Linear Programming & Dieting

Mario L. Gutierrez Abed

The maximum number of daily calories, C, that I would like to impose on my diet is $C = 2000\,\text{cal}$. The range of calories that I will impose for protein (E_{prot}), carbohydrates (E_{carbs}), and fats (E_{fats}) is as follows:

$$0.20C \leq E_{\text{prot}} \leq 0.25C \tag{1a}$$
$$0.35C \leq E_{\text{carbs}} \leq 0.40C \tag{1b}$$
$$0.48C \leq E_{\text{fats}} \leq 0.52C. \tag{1c}$$

Moreover, we impose the following nutritional requirements for sodium (Na), iron (Fe), and calcium (Ca) intake [1]

$$2.0g \leq \text{Na} \leq 2.3g \tag{2a}$$
$$0.006g \leq \text{Fe} \leq 0.010g \tag{2b}$$
$$0.8g \leq \text{Ca} \leq 1.0g. \tag{2c}$$

The following table (which took me ages to put together!) contains all the relevant data:

Foods	Serving (g)	Price ($)	Protein (cal)	Carbs (cal)	Fats (cal)	Na (g)	Fe (g)	Ca (g)
Wheat Bread (x_0)	76	0.20	32	144	18	0.30	0.0016	0.26
Cottage Cheese (x_1)	113	0.90	52	20	14	0.39	0.0002	0.11
Chicken ham (x_2)	51	1.00	32	4	9	0.38	0.0002	0
Eggs (x_3)	100	0.20	48	0	90	0.14	0.0004	0.03
Coconut Oil (x_4)	14	0.14	0	0	126	0	0	0
Parmesan Cheese (x_5)	28	0.50	36	4	63	0.25	0	0.20
Porcini Ravioli (x_6)	166	2.10	64	208	216	1.12	0	0
Olive Oil (x_7)	30	0.15	0	0	252	0	0	0
Ground Beef (x_8)	113	4.00	124	0	180	0.103	0.0014	0.024
Sweet Potato (x_9)	133	0.80	8	108	1	0.073	0.0003	0.04
Black Beans (x_{10})	122	0.45	32	92	5	0.41	0.002	0.048
Yogurt (x_{11})	113	0.60	16	60	14	0.055	0	0.10

Table 1: List of twelve foods to be eaten over the course of a day. The table shows i) the serving size in grams (g); ii) the price (in $USD) of each item, per serving; ii) the amount of calories (rounded) consumed from protein, carbohydrates, and fats/lipids, per serving size; iv) amount of sodium (Na), iron (Fe), and calcium (Ca), per serving size, all expressed in grams.

Figure 1: Nutrition label for (whole wheat!) bread.

[1] I determined these numbers based on advise found on the Mayo Clinic website.

I found these values from nutrition labels (see, e.g., Figure 1). For all items I combined the serving size and number of helpings into a single quantity; for instance, for wheat bread the serving size is one slice ($38g$), while I tipically have two slices for breakfast (hence the serving size shown on the table is $38 \times 2 = 76g$). In order to calculate the calories pertaining to carbohydrates, protein, and fats, I used information found on the USDA webpage, which states that there are nine calories per gram of fat, and four calories per gram of either carbohydrates or protein.

Objective Function: We would like to minimize the total cost T given by

$$T = 0.2x_0 + 0.9x_1 + x_2 + 0.2x_3 + 0.14x_4 + 0.5x_5 + 2.1x_6 + 0.15x_7 + 4x_8 + 0.8x_9 + 0.45x_{10} + 0.6x_{11}. \tag{3}$$

Constraints: Moreover, we need to satisfy the set of constraints presented in (1)-(2). Using the data from the table and plugging into these inequalities (and substituting the value of total allowed calories $C = 2000\,\text{cal}$), we get

$$\begin{aligned}
400 \leq &\ 32x_0 + 52x_1 + 32x_2 + 48x_3 + 0x_4 + 36x_5 + 64x_6 + 0x_7 + 124x_8 + 8x_9 + 32x_{10} + 16x_{11} && \leq 500\\
700 \leq &\ 144x_0 + 20x_1 + 4x_2 + 0x_3 + 0x_4 + 4x_5 + 208x_6 + 0x_7 + 0x_8 + 108x_9 + 92x_{10} + 60x_{11} && \leq 800\\
960 \leq &\ 18x_0 + 14x_1 + 9x_2 + 90x_3 + 126x_4 + 63x_5 + 216x_6 + 252x_7 + 180x_8 + x_9 + 5x_{10} + 14x_{11} && \leq 1040\\
2 \leq &\ 0.3x_0 + 0.39x_1 + 0.38x_2 + 0.14x_3 + 0x_4 + 0.25x_5 + 1.12x_6 + 0x_7 + 0.103x_8 + 0.073x_9 + 0.41x_{10} + 0.055x_{11} && \leq 2.3\\
0.006 \leq &\ 0.0016x_0 + 0.0002x_1 + 0.0002x_2 + 0.0004x_3 + 0x_4 + 0x_5 + 0x_6 + 0x_7 + 0.0014x_8 + 0.0003x_9 + 0.002x_{10} + 0x_{11} && \leq 0.01\\
0.8 \leq &\ 0.26x_0 + 0.11x_1 + 0x_2 + 0.03x_3 + 0x_4 + 0.2x_5 + 0x_6 + 0x_7 + 0.024x_8 + 0.04x_9 + 0.048x_{10} + 0.1x_{11} && \leq 1.
\end{aligned}$$

Thus we have the matrix of constraints

$$M = \begin{bmatrix} 32 & 52 & 32 & 48 & 0 & 36 & 64 & 0 & 124 & 8 & 32 & 16\\ 144 & 20 & 4 & 0 & 0 & 4 & 208 & 0 & 0 & 108 & 92 & 60\\ 18 & 14 & 9 & 90 & 126 & 63 & 216 & 252 & 180 & 1 & 5 & 14\\ 0.3 & 0.39 & 0.38 & 0.14 & 0 & 0.25 & 1.12 & 0 & 0.103 & 0.073 & 0.41 & 0.055\\ 0.0016 & 0.0002 & 0.0002 & 0.0004 & 0 & 0 & 0 & 0 & 0.0014 & 0.0003 & 0.002 & 0\\ 0.8 & 0.11 & 0 & 0.03 & 0 & 0.2 & 0 & 0 & 0.024 & 0.04 & 0.048 & 0.1 \end{bmatrix} \tag{4}$$

as well as the constraint vectors

$$b_{\min} = \begin{bmatrix} 400\\ 700\\ 960\\ 2\\ 0.006\\ 0.8 \end{bmatrix} \quad \text{and} \quad b_{\max} = \begin{bmatrix} 500\\ 800\\ 1040\\ 2.3\\ 0.01\\ 1 \end{bmatrix}. \tag{5}$$

Moreover, I will also add the constraint

$$x_i \geq 0.5 \ \forall i \tag{6}$$

to ensure that I will eat at least half a serving/helping from each food (recall that I combined servings and helpings into a single quantity). Hence the Sage code that was provided in class was modified as follows:

```
#LP for twelve foods; written in Sage


#coefficients of objective function (Eq 3 in assignment sheet)
c = Matrix([[0.2, 0.9, 1.0, 0.2, 0.14, 0.5, 2.1, 0.15, 4.0, 0.8, 0.45, 0.6]]);


```

```
#Constraint matrix (Eq 4 in assignment sheet)
M = Matrix([[32, 52, 32, 48, 0, 36, 64, 0, 124, 8, 32, 16],
            [144, 20, 4, 0, 0, 4, 208, 0, 0, 108, 92, 60],
            [18, 14, 9, 90, 126, 63, 216, 252, 180, 1, 5, 14],
            [0.3, 0.39, 0.38, 0.14, 0, 0.25, 1.12, 0, 0.103, 0.073, 0.41, 0.055],
            [0.0016, 0.0002, 0.0002, 0.0004, 0, 0, 0, 0, 0.0014, 0.0003, 0.002, 0],
            [0.8, 0.11, 0, 0.03, 0, 0.2, 0, 0, 0.024, 0.04, 0.048, 0.1]
           ])


#Constraint vectors (Eq 5 in assignment sheet)
b_min = Matrix([[400],
                [700],
                [960],
                [2],
                [0.006],
                [0.8]
               ]);

b_max = Matrix([[500],
                [800],
                [1040],
                [2.3],
                [0.01],
                [1]
               ]);

p = MixedIntegerLinearProgram(maximization=False, solver = "GLPK");
x = p.new_variable(integer=False, nonnegative=True);

#add constraint b_min <= M*x <= b_max
for i in range(0,M.nrows()):
    expr = 0;
    for j in range(0,M.ncols()):
        p.add_constraint(x[j]>=0.5);   #secures that I'll get to eat at least half a serving of each food
        expr = expr+M[i,j]*x[j];
    p.add_constraint(expr>=b_min[i,0]);
    p.add_constraint(expr<=b_max[i,0]);

#add objective c*x
obj = 0;
for i in range(0,c.ncols()):
    obj = obj+c[0,i]*x[i];
p.set_objective(obj);
p.solve();

print(p.get_objective_value(),p.get_values(x));
```

The output shows

```
10.084747060427068 {0: 0.6483313634080113, 1: 0.5, 2: 0.5,
                    3: 2.5259912210675255, 4: 0.5, 5: 0.5,
                    6: 0.5, 7: 1.3477152171724225, 8: 0.8503198793325981,
                    9: 3.617795230496156, 10: 0.7382434649528435, 11: 0.5}
```

Thus the program has decided that the optimal solution would be to spend roughly $10 daily on my selected foods, given the imposed constraints. I am quite happy with the servings I am getting for each food, except I only get to eat one slice (and a meager tiny piece extra) of wheat bread ☹