The Art of the Steal

MATH 381 HW 1

Tim Ferrin, 1764309

October 9, 2020

Setup

I am the head of a large criminal organization and next week three teams of my henchmen are going to rob three different locations that store precious artwork. Each team consists of men of equal ability, driving getaway cars of equal size and volume capacity. Each can hold a maximum of 1000 kg in a space of 100 L due to the physical constraints of the car. At each location the same 50 items are available for stealing and only those 50 items. Each item weighs the same amount, is the same volume, and has been established to be the same monetary value as at the other locations. The teams are hitting locations described in the following way:

1. The first team will target an art museum which contains only one copy of each piece of art.
2. The second team will target a different art museum which also holds only one copy of each piece of art. However at this location, there are security sensors placed on some of the composite (non-prime) numbered pieces of artwork. In order to minimize risk of getting caught, I have decided that 70% of the total value stolen by this team must be from prime numbered pieces.
3. The third team will target a warehouse depot storing a large number of duplicates of the artwork. This means that the team can steal as many copies of any particular piece as possible.

Each team is trying to maximize the total value of the items they steal while operating under their individual conditions and the generic weight and volume constraints stated above.

Mathematical Model

In order to figure out which pieces of art each team should steal, we will formulate these conditions into a linear programming problem. Each team is essentially a different linear program so we will use the same variable names for each case.

Each item will be labeled with an index i which will take on a value 1-50. How many of each of these items is stolen from each location will be labeled as x_i.

The **value** v_i in thousands of dollars of each item is given by the function

$$v_i = \lfloor 100 + 50\cos(i) \rfloor, \tag{1}$$

where $\lfloor \cdot \rfloor$ represents the floor function that rounds any number between the two symbols down to the largest integer less than or equal to that number.

The **weight** w_i in kilograms of each item is given by the function

$$w_i = \lfloor 100 + 50\cos(7i + 1) \rfloor. \tag{2}$$

The **volume** s_i in liters of each item is given by the function

$$s_i = \lfloor 20 + 10\cos(8i - 3) \rfloor. \tag{3}$$

So altogether, the quantity each team of thieves is trying to maximize is the sum of the values of each item times the number of those items they steal. Represented mathematically our objective function is

$$v_1x_1 + v_2x_2 + ... + v_{50}x_{50}, \tag{4}$$

or more compactly

$$\sum_{i=1}^{50} v_i x_i. \tag{5}$$

The first constraint is the constraint of weight, which must be under 1000 kg for each team. Represented mathematically this is

$$w_1x_1 + w_2x_2 + ... + w_{50}x_{50} \leq 1000, \tag{6}$$

or more compactly

$$\sum_{i=1}^{50} w_i x_i \leq 1000. \tag{7}$$

The second constraint is that of volume, which must be under 100 L for each team. Represented mathematically this is

$$s_1x_1 + s_2x_2 + ... + s_{50}x_{50} \leq 100, \tag{8}$$

or more compactly

$$\sum_{i=1}^{50} s_i x_i \leq 100. \tag{9}$$

While all three teams must abide by these two constraints, there is a unique constraint for team 2. Because of the prime number requirement, team 2's total value must be made up of at least 70% of prime numbered items. Represented mathematically this is

$$v_2x_2 + v_3x_3 + v_5x_5 + ... + v_{43}x_{43} + v_{47}x_{47} \geq 0.7(v_1x_1 + v_2x_2 + ... + v_{50}x_{50}). \tag{10}$$

By subtracting the right-hand side from both sides, this can be simplified down to

$$-0.7v_1x_1 + 0.3v_2x_2 + 0.3v_3x_3 - 0.7v_4x_4 + ... - 0.7v_{49}x_{49} - 0.7v_{50}x_{50} \geq 0, \tag{11}$$

or more compactly

$$0.3\left(\sum_{i \text{ is prime}} v_i x_i\right) - 0.7\left(\sum_{i \text{ isn't prime}} v_i x_i\right) \geq 0. \tag{12}$$

As stated in the setup, teams 1 and 3 must abide by only the first two constraints, but team 2 must abide by all three. Additionally, teams 1 and 2 can only take 0 or 1 of each item, whereas team 3 can take any non-negative integer amount of any item.

Linear Programs

As described above the three linear programs that need to be solved are

1. Maximize
$$\sum_{i=1}^{50} v_i x_i$$
subject to
$$\sum_{i=1}^{50} w_i x_i \leq 1000$$
$$\sum_{i=1}^{50} s_i x_i \leq 100$$
where x_i is either 0 or 1.

2. Maximize
$$\sum_{i=1}^{50} v_i x_i$$
subject to
$$\sum_{i=1}^{50} w_i x_i \leq 1000$$
$$\sum_{i=1}^{50} s_i x_i \leq 100$$
$$0.3\left(\sum_{i \text{ is prime}} v_i x_i\right) - 0.7\left(\sum_{i \text{ isn't prime}} v_i x_i\right) \geq 0$$
where x_i is either 0 or 1.

3. Maximize
$$\sum_{i=1}^{50} v_i x_i$$
subject to
$$\sum_{i=1}^{50} w_i x_i \leq 1000$$
$$\sum_{i=1}^{50} s_i x_i \leq 100$$
where x_i is a non-negative integer.

Code

The code to generate the linear program was written in MATLAB. This linear program was written to an .lp file which is then opened in LPSolve IDE which actually solved the LPs. The commented MATLAB code is written below.

```
%% Initialization
% These lines calculate the values, weights and volumes of each of the
% items, indexed by is 1-50.

is = 1:50;
values = floor(100 + 50*cos(is));
weights = floor(100 + 50*cos(7*is + 1));
volumes = floor(20 + 10*cos(8*is - 3));

%% Part a)

% The next three lines concatenate the object function and constraint
% coefficients and indices and flatten them for later use.
objective = reshape([(values).', is.'].',1,[]);
constraint1 = reshape([weights.', is.'].',1,[]);
constraint2 = reshape([volumes.', is.'].',1,[]);

% The remaining blocks of code in this part print the LP to the .lp file
fileID = fopen('PartA.lp', 'w');
fprintf(fileID, 'max: ');
fprintf(fileID, '%dx_%d + ', objective(1:98));
% ^First prints the coefficient, then x_, then the index and the +
fprintf(fileID, [num2str(objective(99)),'x_50;\n']);
% ^Solves a formatting issue where a + was on the end of the line

% Carries out similar process now for the constraints
fprintf(fileID, '%dx_%d + ',constraint1(1:98));
fprintf(fileID, [num2str(constraint1(99)),'x_50 <= 1000;\n']);

fprintf(fileID, '%dx_%d + ',constraint2(1:98));
fprintf(fileID, [num2str(constraint2(99)),'x_50 <= 100;\nbin ']);
% ^Specifies that the numbers must be binary

fprintf(fileID, 'x_%d,',is(1:49));
fprintf(fileID, 'x_50;');
fclose(fileID);

%% Part b)
% This section of code is the same as the previous except for the third
% constraint

primeMarkers = ismember(is,primes(50))-.7*ones(1,50);
% ^Calculates the coeffiecients for the third constraint
constraint3 = reshape([primeMarkers.', is.'].',1,[]);

fileID = fopen('PartB.lp', 'w');
fprintf(fileID, 'max: ');
fprintf(fileID, '%dx_%d + ', objective(1:98));
fprintf(fileID, [num2str(objective(99)),'x_50;\n']);

fprintf(fileID, '%dx_%d + ',constraint1(1:98));
fprintf(fileID, [num2str(constraint1(99)),'x_50 <= 1000;\n']);

fprintf(fileID, '%dx_%d + ',constraint2(1:98));
```

```
fprintf(fileID, [num2str(constraint2(99)),'x_50 <= 100;\n']);

fprintf(fileID, '%2.1fx_%d + ',constraint3(1:98));
fprintf(fileID, [num2str(constraint3(99)),'x_50 >= 0;\nbin ']);

fprintf(fileID, 'x_%d,',is(1:49));
fprintf(fileID, 'x_50;');
fclose(fileID);

%% Part c)
% This section of code is the same as Part A except that it specifies that
% the variables can be non-negative integers

fileID = fopen('PartC.lp', 'w');
fprintf(fileID, 'max: ');
fprintf(fileID, '%dx_%d + ', objective(1:98));
fprintf(fileID, [num2str(objective(99)),'x_50;\n']);

fprintf(fileID, '%dx_%d + ',constraint1(1:98));
fprintf(fileID, [num2str(constraint1(99)),'x_50 <= 1000;\n']);

fprintf(fileID, '%dx_%d + ',constraint2(1:98));
fprintf(fileID, [num2str(constraint2(99)),'x_50 <= 100;\nint ']);

fprintf(fileID, 'x_%d,',is(1:49));
fprintf(fileID, 'x_50;');
fclose(fileID);
```

Results

1. Team 1 should take items 7, 11, 18, 25, 26, 29, 33, 37, and 44. This comes to a value total of $1,099,000, a weight total of 979 kg, and a volume total of 99 L.

2. Team 2 should take items 7, 11, 18, 19, 29, 37, 43, and 44. This comes to a value total of $995,000, a weight total of 772 kg, and a volume total of 98 L. The value percentage of prime-indexed items is 71.7%. Notice that this solution includes items 19 and 43 which aren't in the first teams solution and doesn't have items 25, 26, and 33 which are.

3. Team 3 should take 1 copy of item 19, 1 copy of item 37, and 7 copies of item 44. This comes to a value total of $1,330,000, a weight total of 986 kg, and a volume total of 100 L. Notice that this team's solution is mostly made up of item 44 and all three of its item types are either in the team 1 solution, the team 2 solution, or both.