The Buyer-Seller Problem

Takemyprocs

September 4, 2021

1 The Seller Problem: a basic framework

Suppose a seller is endowed with a fixed amount of m inputs, namely: $s_1, s_2, ..., s_m$. With them the seller can craft n products, namely: $x_1, x_2, ..., x_n$. Not all inputs may be required for crafting a particular product, so we define A as the technical matrix that captures the input requirements for each output:

$$A = \begin{array}{c c} & \begin{array}{cccc} x_1 & x_2 & ... & x_n \end{array} \\ \begin{array}{c} s_1 \\ s_2 \\ ... \\ s_m \end{array} & \begin{pmatrix} a_{11} & a_{12} & ... & a_{1n} \\ a_{21} & a_{22} & ... & a_{2n} \\ ... & ... & \ddots & ... \\ a_{m1} & a_{m2} & ... & a_{mn} \end{pmatrix} \end{array}$$

For instance, in order to craft 1 unit of product x_1 we need to use a_{11} units of input s_1, a_{21} units of input s_2, and so on. We assume technical coefficients a_{ij} are fixed. Recall that since not all inputs are required for crafting a particular product some of the elements in this matrix may be 0. Finally, we assume that only outputs can be sold freely at fixed prices: $p_1, p_2, ..., p_n$.

Given those conditions, we can write the seller problem as follows:

$$\begin{aligned} \max_{x_1, x_2, ..., x_n} \quad & p_1x_1 + p_2x_2 + ... + p_nx_n \\ \text{s.t.} \quad & a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n \leq s_1 \\ & a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \leq s_2 \\ & \cdots \\ & a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n \leq s_m \\ & x_1, x_2, ..., x_n \geq 0 \end{aligned} \tag{1}$$

Intuitively, the seller problem is to choose production quantities $x_1, x_2, ..., x_n$ in order to maximize earnings subject to input availability constraints. This resembles the linear programming canonical form. Therefore, we can use almost any mathematical software (e.g. Matlab, R or Python) to solve this problem. In order to apply this model we need information on technical coefficients, input endowments and prices. We provide an example in the next section.

2 An example

Suppose a seller is endowed with 98 units of input 1, 147 units of input 2, 182 units of input 3, 157 units of input 4 and 114 units of input 5. By using those inputs the seller can craft up to 6 different products, say $x_1, ..., x_6$. The input requirements for crafting each product are summarized in the technical matrix shown below:

$$A = \begin{array}{c} \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{array} \begin{array}{c} \begin{array}{cccccc} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{array} \\ \begin{pmatrix} 5 & 0 & 0 & 0 & 0 & 4 \\ 0 & 5 & 0 & 0 & 0 & 4 \\ 0 & 0 & 5 & 0 & 3 & 4 \\ 0 & 0 & 0 & 5 & 3 & 4 \\ 0 & 0 & 0 & 0 & 1 & 3 \end{pmatrix} \end{array}$$

Finally, market prices for products 1 to 6 are 100.2, 80, 110.96, 120.11, 115.90 and 314 respectively. According to the basic framework described in the previous section we formulate the seller problem for this particular application as follows:

$$\begin{aligned} \max_{x_1, x_2, \dots, x_n} \quad & 100.2x_1 + 80x_2 + 110.96x_3 + 120.11x_4 + 115.90x_5 + 314x_6 \\ \text{s.t.} \quad & 5x_1 + 4x_6 \leq 98 \\ & 5x_2 + 4x_6 \leq 147 \\ & 5x_3 + 3x_5 + 4x_6 \leq 182 \\ & 5x_4 + 3x_5 + 4x_6 \leq 157 \\ & x_5 + 3x_6 \leq 114 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{aligned} \tag{2}$$

We use the package `lpSolve` (linear programming) in the R environment to solve this problem. As shown below, our code[1] provides us with optimal quantities for every product. Notice that quantity for product x_5 is null, this is an example of *corner solution* which is very common in the linear programming framework. This solution implies that, at faced prices, endowments and current technical requirements, is not profitable to craft product x_5 at all.

```
     product quantity
[1,] "x1"    "17"
[2,] "x2"    "27"
[3,] "x3"    "34"
[4,] "x4"    "29"
[5,] "x5"    "0"
[6,] "x6"    "3"
```

[1] See the seller.R script in GitHub.

3 Some extentions

Insofar we have described a basic framework of the seller problem. Under this model we assumed that inputs cannot be sold in the market like products, also we