23	500	121	10
Wheat bread	\$ 0.05	0	65	10

Nutrient	Min. ammount in diet	max. ammount in diet
Calories	2000	2250
Vitamin A	5000	50000

2 Problems

(P1) Research Linear Programming

(P2) Find the optimal solution to our problem

Food	Servings
Corn	1.94
Milk	10.0
Bread	10.0

3 solutions

(P1) Linear programming is an optimization method for a problem whose cost function can be written through a linear expression

```
(P2) # modules
import numpy as np
from scipy.optimize import linprog
import sys

# problem
c = np.array([0.18, 0.23, 0.05])
b = np.array([50000, 2250, -5000, -2000])

A = np.zeros([4, 3])
A[0, :] = [107, 500, 0]
A[1, :] = [ 72, 121, 65]
A[2, :] = -A[0, :]
A[3, :] = -A[1, :]

x0_bounds = (0, 10)
x1_bounds = (0, 10)
x2_bounds = (0, 10)

# LP solver
res = linprog(c, A_ub = A, b_ub = b, bounds=(x0_bounds, x1_bounds, x2_bounds),
              options={"disp": True})

# results
print 'optimal solution:', res.x
print 'optimal cost:', res.fun

>>> Optimization terminated successfully.
      Current function value: 3.150000
      Iterations: 4
optimal solution: [ 1.94444444 10.          10.          ]
optimal cost: 3.15
```