Chapter 7: Optimization of Linear Programs with the Simplex Algorithm

Contents

The simplex algorithm was developed by George Dantzig as way to optimize linear programs. The typical linear program conditions we will deal with are given by:

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b} \\ \mathbf{A^T}\lambda + \mathbf{s} &= \mathbf{c} \\ \mathbf{x} &\geq 0 \\ \mathbf{s} &\geq 0 \\ \mathbf{s^T}\mathbf{x} &= 0 \end{aligned}$$

To understand what the variables in the preceding equations are, we can look at an example. The book uses the carpenter example where we wish to

$$\begin{aligned} \text{maximize} \quad & 25x_1 + 30x_2 \\ & \text{subject to} \\ & 20x_1 + 30x_2 \leq 690 \\ & 5x_1 + 4x_2 \leq 120 \\ & \mathbf{x} \geq 0 \end{aligned}$$

In order to optimize the system, we turn our inequalities into equalities by using *slack* variables. Specifically, we would write

$$\begin{aligned} 20x_1 + 30x_2 + y_1 &= 690 \\ 5x_1 + 4x_2 + y_2 &= 120 \end{aligned}$$

where y_i are the slack variables. We can write the system of equations using matrix notation as

$$\underbrace{\begin{bmatrix} 20 & 30 & 1 & 0 \\ 5 & 4 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ y_1 \\ y_2 \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} 690 \\ 120 \end{bmatrix}}_{\mathbf{b}}$$

We can partition the matrix **A** and the vector **x** into two sets, the **columns which are not** (**N**) part of the current model and **those which are** (**B**). Specifically,

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{N} & \mathbf{B} \end{bmatrix} \\ \mathbf{x} &= \begin{bmatrix} \mathbf{x_N} \\ \mathbf{x_B} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{x_B} \end{bmatrix}; \end{aligned}$$

note that $\mathbf{x_N} = 0$ because these variables are not part of the model. The last piece of information which we take from the problem is the optimization function which defines the vector $\mathbf{c}$ such that we wish to minimize/maximize $\mathbf{c}^T\mathbf{x}$, so in our example

$$\mathbf{c}^T = \begin{bmatrix} 25 & 30 & 0 & 0 \end{bmatrix}$$

which we can again partition such that $\mathbf{c}^T = \begin{bmatrix} \mathbf{c_N} & \mathbf{c_B} \end{bmatrix}$. We can solve for the Langrange multipliers $\lambda, \mathbf{s}$ by using the fact that $\mathbf{s_B} = 0$; this condition implies that

$$\mathbf{B}^T\lambda = \mathbf{c_B} \quad \text{thus} \quad \lambda = \left(\mathbf{B}^T\right)^{-1}\mathbf{c_B}$$

and, subsequently,

$$\mathbf{N}^T\lambda + \mathbf{s_N} = \mathbf{c_N} \quad \text{thus} \quad \mathbf{s_N} = \mathbf{c_N} - \mathbf{N}^T\lambda.$$

Importantly, it can be shown that

$$\frac{\partial(\mathbf{c}^T\mathbf{x})}{\partial x_i} = s_i$$

thus the vector $\mathbf{s_N}$ is our **decision vector** and show how adding a variable that is currently not in the model (in $\mathbf{N}$) would change the objective function. If we are looking to maximize the objective, then, according the formulation above, we would choose the variable x_i that most *positively* changes $\mathbf{s_N}$ (if minimizing, then we want the variable that most negatively changes $\mathbf{s_N}$). Specifically, we choose

$$\text{argmax}_i(s_i)$$

where $i \in \mathbf{N}$ for maximization. Because the i^{th} variable enters the model, we must choose a variable to leave the model. To do this we will use a ratio test. We calculate $\mathbf{d} = \mathbf{B}^{-1}\mathbf{A_i}$ ($\mathbf{A_i}$ is the i^{th} column of $\mathbf{A}$). The leaving variable is then

$$\text{argmin}_i(x_i/d_i|d_i > 0)$$

where $i \in \mathbf{B}$; note the constraint that $d_i > 0$. Now that we have the indices i, j of both the entering and exiting variables, respectively, we then update $\mathbf{x}$ as follows:

$$x_{N,i} = \frac{x_j}{d_j}\mathbf{x_B} = \mathbf{x_B} - \mathbf{d}\frac{x_j}{d_j}$$

Finally, the indices in $\mathbf{B}, \mathbf{N}$ must be updated to reflect the new model. The process is the repeated until the signs of all the variable in $\mathbf{s_N}$ indicate the optimization function can not be increased/decreased any further.

Returning to the carpenter example, we wish to start the optimization at one of the extreme points of the system (i.e., a point where 2 of our constraints intersect). We can do this by choosing the origin, thus the first value of $\mathbf{x}$ is

$$\mathbf{x} = c(0, 0, 690, 120).$$

The indices of $\mathbf{N}$ are {1,2} and $\mathbf{B}$ are {3,4}. Thus,

$$
\begin{aligned}
\mathbf{B} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
\mathbf{c_B} &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
\lambda = \left(\mathbf{B}^T\right)^{-1} \mathbf{c_B} &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
\mathbf{s_N} = \mathbf{c_N} - \mathbf{N}^T \lambda &= \begin{bmatrix} 25 \\ 30 \end{bmatrix} - \begin{bmatrix} 20 & 5 \\ 30 & 4 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 25 \\ 30 \end{bmatrix}
\end{aligned}
$$

Because 30 is the largest value, we choose the index of $\mathbf{N}$ corresponding to its second index. In this case that would be element 2 of {1,2} which is 2, indicating that x_2 should enter the model. Next we perform the ratio test by

$$
\begin{aligned}
\mathbf{d} = \mathbf{B}^{-1}\mathbf{A_2} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 30 \\ 4 \end{bmatrix} = \begin{bmatrix} 30 \\ 4 \end{bmatrix} \\
\mathbf{x_B} &= \begin{bmatrix} 690 \\ 120 \end{bmatrix} \\
\mathbf{x_B}/\mathbf{d} &= \begin{bmatrix} \frac{690}{30} \\ \frac{120}{4} \end{bmatrix} = \begin{bmatrix} 23 \\ 30 \end{bmatrix}
\end{aligned}
$$

So the minimum index of $\mathbf{x_B}/\mathbf{d}|d_i > 0$ is 1 $(= 23)$, so the index of the leaving variable is the first element of {3,4}, or $x_3 (= y_1)$. To update $\mathbf{x}$, x_2 is then 23 and

$$
\begin{aligned}
\mathbf{x_B} &= \begin{bmatrix} 690 \\ 120 \end{bmatrix} - \begin{bmatrix} 30 \\ 4 \end{bmatrix} \times 23 = \begin{bmatrix} 0 \\ 28 \end{bmatrix} \quad \text{and} \\
\mathbf{x} &= \begin{bmatrix} 0 \\ 23 \\ 0 \\ 28 \end{bmatrix}
\end{aligned}
$$

The final step is to update the indices in $\mathbf{B}, \mathbf{N}$ to be {2,4} and {1,3}, respectively.

The process is then repeated. Now,

$$
\begin{aligned}
\mathbf{B} = \begin{bmatrix} \mathbf{A_2} & \mathbf{A_4} \end{bmatrix} &= \begin{bmatrix} 30 & 0 \\ 4 & 1 \end{bmatrix} \\
\mathbf{c_B} = \begin{bmatrix} c_2 \\ c_4 \end{bmatrix} &= \begin{bmatrix} 30 \\ 0 \end{bmatrix} \\
\lambda = \left(\mathbf{B}^T\right)^{-1} \mathbf{c_B} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
\mathbf{s_N} = \mathbf{c_N} - \mathbf{N}^T \lambda &= \begin{bmatrix} 25 \\ 0 \end{bmatrix} - \begin{bmatrix} 20 & 5 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ -1 \end{bmatrix}
\end{aligned}
$$

Because 5 is now the largest value, we choose the index of $\mathbf{N}$ corresponding to its first index (the indices are currently {1,3} thus the chosen index is 1, i.e. x_1 now enters the model). We again perform the ratio test to find the leaving variable:

$$
\begin{aligned}
\mathbf{d} = \mathbf{B}^{-1}\mathbf{A_1} &= \begin{bmatrix} 30 & 0 \\ 4 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 20 \\ 5 \end{bmatrix} = \begin{bmatrix} \frac{2}{3} \\ \frac{7}{3} \end{bmatrix} \\
\mathbf{x_B} &= \begin{bmatrix} 23 \\ 28 \end{bmatrix} \\
\mathbf{x_B}/\mathbf{d} &= \begin{bmatrix} \frac{69}{2} \\ 12 \end{bmatrix} = \begin{bmatrix} 34.5 \\ 12 \end{bmatrix}
\end{aligned}
$$

So the minimum index of $\mathbf{x_B}/\mathbf{d}|d_i > 0$ is 2 $(= 12)$, so the index of the leaving variable is the second element of {2,4}, or $x_4 (= y_2)$. To update $\mathbf{x}$, x_1 is then 12 and

$$
\begin{aligned}
\mathbf{x_B} &= \begin{bmatrix} 23 \\ 28 \end{bmatrix} - \begin{bmatrix} \frac{2}{3} \\ \frac{7}{3} \end{bmatrix} \times 12 = \begin{bmatrix} 15 \\ 0 \end{bmatrix} \quad \text{and} \\
\mathbf{x} &= \begin{bmatrix} 12 \\ 15 \\ 0 \\ 0 \end{bmatrix}
\end{aligned}
$$

Again, we update the indices in $\mathbf{B}, \mathbf{N}$ which are now {1,2} and {3,4}, respectively.

We iterate the process once again with the new values of $\mathbf{x}$:

$$
\begin{aligned}
\mathbf{B} = \begin{bmatrix} \mathbf{A_1} & \mathbf{A_2} \end{bmatrix} &= \begin{bmatrix} 20 & 30 \\ 5 & 4 \end{bmatrix} \\
\mathbf{c_B} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} &= \begin{bmatrix} 25 \\ 30 \end{bmatrix} \\
\lambda = \left(\mathbf{B}^T\right)^{-1} \mathbf{c_B} &= \begin{bmatrix} \frac{5}{7} \\ \frac{15}{7} \end{bmatrix} \\
\mathbf{s_N} = \mathbf{c_N} - \mathbf{N}^T \lambda &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{5}{7} \\ \frac{15}{7} \end{bmatrix} = \begin{bmatrix} -\frac{5}{7} \\ -\frac{15}{7} \end{bmatrix}
\end{aligned}
$$

Looking at our decision vector $\mathbf{s_N}$, all values are negative indicating that any entering variable would *decrease* the objective function, thus no longer maximizing its value. Therefore the current value of $\mathbf{x}^T = \begin{bmatrix} 12 & 15 & 0 & 0 \end{bmatrix}$ is a maximum for the linear program, i.e. $x_1 = 12$ and $x_2 = 15$.