CMS-A-DSE-B--1-TH: Operation Research (O.R.)
Lesson VI: Degeneracy in Linear Programming Problem

SPECIAL CASES IN THE SIMPLEX METHOD

This section considers four special cases that arise in the use of the simplex method.

1. Degeneracy
2. Alternative optima
3. Unbounded solutions
4. Nonexisting (or infeasible) solutions

Our interest in studying these special cases is twofold: (1) to present a *theoretical* explanation of these situations and (2) to provide a *practical* interpretation of what these special results could mean in a real-life problem.

Degeneracy

In the application of the feasibility condition of the simplex method, a tie for the minimum ratio may occur and can be broken arbitrarily. When this happens, at least one *basic* variable will be zero in the next iteration and the new solution is said to be **degenerate**.

There is nothing alarming about a degenerate solution, with the exception of a small theoretical inconvenience, called **cycling** or **circling**, which we shall discuss shortly. From the practical standpoint, the condition reveals that the model has at least one *redundant* constraint. To provide more insight into the practical and theoretical impacts of degeneracy, a numeric example is used.

CMS-A-DSE-B--1-TH: Operation Research (O.R.)
Lesson VI: Degeneracy in Linear Programming Problem

Example (Degenerate Optimal Solution)

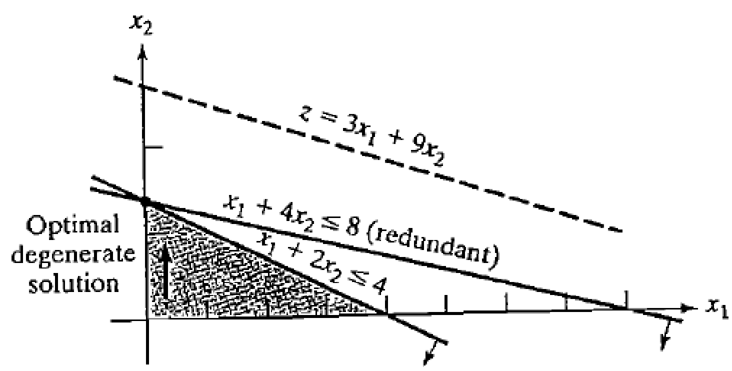
$$\text{Maximize } z = 3x_1 + 9x_2$$

subject to

$$x_1 + 4x_2 \le 8$$
$$x_1 + 2x_2 \le 4$$
$$x_1, x_2 \ge 0$$

Given the slack variables x_3 and x_4, the following tableaus provide the simplex iterations of the problem:

Iteration	Basic	x_1	x_2	x_3	x_4	Solution
0	z	-3	-9	0	0	0
x_2 enters	x_3	1	4	1	0	8
x_3 leaves	x_4	1	2	0	1	4
1	z	$-\frac{3}{4}$	0	$\frac{9}{4}$	0	18
x_1 enters	x_2	$\frac{1}{4}$	1	$\frac{1}{4}$	0	2
x_4 leaves	x_4	$\frac{1}{2}$	0	$-\frac{1}{2}$	1	0
2	z	0	0	$\frac{3}{2}$	$\frac{3}{2}$	18
(optimum)	x_2	0	1	$\frac{1}{2}$	$-\frac{1}{2}$	2
	x_1	1	0	-1	2	0



In iteration 0, x_3 and x_4 tie for the leaving variable, leading to degeneracy in iteration 1 because the basic variable x_4 assumes a zero value. The optimum is reached in one additional iteration.

In iteration 0, x_3 and x_4 tie for the leaving variable, leading to degeneracy in iteration 1 because the basic variable x_4 assumes a zero value. The optimum is reached in one additional iteration.

What is the practical implication of degeneracy? Look at the graphical solution in Figure Three lines pass through the optimum point ($x_1 = 0, x_2 = 2$). Because this is a two-dimensional problem, the point is *overdetermined* and one of the constraints is redundant.[2] In practice, the mere knowledge that some resources are superfluous can be valuable during the implementation of the solution. The information may also lead to discovering irregularities in the construction of the model. Unfortunately, there are no efficient computational techniques for identifying the redundant constraints directly from the tableau.

From the theoretical standpoint, degeneracy has two implications. The first is the phenomenon of **cycling** or **circling**. Looking at simplex iterations 1 and 2, you will notice that the objective value does not improve ($z = 18$). It is thus possible for the simplex method to enter a repetitive sequence of iterations, never improving the objective value and never satisfying the optimality condition Although there are methods for eliminating cycling, these methods lead to drastic slowdown in computations. For this reason, most LP codes do not include provisions for cycling, relying on the fact that it is a rare occurrence in practice.

The second theoretical point arises in the examination of iterations 1 and 2. Both iterations, though differing in the basic-nonbasic categorization of the variables, yield identical values for all the variables and objective value—namely,

$$x_1 = 0, x_2 = 2, x_3 = 0, x_4 = 0, z = 18$$

Is it possible then to stop the computations at iteration 1 (when degeneracy first appears), even though it is not optimum? The answer is no, because the solution may be *temporarily* degenerate

---

[2]Redundancy generally implies that constraints can be removed without affecting the feasible solution space. A sometimes quoted counterexample is $x + y \leq 1, x \geq 1, y \geq 0$. Here, the removal of any one constraint will change the feasible space from a single point to a region. Suffice it to say, however, that this condition is true only if the solution space consists of a single feasible point, a highly unlikely occurrence in real-life LPs.