MATH3161 Optimisation

Nathan Wilson

June 6, 2014

Minimum Cost Pizza Problem

Using only the items given in the tables below (See Appendix A, Assignment Sheet), create a minimum cost pizza which satisfies both the nutritional requirements of Table 1 and bounds on item quantities given in Table 2. Use the nutritional data of Table 3 and the cost data of Table 4 in your model.

Solution Summary

The minimum cost pizza calculated using the model below costs $2.40, and consists of the following ingredients:

Ingredient	Quantity(grams)
Dough	425.6
Sauce	113.5
Cheese	170.3
Ham	99.3
Onion	90.333
Mushrooms	92.2
Green Pepper	8.967

Table 1: Ingredient Quantities

Model formulation

Objective

Our objective is to minimize total cost of ingredients.

Let $x_i = x_1, \ldots, x_n$ be the quantity of each ingredient i in hundreds of grams.

Let $c_i = c_1, \ldots, c_n$ be the cost of ingredient x_i per 100 grams.

The cost of the pizza is equal to the sum of the costs of each ingredient, which can be represented by the objective function:

$$f = \sum_{i=1}^{12} c_i x_i \tag{1}$$

As f is a linear function it is convex and therefore is a valid linear optimisation problem.

Equality Constraints

This problem does not have any equality constraints.

Inequality Constraints

We are given a number of constraints that can be represented via linear equations.

Bounds on quantities of Ingredients:

Let ingredient in_i, i=1..12 be

i	in_i
1	Cheese
2	Sauce
3	Dough
4	Pepperoni
5	Ham
6	Bacon
7	Green Pepper
8	Onion
9	Celery
10	Mushrooms
11	Tomatoes
12	Pineapple

Table 2: Ingredient indices

Let nutrient n_i, i=1..8 be

i	n_i
1	Calcium
2	Iron

i	n_i
3	Protein
4	Vitamin A
5	Thiamine
6	Niacin
7	Riboflavin
8	Vitamin C

Table 3: Nutrient indices

We are given a table of upper and lower bounds on ingredient quantities which can be represented in a series of inequality constraints g(x). In standard form these are:

$$-x_1 + 1.703 <= 0 \tag{2}$$

$$x_1 - 2.270 <= 0 \tag{3}$$

$$-x_2 + 1.135 <= 0 \tag{4}$$

$$x_2 - 1.986 <= 0 \tag{5}$$

$$-x_3 + 4.256 <= 0 \tag{6}$$

$$x_3 - 5.249 <= 0 \tag{7}$$

$$x_4 - 0.993 <= 0 \tag{8}$$

$$x_5 - 1.135 <= 0 \tag{9}$$

$$x_6 - 0.993 <= 0 \tag{10}$$

$$x_7 - 1.561 <= 0 \tag{11}$$

$$x_8 - 0.993 <= 0 \tag{12}$$

$$x_9 - 1.561 <= 0 \tag{13}$$

$$x_{10} - 1.135 <= 0 \tag{14}$$

$$x_{11} - 1.703 <= 0 \tag{15}$$

$$x_{12} - 1.703 <= 0 \tag{16}$$

$$-x_4, -x_5, -x_6, -x_7, -x_8, -x_9, -x_{10}, -x_{11}, -x_{12} <= 0 \tag{17}$$

Three more constraints also refer to bounds on 'classes' of ingredient, where the classes are Meat, Vegetables, and Fungi. The ingredient set in each class was not given, so assumptions have been made as follows:

Category	Ingredient
Meat	Pepperoni, Ham, Bacon
Vegetables	Green Pepper, Onion, Celery
Fungi	Mushrooms

Table 4: Ingredient classes

Hence:

$$0.993 - x4 - x5 - x6 <= 0 \tag{18}$$

$$0.993 - x7 - x8 - x9 <= 0 \tag{19}$$

$$0.993 - x10 <= 0 \tag{20}$$

Note that as tomato and pineapple are both technically fruits, they were not included in these constraints.

All g(x) are convex as they are linear functions, thus the constraints represent a valid linear optimisation problem in standard form.

Bounds on quantities of Nutrients:
The amount of a nutrient q_i (as defined in Table 3 above) can be calculated by:

$$f(i) = \sum_{j=1}^{8} n_{i,j} x_j \tag{21}$$

$$f(i) = n_1x_1 + n_2x_2 + n_3x_3 + n_4x_4 + n_5x_5 + n_6x_6 + n_7x_7 + n_8x_8 \tag{22}$$

where n_j is the quantity of nutrient q_i for x_j per 100grams. Obtained from Appendix A, Table 3. In Appendix A, Table 1, we are given minimum requirements r_i for each nutrient q_i. We can represent these requirements as inequality constraints using the formula above where f(x) >= r_i.

In standard form, this is:

$$-1 * \sum_{j=1}^{8} n_{i,j} x_j <= -r_i, \forall i \in 1, ..., 8 \tag{23}$$

Both sides have been multiplied by -1 to get the inequality into standard form.

Definition of problem variables

The cost per 100g for each ingredient:

$$C = \begin{bmatrix} 46.6 \\ 35.24 \\ 9.63 \\ 44.11 \\ 44.05 \\ 44.27 \\ 25.10 \\ 5.33 \\ 14.14 \\ 31.20 \\ 22.03 \\ 25.98 \end{bmatrix}. \tag{24}$$

Matrix of nutrient quantity per ingredient (where the final three rows represent the ingredient class constraints):

$$A = -1* \begin{bmatrix} 517.7 & 14 & 18.233 & 10 & 9.031 & 13 & 9.459 & 27.273 & 40 & 6 & 13.333 & 12.016 \\ .222 & 1.8 & 3.826 & 2.5 & 2.291 & 1.189 & .675 & 0.545 & 0.250 & 0.800 & 0.533 & .310 \\ 20 & 2 & 14.224 & 15 & 14.692 & 8.392 & 1.351 & 1.818 & 0 & 3 & 1.333 & 0.387 \\ 3000 & 800 & 0 & 0 & 0 & 0 & 209.46 & 18.182 & 125 & 0 & 450 & 25.194 \\ .022 & 0.1 & 0.586 & 0 & 0.74 & 0.361 & 0.081 & 0.363 & 0.025 & 0.100 & 0.066 & 0.081 \\ 6 & 1.4 & 8.852 & 2 & 4.009 & 1.828 & 0.540 & 0.545 & 0.5 & 4.3 & 0.8 & 0.193 \\ 0.244 & 0.060 & 0.628 & 0 & 0.178 & 0.114 & 0.081 & 0.036 & 0.025 & 0.460 & 0.04 & 0.019 \\ 0 & 6 & 0 & 0 & 0 & 0 & 127.03 & 10 & 10 & 3 & 22.667 & 6.977 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \tag{25}$$

Minimum nutrient quantities:

$$b = -1 * \begin{bmatrix} 750 \\ 12 \\ 48.5 \\ 4500 \\ 1.3 \\ 16 \\ 1.6 \\ 30 \\ 0.993 \\ 0.993 \\ 0.922 \end{bmatrix}. \tag{26}$$

Upper and lower bounds on ingredients:

$$xlb = \begin{bmatrix} 1.703 \\ 1.135 \\ 4.256 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{27}$$

$$xlb = \begin{bmatrix} 2.270 \\ 1.986 \\ 5.249 \\ 0.993 \\ 1.135 \\ 0.993 \\ 1.561 \\ 0.993 \\ 1.561 \\ 1.135 \\ 1.703 \\ 1.703 \end{bmatrix}. \tag{28}$$

Computer Output

```
>> ss14
Optimization terminated.
Minimum cost of pizza: $2.399160
Quantity of          Cheese: 170.300 grams
Quantity of           Sauce: 113.500 grams
Quantity of           Dough: 425.600 grams
Quantity of       Pepperoni: 0.000 grams
Quantity of             Ham: 99.300 grams
Quantity of           Bacon: 0.000 grams
Quantity of    Green Pepper: 8.967 grams
Quantity of           Onion: 90.333 grams
Quantity of          Celery: 0.000 grams
Quantity of       Mushrooms: 92.200 grams
Quantity of        Tomatoes: 0.000 grams
Quantity of       Pineapple: 0.000 grams
```

Portfolio Selection Problem

An individual with $10,000 to invest has identified three mutual funds as attractive opportunities. Over the last five years, dividend payments (in cents per dollar invested) have been as shown in Table 5, Appendix A, and the individual assumes that these payments are indicative of what can be expected in the future. This particular individual has two requirements: (1) the combined expected yearly return from his/her investments must be no less than $800 (the amount $10,000 would earn at 8 percent interest) and (2) the variance in future, yearly, dividend payments should be as small as possible. How much should this individual invest in each fund to achieve these requirements?

Solution Summary

Following the imposed requirements, given total assets of $10,000 he should invest his money as below:

Investment	%	Amount
1	10.14%	$1013.78
2	63.29%	$6328.74
3	26.57%	$2657.48

In order to achieve a minimum possible variance of 1.812402.

Model formulation

Objective

Our objective is to minimise the variance of future yearly dividend payments. This can be represented by the function as given (Appendix A):

$$f(x_1, x_2, x_3) = \sum_{i=1}^{3} \sum_{j=1}^{3} \sigma_{ij}^2 x_i x_j \tag{29}$$

where x_1, x_2, x_3 are the proportion to invest in each investment respectively.

Equality Constraints

This problem does not have any equality constraints.

Inequality Constraints

We are given several requirements that can be represented as inequality constraints.

Minimum Expected Return

First we are told that the yearly return from dividends must be greater than or equal to $800, or the value earned at 8 percent interest. The expected return for each investment is calculated from Table 5 using the equation:

$$E_i = 1/5 \sum_{k=1}^{5} x_{ik} \tag{30}$$

where x_{ik} denotes the return per dollar invested from investment i during the kth time period in the past ($k = 1, 2, ..., 5$).

The calculated values from Table 5 gave:

	Investment 1	Investment 2	Investment 3
% Return	9	7	10

Hence the expected return must be greater than or equal to 8%:

$$0.09x_1 + 0.07x_2 + 0.1x_3 >= 0.08 \tag{31}$$

or in standard form:

$$-0.09x_1 - 0.07x_2 - 0.1x_3 <= -0.08 \tag{32}$$

Bound on Total Investment

We must also introduce a constraint to represent the fact he cannot invest more than 100% of his assets. In some areas of finance it is possible to do this, but we are given the fact that the investments are mutual funds, and you cannot short stock on a mutual fund. Therefore we may assume that the total investment must be less than or equal to 100%:

$$x_1 + x_2 + x_3 <= 1 \tag{33}$$

We also place upper and lower bounds on $x_{1,2,3}$ to to represent the fact it is a proportion:

$$-x_1, -x_2, -x_3 <= 0 \tag{34}$$

$$x_1, x_2, x_3 <= 1 \tag{35}$$

Computer Output

```
>> qp14
Optimization terminated.
Minimum variance on future yearly dividend payments: 1.812402
Proportion to invest in Investment 1: 10.1378%
Proportion to invest in Investment 2: 63.2874%
Proportion to invest in Investment 3: 26.5748%
```

Appendix A

Please see attached.

Appendix B

Problem 1 Code

```
% MATH3161/MATH5165 - Optimization
% Script to solve Assignment Question 1 - Minimal cost pizza
% Nathan Wilson
% z3287546

format compact
%format short e

% Objective gradient (semi-colons ; for column vector)
g = [46.6; %cheese
    35.24; %sauce
    9.63;  %dough
    44.11; %pepperoni
    44.05; %ham
    44.27; %bacon
    25.10; %green pepper
    5.33;  %onion
    14.14; %celery
    31.20; %mushrooms
    22.03; %tomatoes
    25.98];%pineapple

labels = cell(12,1);
labels{1} = 'Cheese';
labels{2} = 'Sauce';
labels{3} = 'Dough';
labels{4} = 'Pepperoni';
labels{5} = 'Ham';
labels{6} = 'Bacon';
labels{7} = 'Green Pepper';
labels{8} = 'Onion';
labels{9} = 'Celery';
labels{10} = 'Mushrooms';
labels{11} = 'Tomatoes';
labels{12} = 'Pineapple';

% General linear constraints (by rows)
A = [-517.7  -14    -18.233  -10  -9.031   -13     -9.459    -27.273  -40     -6      -13.333  -12.016;
      -.222   -1.8   -3.826   -2.5 -2.291   -1.189  -.675     -0.545   -0.250  -0.800  -0.533   -.310;
      -20     -2     -14.224  -15  -14.692  -8.392  -1.351    -1.818   0       -3      -1.333   -0.387;
      -3000   -800   0        0    0        0       -209.46   -18.182  -125    0       -450     -25.194;
```

```
     -.022   -0.1   -0.586   0     -0.74    -0.361  -0.081
         -0.363   -0.025  -0.100  -0.066   -0.081;
     -6      -1.4   -8.852   -2    -4.009   -1.828  -0.540
         -0.545   -0.5    -4.3    -0.8     -0.193;
     -0.244  -0.060 -0.628   0     -0.178   -0.114  -0.081
         -0.036   -0.025  -0.460  -0.04    -0.019;
     0       -6     0        0     0        0       -127.03
         -10      -10     -3      -22.667  -6.977;
% meats >= 0.993
% veg >= 0.993
% fung >= 0.992
     0       0      0        -1    -1       -1      0
         0        0       0       0        0;
     0       0      0        0     0        0       -1
         -1       -1      0       0        0;
     0       0      0        0     0        0       0
         0        0       -1      0        0;

    ];

b = [-750; %calcium
     -12;  %iron
     -48.5;%prot
     -4500;%vitamin a
     -1.3; %Thiamine
     -16;  %Niacin
     -1.6; %Riboflavin
     -30;  %Vitamin C
     -0.993;%Meats
     -0.993;%Veg
     -0.922];%Fungi

% Equality constraints = none
Aeq = [];
beq = [];

% Simple lower and upper bounds on the variables
xlb = [1.703; 1.135; 4.256; 0;     0;     0;     0;     0;
       0;     0;     0;     0];
xub = [2.270; 1.986; 5.249; 0.993; 1.135; 0.993; 1.561;
    0.993; 1.561; 1.135; 1.703; 1.703];

% Solution x and Lagrange multipliers lm
xlp = linprog(g, A, b, Aeq, beq, xlb, xub);

%Objective
obj = g'*xlp;

fprintf(1,'Minimum cost of pizza: $%f\n', obj/100);

```

```
for i = 1:length(xlp)
    fprintf(1,'Quantity of %s: %.3f grams\n', sprintf('%15s
        ',labels{i,1}), xlp(i)*100);
end
```

Appendix C

Problem 2 Code

```
% MATH3161/MATH5165 - Optimization
% Script to solve Assignment Question 2 - Portfolio Selection Problem
% Nathan Wilson
% z3287546
format compact
%format short e

%Quadratic programming formulation
%Hessian of quadratic and gradient at 0
G = 2 .* [30 -5.6 23;
          -5.6 2.8 -12;
          23 -12 55.2];
g0 = [0; 0; 0];
c = [0];
%General linear constraints
A = [-0.09  -0.07  -0.1;
    1 1 1];
b = [-0.08;
     1];
% No equality constraints for this problem so use empty matrices
Aeq = [];
beq = [];
% Lower bounds
xlb = [0; 0; 0];
%Upper bounds
xub = [1; 1; 1];

% Solve using Matlab QP routine
xqp = quadprog(G, g0, A, b, [], [], xlb, xub);


% Objective
obj = 0.5*xqp'*G*xqp + g0'*xqp + c;

fprintf(1,'Minimum variance on future yearly dividend payments: %f\n', obj);

for i = 1:length(xqp)
    fprintf(1,'Proportion to invest in Investment %d: %.4f%%\n', i, xqp(i)*100);
end
```